

R.P. Gogate College of Arts & Science and  
R.V. Jogalekar College of Commerce  
(Autonomous), Ratnagiri



**Bachelor of Science  
In Information Technology  
(B.Sc. I.T.)**

Three Year Integrated Programme  
Six Semesters Course Structure  
Revised Scheme of Examination

Faculty of Science  
(Under-graduate Programme)

Choice Based Credit System (CBCS)  
**Academic Year- 2023-24**

R. E. Society's  
**R. P. Gogate College of Arts & Science And**  
**R. V. Jogalekar College of Commerce (Autonomous), Ratnagiri**  
**Department of Information Technology**

Name of Programme	<b>B.Sc. Information Technology</b>
Level	UG
No of Semesters	06
Year of Implementation	<b>2023-24</b>
Programme Specific Outcomes (PSO)	<ol style="list-style-type: none"> <li>1. Programming Proficiency: <ol style="list-style-type: none"> <li>a. PSO1: Proficient in Python and Java.</li> <li>b. PSO2: Design real-world applications.</li> <li>c. PSO3: Apply programming libraries for data analysis.</li> </ol> </li> <li>2. Networking, Database, and Data Structures: <ol style="list-style-type: none"> <li>a. PSO4: Understand computer networks and databases.</li> <li>b. PSO5: Configure and troubleshoot networks.</li> <li>c. PSO6: Implement data structures and algorithms.</li> </ol> </li> <li>3. Applied Mathematics and Statistics: <ol style="list-style-type: none"> <li>a. PSO7: Apply math and statistics for problem-solving.</li> </ol> </li> <li>4. Embedded Systems and Real-Time Applications: <ol style="list-style-type: none"> <li>a. PSO8: Develop real-time applications.</li> <li>b. PSO9: Utilize microcontrollers and sensors.</li> </ol> </li> </ol>
Relevance of PSOs to the local, regional, national, and global developmental needs (200 words)	The PSOs align with India's growing IT industry, Konkan region's local needs, national development, and global technology demands. Proficient programmers support the country's digital transformation, and networking expertise enhances connectivity. Applied math aids local challenges, while embedded systems find applications in healthcare and smart cities. These PSOs contribute to India's IT leadership, regional industries, and global tech market presence, addressing developmental needs at all levels.

**Under Choice Based Credit System (CBCS)  
Course Structure (Autonomous)  
Department of Information Technology**

No. of Courses	Semester I	Credits	No. of Courses	Semester II	Credits
	<b>Discipline Specific Course (DSC)</b>			<b>Discipline Specific Course (DSC)</b>	
	<b>Major</b>			<b>Major</b>	
USIT101	C Programming	02	USIT201	Object Oriented Programming with C++	02
USIT102	Web Programming	02	USIT202	Operating System	02
USIT103	Programming Practical 1	02	USIT203	Programming Practical 2	02
	<b>Minor</b>			<b>Minor</b>	
USIT104	Digital logic and Applications	02	USIT204	Data Communication and Networking	02
USIT105	Practical M1	02	USIT205	Microprocessor Architecture	02
			USIT206	Practical M2	02
	<b>Indian Knowledge System(IKS)</b>				
USIT106	Vedic Mathematics	02			---
	<b>Vocational Skill Course (VSC)</b>			<b>Skill Enhancement Course (SEC)</b>	
USIT107	Computer Skills-1 practical	02	USIT207	Computer Skills-2 Practical	02
	<b>Ability Enhancement Course (AEC)</b>			<b>Ability Enhancement Course (AEC)</b>	
USIT108	Technical Communication Skills I	02	USIT208	Technical Communication Skills II	02
	<b>Generic / Open Elective</b>			<b>Generic / Open Elective</b>	
	One from the pool of Open elective courses	02		One from the pool of Open elective courses	02
	<b>Value Education Course (VEC)</b>			<b>Value Education Course (VEC)</b>	
USES1	Environmental studies 1	02	USES2	Environmental studies 2	02
	<b>Co-Curricular</b>			<b>Co-Curricular</b>	
	Any one course from CC courses	02		Any one course from CC courses	02
<b>Total Credits</b>		<b>22</b>	<b>Total Credits</b>		<b>22</b>

**(To be implemented from Academic Year- 2023-24)**

# SEMESTER I

<b>SEMESTER I (Major)</b>				
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs./Week</b>
USIT101	C Programming	Theory	2	2
USIT102	Web Programming	Theory	2	2
USIT103	Programming Practical 1	Practical	2	4

<b>SEMESTER I (Minor)</b>				
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs./Week</b>
USIT104	Digital Logic and Applications	Theory	2	2
USIT105	M1 Practical	Practical	2	4

<b>SEMESTER I (Indian Knowledge System)</b>				
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs./Week</b>
USIT106	Vedic Mathematics	Theory	2	2

<b>SEMESTER I (Vocational Skill Enhancement Course)</b>				
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs./Week</b>
USIT107	Computer Skills-1 Practical	Practical	2	4

<b>SEMESTER I (Ability Enhancement Course)</b>				
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs./Week</b>
USIT108	Technical Communication Skills I	Theory	2	2

**SEMESTER I(Open Elective Course)(Any One to be Opted by IT Students)**

<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs.</b>
USOE101	Astronomy for Beginners	Theory/Practical	2	2
USOE102	Discrete Mathematics	Theory/Practical	2	2
USOE103	Google workspace and multimedia applications	Tutorial+Practical	1+1	1+2
USOE104	Health and Hygiene - I	Theory/Practical	2	2
USOE105	Health and Nutrition	Theory/Practical	2	2
USOE106	Social media marketing	Theory/Practical	2	2
USOE107	Plants in Human Welfare - I	Theory/Practical	2	2

**SEMESTER I (Value Education Course)Offered by Botany Department**

<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs./Week</b>
USES1	Environmental Studies I	Theory	2	2

Name of the Course	<b>C Programming</b>
Course Code (refer to student handbook)	<b>USIT101</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>I</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Major</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	Restructuring of syllabus has been done to ensure a smooth and logical flow of content throughout the curriculum. It also facilitates the logical progression of subjects which allows learners to build their understanding of the subject progressively and systematically and to grasp contents more effectively. The syllabus also focuses on practical programming exercises that require learners to apply the theory concepts and principles to real world scenarios. This will emphasize skill development among learners and will encourage learners to think critically and to analyze technical concepts from different perspectives. Additionally, some of the topics in this theory course will be covered in following Practical courses which will reinforce learners' theoretical understanding to real world applications.

## Nomenclature: C Programming

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Learn the basic principles of programming.
2. Develop logic using algorithms and flowchart.
3. Acquire the information about data types.
4. Understand input and output functions.
5. Enhance advanced concepts using programs.

### Curriculum:

Unit	Title	Learning Points	No of Hours
I	<b>Introduction</b>	What is Programming? Program Characteristics, Algorithms, Flowchart symbols, Algorithm Practice.	10
	<b>Foundation Of C</b>	History of C, Compiler, C Character Set, identifiers and keywords, data types and sizes , constants and its types ,variables, Character and character strings	
	<b>Type of operators</b>	Arithmetic operators, relational and logical operators, Increment and Decrement operators, assignment operators, the conditional operator, Assignment operators.	
II	<b>Control Flow</b>	Statements and Blocks, if-else, else-if, switch, Loops- while, for loops, do-while, break and continue, goto statement and Labels	10
	<b>Pointer, Arrays And Structure</b>	Pointer and Addresses, 1D, 2D, Multidimensional Array. Structure Definition	
III	<b>Functions and Program</b>	Basics of functions. User defined and Library functions, Types of functions, Function parameters, Return values, Recursion, Pointers to Functions, Dynamic memory allocation.	10
	<b>File management in C</b>	Defining and Opening file, Closing a file, Input / Output operations on file.	



**Learning Resources recommended:**

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1.	Programming in ANSI C	E.Balagurusamy	Tata McGraw-Hill	7th	1982
2.	Let us C	Yashwant P. Kanetkar	BPB publication		
3.	Programming with C	Byron Gottfried	Tata McGRAW-Hill	2nd	1996
4.	Mastering C	K R Venugopal	Tata McGraw-Hill	6th	2007
5.	Programming Language	Brian W. Kernighan and Denis M. Ritchie.	PHI	2nd	1988

**Evaluation Pattern:****A. Continuous Internal Evaluation (40 Marks):**

<b>Method</b>	<b>Marks</b>
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

**B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):**

Question No	Unit	Marks
1	I	15
2	II	15
3	III	15
4	I II III	05 05 05

**Guidelines for paper pattern for Semester End Evaluation:**

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
  3. Refer **Appendix I** for Paper pattern.
-

Name of the Course	<b>Web Programming</b>
Course Code (refer to student handbook)	<b>USIT102</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>I</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Major</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Web Programming

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand basic concepts of Internet and World Wide Web.
2. Comprehend different HTML elements that can be used to develop static web pages.
3. Become familiar with the concept of stylesheets and various CSS effects.
4. Pursue JavaScript as a tool to add dynamism to static HTML pages.
5. Explore how server-side script works on the web.

### Curriculum:

Unit	Title	Learning Points	No of Hours
I	Internet and the World Wide Web:	What is the Internet? Applications of Internet, E-mail, Telnet, FTP, E-commerce and E-business. Internet Service Providers, Domain Name Server, Internet Address, World Wide Web (WWW): World Wide Web and its Evolution, Uniform Resource Locator (URL), Browsers, Common Features of Browsers, Search Engine, Web Server, HTTP Protocol.	10
	HTML5:	Introduction, Formatting Text by using Tags, Using Lists, Creating Hyperlinks and Bookmarks, Defining Metadata about an HTML Document, Redirecting to another URL.	
	CSS:	Implementing Styles using CSS – Stylesheets, Formatting Text and Links using CSS, CSS Selectors, Changing Background, Adding Border, Margin and Padding, Setting Dimensions, Using Inline Container to mark up a part of a text	
II	HTML5 Page layout and navigation:	HTML Page Layout: Using Layout Elements, Semantic Elements, Creating, Positioning and Formatting Divisions, Floating Divisions next to each other, Responsive Web Design, Inline Frames. HTML Media, Embedding Images, Creating Client-side and Server-side Image Map, adding Favicon, Embedding audio and video on web page.	10
	Tables and Forms:	Creating Simple Table, Table Dimension, Merging Table Cells, Formatting Tables: Applying Borders, Background and Foreground fills, Changing Cell Padding, Spacing and Alignment Collecting user input with HTML Forms, Additional Input Types in HTML5.	
III	JavaScript:	Introduction, Difference between Client-side and Server-side Scripting, JavaScript Variables and Constants, Data Types, Statements, Comments,	10

		<p>Functions, Variable Scope, Hoisting, Strict Mode, JavaScript Objects, Dialog Boxes, void Keyword</p> <p>Operators: Arithmetic Operators, Assignment Operators, Comparison Operators, Logical Operators, Bitwise Operators</p> <p>Statements: Conditional Statements – if else, switch, Loops – while, do while, for, for in, for of, Loop Control – break, continue, labels</p> <p>JavaScript Objects: User-defined Objects, with Keyword, Native Objects – Array, String, Date, Math, Number, RegExp</p>	
		<p>DOM: Introduction, DOM Properties and Methods.</p> <p>Events and Event Handlers: HTML Events, DOM Events, DOM Event Listener, onAbort, onBlur, onChange, onClick, onDbClick, onError, onFocus, onKeyDown, onKeyPress, onKeyUp, onLoad, onMouseDown, onMouseMove, onMouseOut, onMouseOver, onMouseUp, onReset, onResize, onSelect, onSubmit, onUnload</p>	
	Introduction to PHP	<p>PHP: Introduction, Server-side Scripting, PHP Syntax and Comments, Variables and Constants, Data Types, Control Structures, Looping, Loop Termination, Functions, PHP Form Handling, PHP Form Validation, Superglobals, PHP Arrays, PHP Strings, PHP RegEx, PHP Numbers, PHP Math, Basic PHP Errors</p>	

### Learning Resources recommended:

Sr. No	Title	Author/s	Publisher	Edition	Year
1.	The Complete Reference HTML & CSS	Thomas A. Powell	McGrawHill	5 th	-
2.	Step by Step HTML5	Faithe Wempen	Microsoft Press	-	2011
3.	The Complete Reference JavaScript	Thomas A. Powell & Fritz Schneider	McGrawHill	3 rd	2012
4.	Learning Web Design A Beginner's Guide to Html, CSS, JavaScript, And Web Graphics	Jennifer Niederst Robbins	O'Reilly	5th	2018

5.	PHP & MySQL Novice to Ninja	Tom Butler	SPD	7th	2022
----	-----------------------------	------------	-----	-----	------

### Evaluation Pattern:

#### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks)	20
Assignments	10
Attendance and active participation in classroom	10

#### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	15
2	II	15
3	III	15
4	I II III	05 05 05

#### Guidelines for paper pattern for Semester End Evaluation:

4. All questions will be compulsory.
5. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
6. Refer **appendix I** for Paper pattern

Name of the Course	<b>Programming Practical 1</b>
Course Code (refer to student handbook)	<b>USIT103</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>I</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	<b>Major</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>In the previous Syllabus, a Mini project was added as a part of the practical list. As it is important in the sense of employability, entrepreneurship and skill development, it is decided to assign one credit to application development so that students will get deep knowledge and enough time for that.</b>

## Nomenclature: Programming Practical 1

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Work with textual information, characters and strings.
2. Understanding of a functional hierarchical code organization
3. Debug the program
4. Understand the differences between syntax errors, runtime errors, and logic errors.
5. Develop the application using C Programming language.

### Section I: C Programming Practical

#### Curriculum:

Sr. No.	Title	Learning Points	No of Hours
1	<b>Algorithm and Flowchart</b>	<ol style="list-style-type: none"><li>Write an algorithm and draw flowchart for Area of circle.</li><li>Write an algorithm and draw flowchart to print the given no. is even or odd.</li><li>Write an algorithm and draw flowchart to print 1 to 10 numbers.</li><li>Write an algorithm and draw flowchart for sum of 1 to 5 numbers.</li><li>Write an algorithm and draw flowchart to compute the addition of digits of a given number.</li></ol>	2
2	<b>Conditional Statements ,Goto Statements</b>	<ol style="list-style-type: none"><li>Write a program in C to check entered character vowel or consonant</li><li>Write a program to C program to print day name of week using switch-case</li><li>Write a program to read three values from the keyboard and print out the largest of them without using the if statement.</li><li>Write a program using goto statement.</li></ol>	2+2
3	<b>Loops</b>	<ol style="list-style-type: none"><li>Write a program using a while loop to reverse the digits of a number.</li><li>Write a program to calculate the factorial of a given number.</li><li>Write a program to print the Fibonacci series.</li></ol>	



<b>4</b>	<b>Programs on Patterns</b>	<p>a. Write a program to print the pattern of asterisks as shown below :</p> <pre> * * * * * * * * * * </pre> <p>b. Write a program to print the pattern of asterisks as shown below :</p> <pre> * * * * * * * * * * * * * * * </pre> <p>c. Write a program to print Floyd's Triangle.</p>	2+2
<b>5</b>	<b>Arrays</b>	<p>a. Write a program to print roll no and names of 10 students using an array.</p> <p>b. Write a program to read a matrix of size m*n.</p> <p>c. Write a program to find largest element of array.</p>	2+2
<b>6</b>	<b>Built-in and User Defined Functions</b>	<p>a. Write a program to print the area of a square using a function.</p> <p>b. Write a program to square root, abs() value using function.</p>	2
<b>7</b>	<b>Recursive Function</b>	<p>a. Write a program using a recursive function.</p>	2
<b>8</b>	<b>Pointers</b>	<p>a. Write a program to display the values using different data types and its address using a pointer.</p> <p>b. Write a program to perform addition and subtraction using a pointer.</p>	2+2
<b>9</b>	<b>Files</b>	<p>a. Write a program to copy the contents of the file from one file into another.</p>	2
<b>10</b>	<b>Files</b>	<p>a. Write a program to read text from the user and store that text into a file.</p>	2

This is sample Practical list. Course instructor may change the practical as per syllabus.

## Section II: Application Development using C

### Curriculum:

1. Each student has to select one topic for their application.
2. Application should be developed using C programming.
3. Individual students have to submit the project report (soft copy and executable application) (**Refer Appendix II**)

Sr. No.	Title	Learning Points	No. of Hours
0	<b>Feasibility Studies</b>	Decide Topic, Requirement Gathering Phase	4
1	<b>Design Phase-I</b>	Decide Module and Draw Algorithm	4
2	<b>Design Phase-II</b>	Design system, Draw Flowchart	4
3	<b>Implementation Phase</b>	Develop Module by coding phase	8
4	<b>Integration Phase</b>	Integration of modules	6
5	<b>Documentation</b>	Create a Documentation	4

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Programming in ANSI C	E.Balagurusamy	Tata McGraw-Hill	7th	1982
2.	Let us C	Yashwant P. Kanetkar	BPB publication		
3.	Programming with C	Byron Gottfried	Tata McGRAW-Hill	2nd	1996
4.	Mastering C	K R Venugopal	Tata McGraw-Hill	6th	2007
5.	Programming Language	Brian W. Kernighan and Denis M. Ritchie.	PHI	2nd	1988

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Certified Journal (section I)	10
Project Documentation(Section II)	10
Attendance and active participation in both Laboratory	20

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 4 hours):

#### Section I: (30 Marks - 2 Hours)

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

#### Section II: (30 Marks - 2 Hours)

Question No	Unit	Marks
1	User Interaction	05
2	C Syllabus Coverage	05
3	Running Application	15
4	Viva	05

Name of the Course	<b>Digital Logic and Applications</b>
Course Code (refer to student handbook)	<b>USIT104</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>I</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Minor(Interdisciplinary)</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Digital Logic and Applications

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. To introduce the basics of logic in digital electronics as an entry level course.
2. To interpret and assess number systems and the conversions of number systems
3. To analyze the boolean expressions and reduce the expression to the minimum.
4. To design simple logic circuits using tools such as Boolean algebra and Karnaugh Mapping.
5. To design and implement a variety of logical devices using combinational and sequential circuits concepts.
6. To introduce different applications of sequential circuits.

### Curriculum:

Unit	Title	Learning Points	No of Hours
I	<b>Digital Systems and Binary numbers</b>	Introduction to Number systems, Positional Number systems, Conversions (converting between bases), Non positional number systems, Unsigned and Signed binary numbers, Binary Codes, Number representation and storage in computer system.	10
	<b>Logic gates and Logic Circuits</b>	Basic and Universal Gates	
II	<b>Boolean algebra and Gate level minimization</b>	Introduction, Postulates of Boolean Algebra, Two Valued Boolean Algebra, Principle of Duality, Basic Theorems of Boolean Algebra, Boolean Functions and their Representation, Gate-Level Minimization (Simplification of Boolean Function), Quine- McCluskey Method, Review questions	10
	<b>Combinational logic</b>	Introduction, Analysis and Design Procedure for Combinational Logic Circuits, Types of Combinational Circuit, Review Questions	
III	<b>Sequential circuits</b>	Introduction, Latch, Flip-Flops, Registers, Counters, Review Questions	10
	<b>Applications</b>	Bit Arithmetic and Logic unit, Carry look ahead generator, Binary Multiplication and Division algorithm, Booth's multiplication algorithm.	

**Learning Resources recommended:**

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Digital Logic Design	Sonali Singh	BPB publications	1st	2015
2	Fundamentals of Digital Electronics and Logic Design	Subir Kumar Sarkar, Asish Kumar De, Souvil Sarkar	Pan Stanford Publishing	1st	2014
3	Digital Electronics Principles, Design and Applications	Anil K Maini	Wiley	1st	2007
4	Fundamentals of Logic Design	Charles H Roth, Jr., Larry L Kinney	Cengage Learning	7th	2014

**Evaluation Pattern:****A. Continuous Internal Evaluation (40 Marks):**

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks)	20
Assignments	10
Attendance and active participation in classroom	10

**B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):**

Question No	Unit	Marks
1	I	15
2	II	15
3	III	15
4	I	05
	II	05
	III	05

**Guidelines for paper pattern for Semester End Evaluation:**

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numerical based questions, single line answers, etc., will contain internal options.
  3. Refer **appendix I** for Paper pattern.
-

Name of the Course	<b>M1 Practical</b>
Course Code (refer to student handbook)	<b>USIT105</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>I</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	<b>Minor(Interdisciplinary)</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	



## Nomenclature: M1 Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able

1. To apply and test the gates learnt using various IC's .
2. To evaluate the Boolean expression to reduce and minimize the gates used
3. To understand different commands and functions of SCILAB.
4. To use tools to compute solutions of various discrete mathematical structures.

### Curriculum:

#### Section I: Digital Logic and Applications Practical

Unit	Title	Learning Points	No of Hours
1	<b>Study of basic gates and Universal gates</b>	<ol style="list-style-type: none"><li>To verify the truth tables of OR, AND, NOR, NAND, EX-OR, EX-NOR gates</li><li>To study IC 7400, 7402, 7404, 7408, 7432, 7486, 74266</li><li>To implement and verify NAND and NOR as Universal gates</li></ol>	2+2
2	<b>Study of Boolean expressions</b>	<ol style="list-style-type: none"><li>To verify De Morgan's laws</li><li>Implement the given expression using a minimum number of gates.</li><li>Implement the given expression using a minimum number of ICs.</li></ol>	2+2
3	<b>Design of Combinational Circuits using K-maps</b>	<ol style="list-style-type: none"><li>Design and implement combinational circuits for the given problem/problems using minimization techniques of K-maps.</li></ol>	2
4	<b>Design and implement code converters</b>	<ol style="list-style-type: none"><li>Design the circuit and implement Binary to gray code converter</li><li>Design the circuit and implement Gray to Binary code converter</li><li>Design the circuit and implement Binary to BCD code converter</li><li>Design the circuit and implement Binary to XS-3 code converter</li></ol>	2+2
5	<b>Implement Adder and Subtractor circuits</b>	<ol style="list-style-type: none"><li>Design the circuit and implement Half Adder and Full Adder</li><li>Design the circuit and implement BCD Adder, XS-3 Adder , Binary Subtractor</li></ol>	2
6	<b>Design and implement</b>	<ol style="list-style-type: none"><li>Design and implement 2-by-2 bit multiplier</li></ol>	2

	<b>Arithmetic circuits</b>		
7	<b>Implement Encoders and Decoders</b>	a. Design and implement 8: 3 encoder b. Design and implement 3:8 decoder	2
8	<b>Multiplexers and Demultiplexers</b>	a. Design and Implement 4:1 multiplexer b. Design and Implement 1:4 demultiplexer c. Study IC 74151 8: 1 multiplexer and implement the expression d. Study IC 74138 3: 8 decoder and implement the expression	2+2
9	<b>Study of Flip Flops and Counters</b>	a. Study of IC's 7473, 7474, and 7476 b. Design a 3-bit ripple/ synchronous counter using IC 7473 and required gates	2+2
10	<b>Design of Shift Registers</b>	a. Design of Shift registers using IC 7474 b. Implementation of digits using seven segment displays	2

This is sample Practical list. Course instructor may change the practical as per syllabus.

#### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Digital Logic Design	Sonali Singh	BPB publications	1st	2015
2	Fundamentals of Digital Electronics and Logic Design	Subir Kumar Sarkar, Asish Kumar De, Souvil Sarkar	Pan Stanford Publishing	1st	2014
3	Digital Electronics Principles, Design and Applications	Anil K Maini	Wiley	1st	2007
4	Fundamentals of Logic Design	Charles H Roth, Jr. , Larry L Kinney	Cengage Learning	7th	2014

## Section II: Introduction to Scilab tool Practical

Unit	Title	Learning Points	No of Hours
1	<b>Introduction to Scilab</b>	a. Basics of variables, operators b. Inbuilt functions c. Branches and looping statements	2+2
2	<b>Set Theory</b>	a. Inclusion Exclusion principle b. Cardinality and Power Sets c. Set Operations d. Properties of Sets	2+2
3	<b>Functions and Algorithms</b>	a. Recursively defined functions b. Roots of Polynomial c. Polynomial evaluation d. Greatest Common Divisor	2+2
4	<b>Probability Theory 1</b>	a. Sample space and events b. Finite probability spaces c. Equiprobable spaces d. Addition Principle	2+2
5	<b>Probability Theory 2</b>	a. Conditional Probability b. Multiplication theorem for conditional probability c. Independent events d. Repeated trials with two outcomes	2+2
6	<b>Counting 1</b>	a. Sum rule principle b. Product rule principle c. Factorial	2
7	<b>Counting 2</b>	a. Permutations b. Permutations with repetitions c. Combinations d. Combinations with repetitions	2
8	<b>Counting 3</b>	a. Ordered partitions b. Unordered partitions	2
9	<b>Graph Theory</b>	a. Paths and connectivity b. Minimum spanning tree c. Isomorphism	2
10	<b>Directed Graphs</b>	a. Adjacency matrix b. Path matrix	2

This is sample Practical list. Course instructor may change the practical as per syllabus.

### Learning Resources recommended:

<https://www.scilab.org/about/community/books>

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Certified Journal (section I and II)	20
Attendance and active participation in both Laboratory	20

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 4 hours):

#### Section I: (30 Marks - 2 Hours)

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

#### Section II:(30 Marks - 2 Hours)

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

Name of the Course	<b>Vedic Mathematics</b>
Course Code (refer to student handbook)	<b>USIT106</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>I</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Indian Knowledge System</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	Vedic Mathematics will help to eliminate math-phobia and increase speed and accuracy. It will give the student flexibility, fun and immense satisfaction, provides a powerful checking tool and saves precious time in examinations. It will develop Left & Right Sides of the brains by increasing visualization and concentration abilities. Knowledge of Vedic Mathematics will be helpful to crack the Numerical Aptitude part for students appearing for Competitive Examinations.

## Nomenclature: Vedic Mathematics

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Make critical thinking.
2. Recognize and understand simple techniques of Arithmetic Calculations.
3. Use the ideas of Vedic Mathematics in daily calculations and make those calculations with accuracy and speed.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	<b>Introduction</b>	History and Evolution of Vedic Mathematics, Techniques in Multiplication (Series of 9, Series of 1 etc.), Tables etc.,	10
	<b>Basic Operations</b>	Various techniques to carry out basic operations covering Addition, Subtraction, Multiplication (Vertically Crosswise), Division, Complements and Bases, Vinculum number	
II	<b>Techniques of Multiplication and Division</b>	Multiplications by numbers near base, Verifying answers by use of digital roots, Divisibility tests, Division of numbers near base, Comparison of fractions, Applications of Vinculum, Different methods of Squares (General method, Base method, Duplex method etc.)	10
III	<b>Equations</b>	Cubes, Cube roots, Square roots, Quadratic Equations, Simultaneous Equations	10
	<b>Numerical Aptitude</b>	Use of various Vedic Techniques for answering numerical aptitude questions from Competitive Examinations	

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	The Power of Vedic Maths	Atul Gupta	Jaico Publishing House	2nd	
2	Vedic Mathematics Made Easy	Dhaval Bhatiya	Dhaval Bhatiya		

**Evaluation Pattern:****A. Continuous Internal Evaluation (40 Marks):**

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks)	20
Assignments	10
Attendance and active participation in classroom	10

**B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):**

Question No	Unit	Marks
1	I	15
2	II	15
3	III	15
4	I II III	05 05 05

**Guidelines for paper pattern for Semester End Evaluation:**

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
3. Refer **appendix I** for Paper pattern.

Name of the Course	<b>Computer Skills-1 Practical</b>
Course Code (refer to student handbook)	<b>USIT107</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>I</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	<b>Vocational Skill Course</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	To demonstrate a basic understanding of computer hardware and software and ability of problem-solving skills. To apply logical skills to programming in a variety of languages and utilize web technologies. To present conclusions effectively, orally, and in writing. Developing profiles and email features.



## Nomenclature: Computer Skills-1 Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand how to effectively implement HTML.
2. Write CSS effectively to create well organized, styled web pages.
3. Add versatility to a web page with client-side scripting.
4. Deploy a local web server and run a simple web application.
5. Learn basic features of MS Word.
6. Articulate formal and informal reports.
7. Analyze and interpret data and learn visualization of data.
8. Learn effective tools of presentation.
9. Learn mail formation and create profiles.

### Section I: Web Programming Practical

#### Curriculum:

Sr. No.	Title	Learning Points	No of Hours
1	<b>Use of Basic Tags</b>	a. Design a web page using different text formatting tags. b. Design a web page with links to different pages and allow navigation between web pages. c. Design a web page that automatically redirects the user to another page.	2+2
2	<b>Use of CSS</b>	a. Design a web page demonstrating different stylesheet types. b. Design a web page demonstrating grouping selectors.	2+2
3	<b>Layout and Media</b>	a. Design a web page demonstrating different semantics. b. Design a web page embedding image, audio and video.	2
4	<b>Image Maps</b>	a. Design a web page with Image Maps.	2
5	<b>Tables and Forms</b>	a. Design a web page with different tables. b. Design a web page with a form that uses all types of controls.	2+2
6	<b>JavaScript</b>	a. Using JavaScript, design a web page to accept a number from the user and print its Factorial. b. Using JavaScript, a web page that prints Fibonacci series/any given series. c. Write a JavaScript program to display all the prime numbers between 1 and 100. d. Write a JavaScript program to accept a number from the user and display the sum of its digits.	2+2
7	<b>JavaScript Objects</b>	a. Using JavaScript, design a web page demonstrating different native objects of JavaScript.	2

		b. Write a program in JavaScript to accept a sentence from the user and display the number of words in it. (Do not use split () function).	
8	<b>JavaScript Events</b>	a. Write a JavaScript program to design a simple calculator. b. Design a form and validate all the controls placed on the form using JavaScript.	2
9	<b>PHP - Basic</b>	a. Write a PHP code to find the greater of 2 numbers. Accept the no. from the user. b. Write a PHP Program to accept a number from the user and print it factorial. c. Write a PHP program to accept a number from the user and print whether it is prime or not.	2
10	<b>PHP - Advanced</b>	a. Write a PHP program to demonstrate different string functions. b. Write a PHP program to demonstrate different array functions.	2+2

This is sample Practical list. Course instructor may change the practical as per syllabus.

**Learning Resources recommended:**

Sr. No	Title	Author/s	Publisher	Edition	Year
1.	The Complete Reference HTML & CSS	Thomas A. Powell	McGrawHill	5 th	-
2.	Step by Step HTML5	Faithe Wempen	Microsoft Press	-	2011
3.	The Complete Reference JavaScript	Thomas A. Powell & Fritz Schneider	McGrawHill	3 rd	2012
4.	Learning Web Design A Beginner's Guide to Html, CSS, JavaScript, And Web Graphics	Jennifer Niederst Robbins	O'Reilly	5th	2018
5.	PHP & MySQL Novice to Ninja	Tom Butler	SPD	7th	2022

## Section II: Basic Computer Skills Practical

### Curriculum:

Sr.No.	Title	Learning Points	No of Hours
1	<b>Various Components of Computer</b>	CPU, Parts of CPU, Keyboard, Mouse, Monitor and its types, Printer and its types, Storage devices- Pen drive, Memory card, CD and hard disc	2+2
2	<b>Memories and its Types</b>	RAM, ROM, Cache, Sequential, Virtual, Solidstate, Magnetic, Optical	2+2
3	<b>Windows Operating System</b>	Elements of Windows OS, Recycle bin, accessories, Calculator, Paint, Notepad, Wordpad, Control Panel	2+2
4	<b>Use of word processing tools</b>	What is word processing, Advantages of word processing, Saving New Document, Saving New Changes, View Buttons	2
5	<b>Features of word processing</b>	Building resumes in MS Word	2
6	<b>Basics of MS Excel</b>	Introduction to Microsoft Excel 2007, Workbook Details, Basic features and operations, Worksheets, Printing	2
7	<b>Use of presentation tools like PowerPoint</b>	Features of MS-Powerpoint	2
8	<b>Web Browser And E-mail</b>	Purpose, Protocols and Standards, Popular Browsers and Creating a Email and Use of Email	2+2
9	<b>Use of Mail etiquette for writing effective mails.</b>	Do's and Don'ts of Professional Emails	2
10	<b>Use of Mail merge</b>	a) Features of mail merge and steps b) Creating profile using LinkedIn.	2+2

This is sample Practical list. Course instructor may change the practical as per syllabus.

### Learning Resources recommended:

### Following websites were taken for reference:

1. <https://www.tutorialspoint.com>
2. <https://en.wikipedia.org>

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Certified Journal (section I and II)	20
Attendance and active participation in both Laboratory	20

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 4 hours):

#### Section I: (30 Marks - 2 Hours)

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

#### Section II:(30 Marks - 2 Hours)

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

Name of the Course	<b>Technical Communication Skills I</b>
Course Code (refer to student handbook)	<b>USIT108</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>I</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Ability Enhancement Course</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Technical Communication Skills I

### Course Outcomes:

On the successful completion of this course, the learner will be able

1. To recognize the importance of various types of communication in technical set up.
2. To understand the dynamics in different forms of formal communication.
3. To learn about active listening and the art of giving presentations and interviews.
4. To learn the art of business writing and ethics in business communication across functional areas.
5. To evaluate, analyze and interpret technical data.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	<b>The Seven Cs of Effective Communication:</b>	Completeness, Conciseness, Consideration, Concreteness, Clarity, Courtesy, Correctness	10
	<b>Fundamentals of Communication</b>	Introduction, The process of communication, Functions of communication	
	<b>Barriers to communication</b>	Introduction, Types of Barriers	
II	<b>Non-verbal Communication</b>	Introduction, Definition, significance of nonverbal, forms of non -verbal communication, types of non-verbal communication	10
	<b>Conversations</b>	Introduction, Conversation Management, Non-verbal cues in conversation	
	<b>Meeting and conferences</b>	Introduction, Purpose of Meeting, planning a meeting, Meeting Process, types of teleconferences, advantages and disadvantages.	
III	<b>Group Discussion and team presentation</b>	Introduction, Benefits of GD, Assessment of group discussion , Business and Purpose of Team presentation	10
	<b>Email communication</b>	Introduction, Email etiquettes, Techniques of writing Effective Email	
	<b>Active Listening</b>	Introduction, Type of listening, Barriers to effective listening	

**Learning Resources recommended:**

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1	Business Communication	Meenakshi Raman & Prakash Singh	Oxford- Higher Education	2nd edition	2006
2	Professional Communication	Aruna Koneru	McGraw Hill		2008
3	Professional Communication Skills	Laila Dias	Vipul Prakashan	1st edition	2010

**Evaluation Pattern:****A. Continuous Internal Evaluation (40 Marks):**

<b>Method</b>	<b>Marks</b>
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

**B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):**

<b>Question No</b>	<b>Unit</b>	<b>Marks</b>
1	I	15
2	II	15
3	III	15
4	I	05
	II	05
	III	05

**Guidelines for paper pattern for Semester End Evaluation:**

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.

Refer **appendix I** for Paper pattern.

---



# SEMESTER II

<b>SEMESTER II (Major)</b>				
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs./Week</b>
USIT201	Object Oriented Programming with C++	Theory	2	2
USIT202	Operating System	Theory	2	2
USIT203	Programming Practical 2	Practical	2	4

<b>SEMESTER II (Minor)</b>				
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs./Week</b>
USIT204	Data Communication and Networking	Theory	2	2
USIT205	Microprocessor Architecture	Theory	2	2
USIT206	M2 Practical	Practical	2	4

<b>SEMESTER II (Skill Enhancement Course)</b>				
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs./Week</b>
USIT207	Computer Skills-2 Practical	Practical	2	4

<b>SEMESTER II (Ability Enhancement Course)</b>				
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs./Week</b>
USIT208	Technical Communication Skills II	Theory	2	2

<b>SEMESTER II (Open Elective Course)(Any one to be Opted by IT Students)</b>				
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs.</b>
USOE201	Observational Astronomy	Theory	2	2
USOE202	Numerical Methods	Theory	2	2
USOE203	Basic computer system	Theory	2	2
USOE204	Health and Hygiene - II	Theory	2	2
USOE205	Health and Nutrition	Theory	2	2
USOE206	Introduction to Bioinformatics	Theory	2	2
USOE207	Plants in Human Welfare – II	Theory	2	2
USOE208	Advance Excel	Theory	2	2
USOE209	Food Science	Theory	2	2
USOE210	Social Media Awareness	Tutorial+Practical	1+1	1+2

<b>SEMESTER II (Value Education Course)Offered by Botany Department</b>				
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs./Week</b>
USES2	Environmental Studies II	Theory	2	2

Name of the Course	<b>Object Oriented Programming with C++</b>
Course Code (refer to student handbook)	<b>USIT201</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>II</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Major</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	Restructuring of syllabus has been done to ensure a smooth and logical flow of content throughout the curriculum. It also facilitates the logical progression of subjects which allows learners to build their understanding of the subject progressively and systematically and to grasp contents more effectively. The syllabus also focuses on practical programming exercises that require learners to apply the theory concepts and principles to real world scenarios. This will emphasize skill development among learners and will encourage learners to think critically and to analyze technical concepts from different perspectives. Additionally, some of the topics in this theory course will be covered in following Practical courses which will reinforce learners' theoretical understanding of real world applications.

**Nomenclature:** Object Oriented Programming with C++**Course Outcomes:**

On the successful completion of this course, the learner will be able to

1. Understand the concept of OOPs, features of C++ language.
2. Understand and apply various types of Data Types, Operators, Conversions while designing the program.
3. Understand and apply the concepts of Classes & Objects, friend function, constructors & destructors in program design.
4. Design & implement various forms of inheritance, String class, calling base class constructors.
5. Apply & Analyze operator overloading, runtime polymorphism, Generic Programming.
6. Analyze and explore various Stream classes, I/O operations and exception handling.

**Curriculum:**

Unit	Title	Learning Points	No of Hours
<b>I</b>	<b>Starting With C++</b>	C++ Overview, C++ Character Set, C++ Tokens, Variables, Counting Tokens, Data Types, Qualifiers, Range of Data Types, Your First C++ Program, Programming Examples	10 hrs
	<b>Features Of C++</b>	Introduction, Operators and Expressions, Declaring Constants, Type Conversion, Decision Making: An Introduction, Unconditional Branching Using Goto, Introduction to Looping	
	<b>Operators And References In C++</b>	Introduction, Scope Resolution Operator, Reference Variables, The Bool Data Type, The Operator New and Delete, Malloc Vs. New, Pointer Member Operators	
	<b>Introduction to Object-Oriented Concepts</b>	Introduction To Structure, Structure vs. class, Objects, Class and Instance, Abstraction, Encapsulation, Polymorphism, Inheritance, Message Passing, Dynamic Binding,	
<b>II</b>	<b>Function In C++</b>	Call by Reference, Inline Function, Function Overloading, Function with Default Arguments	10 hrs
	<b>Class and Objects In C++</b>	Working with Class, Structure in C++, Accessing Private Data Passing and Returning Object, Array of Object, Introduction Friend Function, Static Class Members, Constant Member Function	

	<b>Working With Constructor And Destructor</b>	Introduction, Constructor with Parameters, Implicit and Explicit Call to Constructor, Copy Constructor, Dynamic Initialization of Objects, Dynamic Constructor, Destructor	
	<b>Working With Operator Overloading</b>	Introduction, Operator Overloading	
<b>III</b>	<b>Working With Inheritance In C++</b>	Introduction, Types of Inheritance, Public, Private and Protected Inheritance, Multiple Inheritance, Hierarchical Inheritance, Virtual Base Class, Constructor and Destructor in Inheritance	10 hrs
	<b>Pointers To Objects And Virtual Functions</b>	Pointer to Objects, The This Pointer, Virtual Functions, Working of a Virtual Function, Rules for Virtual Function, Pure Virtual Function and Abstract Class	
	<b>File Handling In C++</b>	Introduction, File Streams, Opening and Closing a File, File Opening Modes Checking End of File	
	<b>Template Programming</b>	Introduction, Function Template, Class Template	
	<b>Exception Handling In C++</b>	Introduction, Basics of Exception Handling, Exception Handling Mechanism,	

### Learning Resources recommended:

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1.	Object Oriented Programming in C++	E Balagurusamy	Tata McGraw-Hill	5th Edition	2011
2.	Object-oriented Programming C++ Simplified	Hari Mohan Pandey	University Science Press	1st Edition	2017
3.	Object-Oriented Programming in C++	Robert Lafore	Sams	4th Edition	2002
4.	Programming with ANSI C++	Bhushan Trivedi	Oxford University Press	2nd Edition	2012
5.	Demystified Object-Oriented Programming with C++	Dorothy R. Kirk	Packt Publishing Lt	1st Edition	2021

6.	C++ Programming: An Object-Oriented Approach	Behrouz A. Forouzan , Richard F. Gilberg	McGraw-Hill Education	1st edition	2020
7.	C++ How to Program	Paul Deitel, Harvey Deitel	Pearson Education	10th Edition	2017

### Evaluation Pattern:

#### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

#### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	15
2	II	15
3	III	15
4	I II III	05 05 05

#### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
3. Refer **appendix I** for Paper pattern.

Name of the Course	<b>Operating System</b>
Course Code(refer to student handbook)	<b>USIT202</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>II</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type(applicable to NEP only)	<b>Major</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	Restructuring of syllabus has been done to ensure a smooth and logical flow of content throughout the curriculum. It also facilitates the logical progression of subjects which allows learners to build their understanding of the subject progressively and systematically and to grasp contents more effectively. The syllabus also focuses on problem solving exercises that require learners to apply the theory concepts and principles to real world scenarios. This will emphasize skill development among learners and will encourage learners to think critically and to analyze technical concepts from different perspectives. Additionally, some of the topics in this theory course will be covered in following Practical courses which will reinforce learners' theoretical understanding to real world applications



## Nomenclature: Operating Systems

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand Functions, Services and structure of Operating Systems.
2. Understand processes, threads, schedulers and explanation of CPU scheduling.
3. Understand issues related to Process Synchronization and focus on principles of Deadlock and related problems.
4. Comprehend the mechanisms used in Memory Management and Virtual Memory.
5. Understand the concepts of File System, secondary storage management and Disk Scheduling

### Curriculum:

Unit	Title	Learning Points	No of Hours
I	<b>Introduction</b>	What Is An Operating System? History Of Operating System, Computer Hardware, Different Operating Systems, Operating System Concepts, System Calls, Operating System Structure.	10
	<b>Processes and Threads</b>	Processes, Threads, Interprocess Communication, Scheduling, IPC Problems.	
II	<b>Memory Management</b>	No Memory Abstraction, Memory Abstraction: Address Spaces, Virtual Memory, Page Replacement Algorithms, Design Issues For Paging Systems, Implementation Issues, Segmentation.	10
	<b>File Systems</b>	Files, Directories, File System Implementation, File-System Management And Optimization, Ms-Dos File System, Unix V7 File System, CD-ROM File System.	
III	<b>Deadlock</b>	Resources, Introduction To Deadlocks, The Ostrich Algorithm, Deadlock Detection And Recovery, Deadlock Avoidance, Deadlock Prevention, Issues.	10
	<b>Study of LINUX and ANDROID</b>	History Of Unix And Linux, Linux Overview, Processes In Linux, Memory Management In Linux, I/O In Linux, Linux File System, Security In Linux. Android.	

	<b>Study of Windows</b>	History Of Windows Through Windows 10, Programming Windows, System Structure, Processes And Threads In Windows, Memory Management, Caching In Windows, I/O In Windows, Windows Nt File System, Windows Power Management, Security In Windows	
--	-------------------------	--	--

**Learning Resources recommended:**

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1.	Modern Operating Systems	Andrew S. Tanenbaum, Herbert Bos	Pearson	4 <sup>th</sup>	2014
2.	Operating Systems – Internals and Design Principles	Willaim Stallings	Pearson	8 <sup>th</sup>	2009
3.	Operating System Concepts	Abraham Silberschatz, Peter B. Galvineg Gagne	Wiley	8 <sup>th</sup>	
4.	Operating Systems	Godbole and Kahate	McGraw Hill	3 <sup>rd</sup>	

**Evaluation Pattern:**

**A. Continuous Internal Evaluation (40 Marks):**

<b>Method</b>	<b>Marks</b>
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

**B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):**

Question No	Unit	Marks
1	I	15
2	II	15
3	III	15
4	I II III	05 05 05

**Guidelines for paper pattern for Semester End Evaluation:**

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
  3. Refer **appendix I** for Paper pattern.
-

Name of the Course	<b>Programming Practical 2</b>
Course Code (refer to student handbook)	<b>USIT203</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>II</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	<b>Major</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	In the previous Syllabus, a Mini project was added as a part of the practical list. As it is important in the sense of employability, entrepreneurship and skill development, it is decided to assign one credit to application development so that students will get deep knowledge and enough time for that.

## Nomenclature: Programming Practical 2

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Utilize C++ characteristics in software design and development.
2. Explain object-oriented techniques and explain how C++ supports them.
3. Employ C++ to demonstrate practical skill developing object-oriented solutions.
4. Examine a problem statement and design and develop object-oriented software using good Coding practices and procedures.
5. Use common software patterns and recognize their relevance in other software development contexts.

### Section I: Object Oriented Programming with C++ Practical

#### Curriculum:

Sr.No.	Title	Learning Points	No of hrs
1	<b>C++ Basics Programs</b>	<ol style="list-style-type: none"><li>a. Write a C++ program to create a simple calculator.</li><li>b. Write a C++ program to convert seconds into hours, minutes and seconds.</li><li>c. Write a C++ program to find the volume of a square, cone, and rectangle.</li></ol>	2
2	<b>Conditional Statement and Looping using C++</b>	<ol style="list-style-type: none"><li>a. Write a C++ program to find the greatest of three numbers.</li><li>b. Write a C++ program to find the sum of even and odd n natural numbers</li><li>c. Write a C++ program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.</li></ol>	2+2
3	<b>Classes and Methods</b>	<ol style="list-style-type: none"><li>a. Write a C++ program using classes and object Student to print the name of the student, roll_no.Display the same.</li><li>b. Write a C++ program for Structure bank employee to print name of the employee,account_no. &amp; balance. Display the same also display the balance after withdraw and deposit</li><li>c. Write a C++ Program to design a class having static member function named showcount() which has the property of displaying the number of objects created of the class.</li><li>d. Write a Program to find Maximum out of Two Numbers using the friend function. (Note: Here one number is a member of one class and the other number is member of some other class.)</li></ol>	2+2

		e. Write a C++ Program using a copy constructor to copy data of an object to another object.	
<b>4</b>	<b>Constructor and Overloading</b>	a. Write a C++ program to design a class of performing addition & multiplication of two numbers using operator overloading. b. Write a C++ program to overload new/delete operators in a class. c. Write a C++ Program to generate Fibonacci Series by using Constructor to initialize the Data Members.	2+2
<b>5</b>	<b>Inheritance I</b>	a. Write a C++ Program that illustrates single inheritance. b. Write a C++ Program that illustrates multiple inheritance. c. Write a C++ Program that illustrates multilevel inheritance. d. Write a C++ Program that illustrates Hierarchical inheritance.	2+2
<b>6</b>	<b>Inheritance II</b>	a. Write a C++ Program to design a student class representing student roll no. and a test class (derived class of student) representing the scores of the student in various subjects and sports class representing the score in sports. The sports and test class should be inherited by a result class having the functionality to add the scores and display the final result for a student.	2+2
<b>7</b>	<b>Use of this pointer</b>	a. Write a C++ program to maintain the records of person with details (Name and Age) and find the eldest among them. The program must use this pointer to return the result.	2
<b>8</b>	<b>File Handling</b>	a. Write a C++ program to copy the contents of one file to another.	2
<b>9</b>	<b>Exception Handling</b>	a. Write a C++ program to implement the exception handling with multiple catch statements.	2
<b>10</b>	<b>Template Programming</b>	a. Write a C++ Program to create Simple calculator using a Class template. b. Write a C++ Program to get a maximum of two numbers using the Class template.	2

This is sample Practical list. Course instructor may change the practical as per syllabus.

**Learning Resources recommended:**

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1.	Object Oriented Programming in C++	E Balagurusamy	Tata McGraw-Hill	5th Edition	2011
2.	Object-oriented Programming C++ Simplified	Hari Mohan Pandey	University Science Press	1st Edition	2017

## Section II: Application Development using C++

### Curriculum:

1. Each student has to select one topic for their application.
2. Applications should be developed using C++ programming.
3. Individual students have to submit the project report (soft copy and executable application) (**Refer Appendix II**)

### Practical list:

Sr. No.	Title	Learning Points	No. of Hours
0	Feasibility Studies	Decide Topic, Requirement Gathering Phase	4
1	Design Phase-I	Decide Module and Draw Algorithm	4
2	Design Phase-II	Design system, Draw Flowchart	4
3	Implementation Phase	Develop Module by coding phase	8
4	Integration Phase	Integration of modules	6
5	Documentation	Create a Documentation	4

### Evaluation Pattern

#### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Certified Journal (section I)	10
Project Documentation(Section II)	10
Attendance and active participation in both Laboratory	20



**B. Semester End Evaluation (Paper Pattern) (60 Marks – 4 hours):**

**Section I: (30 Marks - 2 Hours)**

<b>Question No</b>	<b>Unit</b>	<b>Marks</b>
<b>1</b>	<b>Program 1</b>	<b>10</b>
<b>2</b>	<b>Program 2</b>	<b>15</b>
<b>3</b>	<b>Viva</b>	<b>05</b>

**Section II: (30 Marks - 2 Hours)**

<b>Question No</b>	<b>Unit</b>	<b>Marks</b>
<b>1</b>	<b>User Interaction</b>	<b>05</b>
<b>2</b>	<b>C++ Syllabus Coverage</b>	<b>05</b>
<b>3</b>	<b>Running Application</b>	<b>15</b>
<b>4</b>	<b>Viva</b>	<b>05</b>

Name of the Course	<b>Data Communication and Networking</b>
Course Code (refer to student handbook)	<b>USIT204</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>II</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Minor (Interdisciplinary)</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	Basics of Data Communication and networking is introduced to learners through the course. It will help learners to choose “Networking” as their employable field in early stages of the curriculum.

## Nomenclature: Data Communication and Networking

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Identify various data communication standards, topologies and terminologies
2. Describe how signals are used to transfer data and communication aspects between nodes
3. Study of underlying Technologies for networking.

### Curriculum:

Unit	Title	Learning Points	No of Hours
I	<b>Introduction</b>	Data communications, Networks, The Internet, Protocols and Standards	10
	<b>Network Models</b>	Layered tasks, The OSI model, Layers in the OSI Model, TCP/IP protocol suite, Addressing,	
	<b>Data and Signals</b>	Analog and Digital, Periodic Analog signals, Digital signals, Transmission Impairment, Data rate limits, Performance	
II	<b>Digital and Analog Transmission</b>	Digital-To-Digital Conversion, Analog-To-Digital Conversion, Transmission Modes, Digital-To-Analog Conversion, Analog-To-Analog Conversion	10
	<b>Bandwidth Utilization and Transmission Media</b>	Multiplexing, Spread Spectrum, Guided Media, Unguided Media: Wireless, Circuit-Switched Networks, Packet Switching	
III	<b>Data Link Layer</b>	Introduction, Error Detection And Correction- Introduction, Block Coding, Cyclic Codes, Checksum, Forward Error Correction, Data-Link Layer Protocols, HDLC, Point-To-Point Protocol (PPP), Data Link Control, Multiple Access, Random Access, Controlled Access, Channelization	10
	<b>Wired LANs</b>	Ethernet Protocol, Standard, Fast, Gigabit, 10 Gigabit Ethernet, Telephone, Cable Networks	
	<b>Wireless LANs and WANs</b>	Introduction, IEEE 802.11 Project, Bluetooth, WiMAX, Cellular Telephony, Satellite Networks	

**Learning Resources recommended:**

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Data Communication and Networking	Behrouz A. Forouzan	Tata McGraw Hill	Fiffth	-----
2.	Computer Networks	Andrew Tanenbaum	Pearson	Fifth	2013
3.	TCP/IP Protocol Suite	Behrouz A. Forouzan	Tata McGraw Hill	Fourth Edition	2010

**Evaluation Pattern:****A. Continuous Internal Evaluation (40 Marks):**

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

**B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):**

Question No	Unit	Marks
1	I	15
2	II	15
3	III	15
4	I II III	05 05 05

**Guidelines for paper pattern for Semester End Evaluation:**

- All questions will be compulsory.
- Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
- Refer **appendix I** for Paper pattern.

Name of the Course	<b>Microprocessor Architecture</b>
Course Code (refer to student handbook)	<b>USIT205</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>II</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Minor (Interdisciplinary)</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	In this revision of Microprocessor Architecture, a strong focus is placed on enhancing employability and fostering entrepreneurship through skill development. The chapter "Microprocessor Architecture and Microcomputer System" explores the practical applications of microprocessors in real-world systems, providing students with valuable industry-relevant knowledge. "Interfacing of I/O Devices" equips learners with hands-on experience in connecting peripherals to microprocessors, enhancing their technical competence. "Introduction to 8085 Assembly Language" and "Programming Techniques" foster critical problem-solving skills and algorithmic thinking, making students proficient in programming microprocessors for various applications. Additionally, "Interrupts" and "Stacks and Subroutines" empower students to design efficient and responsive microprocessor-based systems, creating a solid foundation for future employability or entrepreneurial endeavors.

## Nomenclature: Microprocessor Architecture

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand the basic concepts of Micro Computer Systems
2. Understand the architecture and hardware aspects of 8085
3. Write assembly language programs in 8085

### Curriculum:

Unit	Title	Learning Points	No of Hours
I	<b>Introduction</b>	Microprocessor, microcomputers, and Assembly Language: Microprocessor, Microprocessor Instruction Set and Computer Languages, From Large Computers to Single-Chip Microcontrollers, Applications.	10
	<b>Microprocessor Architecture and Microcomputer System</b>	Microprocessor Architecture and its operations, Memory, I/O Devices, Microcomputer System, Logic Devices and Interfacing, Microprocessor-Based System Application.	
	<b>8085 Microprocessor Architecture and Memory Interface</b>	Introduction, 8085 Microprocessor unit, 8085-Based Microcomputer, Memory Interfacing, Interfacing the 8085 Memory Segment.	
II	<b>Interfacing of I/O Devices</b>	Basic Interfacing concepts, Interfacing Output Displays, Interfacing Input Devices, Memory Mapped I/O, Testing and Troubleshooting I/O Interfacing Circuits.	10
	<b>Introduction to 8085 Assembly Language</b>	The 8085 Programming Model, Instruction Classification, Instruction, Data and Storage, Writing assembling and Execution of a simple program, Overview of 8085 Instruction Set, Writing and Assembling Program.	
	<b>Introduction to 8085 Instructions:</b>	Data Transfer Operations, Arithmetic Operations, Logic Operation, Branch Operation, Writing Assembly Languages Programs, Debugging a Program.	

<b>III</b>	<b>Programming Techniques</b>	Programming Techniques With Additional Instructions: Looping, Counting and Indexing, Additional Data Transfer and 16-Bit Arithmetic Instructions, Arithmetic Instruction Related to Memory, Logic Operations: Rotate, Logics Operations: Compare, Dynamic Debugging.	10
	<b>Counters and Time Delays</b>	Counters and Time Delays, Illustrative Program: Hexadecimal Counter, Illustrative Program: zero-to-nine (Modulo Ten) Counter, Generating Pulse Waveforms, Debugging Counter and Time-Delay Programs.	
	<b>Stacks and Subroutines</b>	Stack, Subroutine, Restart, Conditional Call, Return Instructions, Advanced Subroutine concepts.	
	<b>Interrupts</b>	The 8085 Interrupt, 8085 Vectored and Non vectored Interrupts, Restart as S/W Instructions.	

#### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Microprocessors Architecture, Programming and Applications with the 8085.	Ramesh Gaonkar	PENRAM	5th	2012
2	8080A/8085 Assembly Language Programming	Lance A. Leventhel	Osborne		1978

#### Evaluation Pattern:

##### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

**B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):**

Question No	Unit	Marks
1	I	15
2	II	15
3	III	15
4	I II III	05 05 05

**Guidelines for paper pattern for Semester End Evaluation:**

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
  3. Refer **appendix I** for Paper pattern.
-



Name of the Course	<b>M2 Practical</b>
Course Code (refer to student handbook)	<b>USIT206</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>II</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	<b>Minor(Interdisciplinary)</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

## Nomenclature: M2 Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Apply concepts of 8085 to single & Multiple Memory Locations
2. Apply concepts of microprocessor register operations
3. Implement assembly language programs
4. Find fast and accurate solution to simple and complex numerical problems using scilab programs

### Section I: Microprocessor Architecture Practical

#### Curriculum:

Sr. No.	Title	Learning Points	No of Hours
1	<b>Perform the following Operations related to memory locations</b>	<ol style="list-style-type: none"><li>a. Store the data byte 32H into memory location 4000H.</li><li>b. Exchange the contents of memory locations 2000H and 4000H</li><li>c. Find the 1's complement of the number stored at memory location 4400H and store the complemented number at memory location 4300H.</li><li>d. Find the 2's complement of the number stored at memory location 4200H and store the complemented number at memory location 4300H.</li></ol>	2+2
2	<b>Simple assembly language programs</b>	<ol style="list-style-type: none"><li>a. Add the contents of memory locations 4000H and 4001H and place the result in the memory locations 4002H and 4003H.</li><li>b. Subtract the contents of memory location 4001H from the memory location 4000H and place the result in memory location 4002H.</li><li>c. Add the 16-bit number in memory locations 4000H and 4001H to the 16-bit number in memory locations 4002H and 4003H. The most significant eight bits of the two numbers to be added are in memory locations 4001H and 4003H. Store the result in memory locations 4004H and 4005H with the most significant byte in memory location 4005H.</li><li>d. Subtract the 16-bit number in memory locations 4002H and 4003H from the 16-bit number in memory locations 4000H and 4001H. The most significant eight bits of the two numbers are in memory locations 4001H</li></ol>	2+2

		and 4003H. Store the result in memory locations 4004H and 4005H with the most significant byte in memory location 4005H.	
<b>3</b>	<b>Packing and unpacking operations</b>	<p>a. Pack the two unpacked BCD numbers stored in memory locations 4200H and 4201H and store the result in memory location 4300H. Assume the least significant digit is stored at 4200H.</p> <p>b. Two digit BCD number is stored in memory location 4200H. Unpack the BCD number and store the two digits in memory locations 4300H and 4301H such that memory location 4300H will have lower BCD digit.</p>	2
<b>4</b>	<b>Register Operations.</b>	<p>a. Write a program to shift an eight bit data four bits right. Assume that data is in register C.</p> <p>b. Program to shift 16-bit data 1 bit left. Assume data is in the HL register pair.</p> <p>c. Write a set of instructions to alter the contents of the flag register in 8085.</p> <p>d. Write a program to count number of I's in the contents of D register and store the count in the B register</p>	2+2
<b>5</b>	<b>Multiple memory locations</b>	<p>a. Calculate the sum of a series of numbers. The length of the series is in memory location 4200H and the series begins from memory location 4201H.</p> <p>I. Consider the sum to be an 8 bit number. So, ignore carries. Store the sum at memory location 4300H.</p> <p>II. Consider the sum to be a 16 bit number. Store the sum at memory locations 4300H and 4301H.</p> <p>b. Multiply two 8-bit numbers stored in memory locations 2200H and 2201H by repetitive addition and store the result in memory locations 2300H and 2301H.</p> <p>c. Divide the 16 bit number stored in memory locations 2200H and 2201H by the 8 bit number stored at memory location 2202H. Store the quotient in memory locations 2300H and 2301H and remainder in memory locations 2302H and 2303H.</p>	2+2

6	<b>Calculations with respect to memory locations</b>	<p>a. Find the largest number in a block of data. The length of the block is in memory location 2200H and the block itself starts from memory location 2201H. Store the maximum number in memory location 2300H. Assume that the numbers in the block are all 8 bit unsigned binary numbers.</p> <p>b. Write a program to sort given 10 numbers from memory location 2200H in the ascending order.</p> <p>c. Calculate the sum of a series of even/odd numbers from the list of numbers. The length of the list is in memory location 2200H and the series itself begins from memory location 2201H. Assume the sum to be 8 bit number so you can ignore carries and store the sum at memory location 2 Sample problem.</p>	2+2
7	<b>Assembly programs on memory locations.</b>	<p>a. Write an assembly language program to separate even/odd numbers from the given list of 50 numbers and store them in another list starting from 2300H. Assume starting address of 50 number list is 2200H.</p> <p>b. Find the square of the given numbers from memory location 6100H and store the result from memory location 7000H.</p> <p>c. A list of 50 numbers is stored in memory, starting at 6000H. Find number of negative, zero and positive numbers from this list and store these results in memory locations 7000H, 7001H, and 7002H respectively.</p>	2
8	<b>String operations in assembly programs.</b>	<p>a. Multiply the 8-bit unsigned number in memory location 2200H by the 8-bit unsigned number in memory location 2201H. Store the 8 least significant bits of the result in memory location 2300H and the 8 most significant bits in memory location 2301H.</p> <p>b. Divide the 16-bit unsigned number in memory locations 2200H and 2201H (most significant bits in 2201H) by the B-bit unsigned number in memory location 2300H store the quotient in memory location 2400H and remainder in 2401H</p>	2
9	<b>Calculations on memory locations</b>	<p>a. Arrange an array of 8 bit unsigned no in descending order.</p> <p>b. Transfer ten bytes of data from one memory to another memory block. Source memory block starts from memory location 2200H whereas destination memory block starts from memory location 2300H.</p>	2

<b>10</b>	<b>Operations on BCD numbers.</b>	a. Add two 4 digit BCD numbers in HL and DE register pairs and store the result in memory locations, 2300H and 2301H. Ignore carry after 16 bit. b. Subtract the BCD number stored in E register from the number stored in the D register. c. Write an assembly language program to multiply 2 BCD numbers	2
-----------	-----------------------------------	--	---

This is sample Practical list. Course instructor may change the practical as per syllabus.

**Learning Resources recommended:**

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Microprocessors Architecture, Programming and Applications with the 8085.	Ramesh Gaonkar	PENRAM	Fifth	2012
2	8080A/8085 Assembly Language Programming	Lance A. Leventhel	Osborne		1978

## Section II: Numerical Methods Practical

### Curriculum:

Sr. No.	Title	Learning Points	No of Hours
1	<b>Error Calculation</b>	a. Program to calculate the roots of a quadratic equation using the formula b. Program for error calculation c. Program to evaluate $e^x$ using infinite series	2+2
2	<b>Solution of algebraic and transcendental equations</b>	a. Program to solve algebraic and transcendental equation by bisection method. b. Program to solve algebraic and transcendental equation by false position method. c. Program to solve algebraic and transcendental equation by Secant method. d. Program to solve algebraic and transcendental equation by Newton Raphson method.	2+2
3	<b>Interpolation</b>	a. Program for Newton's forward interpolation. b. Program for Newton's backward interpolation. c. Program for Lagrange's interpolation.	2+2
4	<b>Solving linear system of equations by iterative methods</b>	a. Program for solving linear systems of equations using Gauss Jordan method. b. Program for solving linear systems of equations using Gauss Seidel method.	2
5	<b>Numerical Differentiation</b>	a. Program to obtain derivatives numerically.	2
6	<b>Numerical Integration</b>	a. Program for numerical integration using Trapezoidal rule. b. Program for numerical integration using Simpson's 1/3rd rule. c. Program for numerical integration using Simpson's 3/8th rule.	2+2
7	<b>Solution of differential equations 1</b>	a. Program to solve differential equations using Euler's method. b. Program to solve differential equations using modified Euler's method.	2
8	<b>Solution of differential equations 2</b>	a. Program to solve differential equations using Runge-kutta 2nd order and 4th order methods.	2
9	<b>Regression 1</b>	a. Program for Linear regression. b. Program for Polynomial Regression.	2
10	<b>Regression 2</b>	a. Program for multiple linear regression. b. Program for non-linear regression.	2

This is sample Practical list. Course instructor may change the practical as per syllabus.

**Learning Resources recommended:**

<https://www.scilab.org/about/community/books>

**Evaluation Pattern****A. Continuous Internal Evaluation (40 Marks):**

<b>Method</b>	<b>Marks</b>
Certified Journal (section I and II)	20
Attendance and active participation in both Laboratory	20

**B. Semester End Evaluation (Paper Pattern) (60 Marks – 4 hours):****Section I: (30 Marks - 2 Hours)**

<b>Question No</b>	<b>Unit</b>	<b>Marks</b>
<b>1</b>	<b>Program 1</b>	<b>10</b>
<b>2</b>	<b>Program 2</b>	<b>15</b>
<b>3</b>	<b>Viva</b>	<b>05</b>

**Section II:(30 Marks - 2 Hours)**

<b>Question No</b>	<b>Unit</b>	<b>Marks</b>
<b>1</b>	<b>Program 1</b>	<b>10</b>
<b>2</b>	<b>Program 2</b>	<b>15</b>
<b>3</b>	<b>Viva</b>	<b>05</b>

Name of the Course	<b>Computer Skills -2 Practical</b>
Course Code (refer to student handbook)	<b>USIT208</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>II</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	<b>Vocational Skill Course</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	Restructuring is done to demonstrate a basic understanding of Operating Systems and ability of web designing also to get understanding of advanced web designing technologies.



## Nomenclature: Computer Skills -2 Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Study different operating systems.
2. Analyze the working of an operating system, its programming interface and file system.
3. Develop algorithms for process scheduling, memory management, page replacement algorithms and disk scheduling
4. Use Bootstrap to create websites quickly.
5. Access, setup and use Bootstrap in web projects.
6. Create websites from scratch using Bootstrap.
7. Create responsive websites rapidly.

### Section I: Operating System Practical

#### Curriculum:

Sr. No.	Title	Learning Points	No of hrs
1	Installation 1	a. Installation of virtual machine software.	2
2	Installation 2	a. Installation of Linux operating system (RedHat / Ubuntu) on virtual machine.	2
3	Installation 3	a. Installation of Windows operating system on virtual machine.	2
4	Linux commands: Working with Directories	a. pwd, cd, absolute and relative paths, ls, mkdir, rmdir, b. file, touch, rm, cp, mv, rename, head, tail, cat, tac, more, less, strings, chmod	2+2
5	Linux commands: Working with files	a. ps, top, kill, pkill, bg, fg, b. grep, locate, find, locate. c. date, cal, uptime, w, whoami, finger, uname, man, df, du, free, whereis, which. d. Compression: tar, gzip.	2+2
6	Windows (DOS) Commands 1	a. Date, time, prompt, md, cd, rd, path. b. Chkdsk, copy, xcopy, format, fidsk, cls, defrag, del, move.	2+2
7	Windows (DOS) Commands 2	a. Diskcomp, diskcopy, diskpart, doskey, echo b. Edit, fc, find, rename, set, type, ver	2

<b>8</b>	<b>Working with Linux Desktop and utilities</b>	<ul style="list-style-type: none"> <li>a. The vi editor.</li> <li>b. Graphics</li> <li>c. Terminal</li> <li>d. Adjusting display resolution</li> <li>e. Using the browsers</li> <li>f. Configuring simple networking</li> <li>g. Creating users and shares</li> </ul>	2+2
<b>9</b>	<b>Programming with Linux</b>	<ul style="list-style-type: none"> <li>a. Installing utility software on Linux and Windows Running C/C++ Programs with command line argument in linux</li> </ul>	2+2
<b>10</b>	<b>Introduction to Shell Scripting</b>	<ul style="list-style-type: none"> <li>a. Basic Operators</li> <li>b. Decision Making</li> <li>c. Looping</li> </ul>	2

This is sample Practical list. Course instructor may change the practical as per syllabus.

## Section II: Responsive Web Designing Practical

### Curriculum:

Sr. No	Title	Learning Points	No of Hrs.
1	<b>Setting Up Bootstrap</b>	<ol style="list-style-type: none"><li>Install bootstrap framework and understand various tags.</li><li>Design webpage that shows department name, college name at center of web page by using bootstrap framework and without using bootstrap framework.</li></ol>	2
2	<b>Container Class</b>	<ol style="list-style-type: none"><li>Display student information content on responsive webpage by using container and container fluid classes.</li><li>Use offset column recording column and nesting column to create responsive web page for given format.</li></ol>	2+2
3	<b>Grid System</b>	<ol style="list-style-type: none"><li>Create a responsive web page of your class time table by using the bootstrap grid system.</li></ol>	2
4	<b>Table &amp; Events</b>	<ol style="list-style-type: none"><li>Show at least 3 to 4 co-curricular, extracurricular activities of students that includes responsive tables with style such as overstate when Mouse over different colors of each row table with strip row.</li></ol>	2
5	<b>Text-Decoration &amp; Form Layout</b>	<ol style="list-style-type: none"><li>Use Bootstrap typography to create responsive web pages on a given topic.</li><li>Design responsive web page for student registration form using bootstrap form layout form control bootstrap button</li></ol>	2+2
6	<b>Glyphicons &amp; Components</b>	<ol style="list-style-type: none"><li>Create various types of menus using bootstrap menu components such as right aligned drop down menu drop of menu adding headers of each item and glyphicons component.</li><li>Design responsive webpage that should show odd semesters and even semesters considered as a menu course of 8th semester as sub menu using button groups and button toolbar component.</li></ol>	2+2
7	<b>Bootstrap input groups components</b>	<ol style="list-style-type: none"><li>Used different bootstrap input groups components to create responsive web pages for job applications or any other kind of application.</li></ol>	2
8	<b>Different types of components</b>	<ol style="list-style-type: none"><li>Create responsive web pages of education websites using bootstrap breadcrumb, pagination, labels, JumboTron, page header, thumbnail component.</li></ol>	2

<b>9</b>	<b>Progress Bar &amp; Components</b>	a. Following task to be performed using bootstrap progress bars components i. creating progress bar ii. adding label to progress bar iii. creating multi color stripped and animated progress bars	2+2
<b>10</b>	<b>Carousel bootstrap plugin</b>	a. Design animated photo gallery page Using Carousel bootstrap plugin with minimum 7 photos	2+2

This is sample Practical list. Course instructor may change the practical as per syllabus.

### Evaluation Pattern

#### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Certified Journal (section I and II)	20
Attendance and active participation in both Laboratory	20

#### B. Semester End Evaluation (Paper Pattern) (60 Marks – 4 hours):

##### Section I: (30 Marks - 2 Hours)

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

##### Section II:(30 Marks - 2 Hours)

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

Name of the Course	<b>Technical Communication Skills II</b>
Course Code (refer to student handbook)	<b>USIT208</b>
Class	<b>F.Y.B.Sc.IT</b>
Semester	<b>II</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Ability Enhancement Course</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

### **Nomenclature: Technical Communication Skills II**

#### **Course Outcomes:**

On the successful completion of this course, the learner will be able to

1. Recognize the importance of various types of communication in technical set up.
2. Understand the dynamics in different forms of formal communication.
3. Learn about active listening and the art of giving presentations and interviews.
4. Learn the art of business writing and ethics in business communication across functional areas.
5. Evaluate, analyze and interpret technical data.

#### **Curriculum:**

<b>Unit</b>	<b>Title</b>	<b>Learning Points</b>	<b>No. of hrs</b>
<b>I</b>	<b>Effective presentation Strategies</b>	Introduction, Defining purpose, Analyzing audience and Locale, Organizing contents, preparing outline, Visual Aids, Understanding Nuances of delivery, Kinesics	10
	<b>Interview</b>	Introduction, objectives, types of interview, job interviews	
	<b>Business writing</b>	Introduction, Importance of written Business, Five main strategies of writing business messages	

<b>II</b>	<b>Business correspondence</b>	Business letter writing, common component of Business letter, Strategies for writing body of a letter, Types of Business letter, writing memos	10
	<b>Business reports and proposal</b>	What is a report? Steps in writing routine Business report, parts of report, corporate reports and Business proposals	
	<b>Careers and Resume</b>	Introduction to career building, resume format, traditional, electronic and video resumes, sending resume, follow up letters and online recruitment process	
<b>III</b>	<b>Communication across Functional areas</b>	Financial communication, MIS	10
	<b>Ethics in Business Communication</b>	Ethical communication, Values, ethics and communication, ethical dilemmas facing manager, strategic approaches to corporate ethics	
	<b>Creating and Using Visual Aids</b>	Object, Models, Handouts, Charts and Graphs, Text Visuals , Formatting Computer generated charts, graphs and visuals	

#### Learning Resources recommended:

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1	Business Communication	Meenakshi Raman & Prakash Singh	Oxford-Higher Education	2nd edition	2006
2	Professional Communication	Aruna Koneru	McGraw Hill		2008
3	Professional Communication Skills	Laila Dias	Vipul Prakashan	1st edition	2010

## Evaluation Pattern:

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks)	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	15
2	II	15
3	III	15
4	I II III	05 05 05

#### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numerical based questions, single line answers, etc., will contain internal options.
  3. Refer **appendix I** for Paper pattern.
-

# APPENDIX I

## PAPER PATTERN

### 1. Internal Evaluation

Test: 1 Class test of 20 marks. (Can be taken online/offline)

<b>Q</b>	<b>Attempt <i>any four</i> of the following:</b>	<b>20</b>
<b>a.</b>		
<b>b.</b>		
<b>c.</b>		
<b>d.</b>		
<b>e.</b>		
<b>f.</b>		

### 2. External Examination: (60 marks) (Currently for SY and TY)

	<b>All questions are compulsory</b>	
<b>Q1</b>	<b>(Based on Unit 1) Attempt <i>any three</i> of the following:</b>	<b>12</b>
<b>a.</b>		
<b>b.</b>		
<b>c.</b>		
<b>d.</b>		
<b>e.</b>		
<b>f.</b>		
<b>Q2</b>	<b>(Based on Unit 2) Attempt <i>any three</i> of the following:</b>	<b>12</b>
<b>Q3</b>	<b>(Based on Unit 3) Attempt <i>any three</i> of the following:</b>	<b>12</b>
<b>Q4</b>	<b>(Based on Unit 4) Attempt <i>any three</i> of the following:</b>	<b>12</b>
<b>Q5</b>	<b>(Based on Unit 5) Attempt <i>any three</i> of the following:</b>	<b>12</b>



**3. External Examination: (60 marks) (For FY)**

	<b>All questions are compulsory</b>	
<b>Q1</b>	<b>(Based on Unit 1) Attempt <i>any three</i> of the following:</b>	<b>15</b>
<b>a.</b>		
<b>b.</b>		
<b>c.</b>		
<b>d.</b>		
<b>e.</b>		
<b>f.</b>		
<b>Q2</b>	<b>(Based on Unit 2) Attempt <i>any three</i> of the following:</b>	<b>15</b>
<b>Q3</b>	<b>(Based on Unit 3) Attempt <i>any three</i> of the following:</b>	<b>15</b>
<b>Q4</b>	<b>(Based on Units 1,2 &amp; 3) Attempt <i>any three</i> of the following:</b>	<b>15</b>
	<b>(Set 2 Sub questions on each unit)</b>	

# APPENDIX – II

## Programming Practical 1 Semester I And Programming Practical 2 Semester II

### Section II: Application Development using C/C++/Python/Java

The project documentation must be submitted in the following format.

#### Project Documentation

- **Basic requirement:**
  - Font size:12
  - Font Style: Arial
- **Documentation points:**
  1. Problem Statement
  2. Aim
  3. Overall Idea (About Project) & Objectives
  4. Description Of Components (used in project)
  5. Program
  6. Screenshot of Output

R. E. Society's  
**R. P. Gogate College of Arts & Science And**  
**R. V. Jogalekar College of Commerce (Autonomous), Ratnagiri**  
**Department of Information Technology**

**Open Elective Courses**

<b>SEMESTER I</b>				
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs.</b>
USOE103	<b>Google Workspace and Multimedia Apps</b>	Tutorial+Practical	1+1	1+2

Name of the Course	<b>Google Workspace and Multimedia Apps</b>
Course Code (refer to student handbook)	<b>USOE103</b>
Class	<b>FYBSc/FYBCom/FYBA</b>
Semester	<b>I</b>
No of Credits	1+1
Nature	Tutorial+Practical
Type (applicable to NEP only)	<b>Open Elective( Offered by Department)</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Google Workspace and Multimedia Apps

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Simplify basic office tasks and improve work productivity.
2. To be able to create documents for printing and sharing
3. To be able to manage google forms.
4. To be able to manage and store data in a spreadsheet.
5. Develop a comprehensive understanding of multimedia, encompassing its core components, diverse applications across industries, and a critical analysis of its advantages and disadvantages.
6. Attain proficiency in utilizing essential multimedia tools like Canva for graphic design, Lexis Audio Editor for audio editing, and VN Mobile application for video editing. Gain practical experience in creating, editing, and sharing multimedia content across various platforms.

### Curriculum:

Sr.No.	Title	Learning Points	No of hrs
1	Gmail	<ol style="list-style-type: none"><li>1. Send and receive emails.</li><li>2. Organize your inbox with labels and filters.</li></ol>	2
2	Google Calendar	<ol style="list-style-type: none"><li>1. Schedule appointments and meetings.</li><li>2. Set up event reminders.</li></ol>	2
3	Google Drive	<ol style="list-style-type: none"><li>1. Store and organize files in the cloud.</li><li>2. Collaborate on documents, spreadsheets, and presentations in real-time.</li></ol>	2+2
4	Google Docs	<ol style="list-style-type: none"><li>1. Create and edit documents.</li><li>2. Collaborate with others in real-time.</li></ol>	2+2
5	Google Sheets	<ol style="list-style-type: none"><li>1. Create and manage spreadsheets.</li><li>2. Perform data analysis and calculations.</li><li>3. Share and collaborate on data sheets.</li><li>4. Create charts and graphs.</li></ol>	2+2
6	Google Slides	<ol style="list-style-type: none"><li>1. Create and design presentations</li></ol>	2+2
7	Google Forms	<ol style="list-style-type: none"><li>1. Design surveys and questionnaires.</li><li>2. Collect responses and analyze results.</li><li>3. Customize forms with various question types.</li><li>4. Share forms via a link or embed them on websites</li></ol>	2
8	Lexis Audio Editor	<ol style="list-style-type: none"><li>1. Record an audio file and Save it.</li><li>2. Export Part of file.</li><li>3. Import and Mix: - Define a selection, Copy a part</li></ol>	2+2

		of a song from one to another file 4. Trim a sound file, Append or insert another audio file 5. Record direct into an open sound file	
9	CANVA tool	1. Design a Flyer using Canva.	2
10	VN Mobile	1. Record video for Instagram reels/ YouTube shorts using VN Editor	2

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Lexis Audio Editor	<a href="https://www.lexisaudioeditor.com/tutorial-overview-android/">https://www.lexisaudioeditor.com/tutorial-overview-android/</a>			
2	Canva tool	<a href="https://www.canva.com/">https://www.canva.com/</a>			
3	Google	<a href="https://workspace.google.com/intl/en_in/training/">https://workspace.google.com/intl/en_in/training/</a>			

### Teaching plan:

Sr.No.	Title	Expected date of completion	Teaching methods
1	Gmail	Fourth week of July	Practical
2	Google Calendar	First week of August	Practical
3	Google Drive	Second week of August	Practical
4	Google Docs	Third week of August	Practical
5	Google Sheets	Fourth week of August	Practical
6	Google Slides	First week of September	Practical
7	Google Forms	Second week of September	Practical
8	Lexis Audio Editor	Fourth week of September	Practical
9	CANVA tool	First week of October	Practical
10	VN Mobile	Second week of October	Practical

## Evaluation Pattern

### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

---

<b>SEMESTER II</b>				
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>Credits</b>	<b>No. of Hrs.</b>
<b>USOE210</b>	Social Media Awareness	Tutorial+Practical	1+1	1+2

Name of the Course	<b>Social Media Awareness</b>
Course Code (refer to student handbook)	<b>USIT10</b>
Class	
Semester	<b>II</b>
No of Credits	1+1
Nature	Tutorial+Practical
Type (applicable to NEP only)	<b>Open Elective( Offered by Department)</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Social Media Awareness

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Aware about Social media usage.
2. Take precautions while using social media.
3. Make email and social media accounts secure.
4. Make android phones secure.

### Curriculum:

Sr.No.	Title	Learning Points	No of hrs
1	Mail accounts Security 1	<ol style="list-style-type: none"><li>1. Change your password</li><li>2. Recover your account</li></ol>	2
2	Mail accounts Security 2	<ol style="list-style-type: none"><li>1. Update your account recovery details</li><li>2. Enable multi-factor authentication</li></ol>	2
3	Mail accounts Security 3	<ol style="list-style-type: none"><li>1. Check account mail settings</li><li>2. Check third party application access</li><li>3. Check login activity</li><li>4. Sign out of all other sessions</li></ol>	2+2
4	Facebook	<ol style="list-style-type: none"><li>1. Facebook password configuration</li><li>2. How to set contact information</li><li>3. Two factor authentication</li><li>4. Security and privacy features of facebook</li><li>5. How to be safe from scams and frauds</li></ol>	2+2
5	Instagram	<ol style="list-style-type: none"><li>1. Instagram password configuration</li><li>2. How to set personal details</li><li>3. Two factor authentication</li><li>4. Security and privacy features of instagram</li></ol>	2+2
6	WhatsApp	<ol style="list-style-type: none"><li>1. Two step Verification</li><li>2. Security and privacy features of WhatsApp</li><li>3. How to safe from frauds and unidentified numbers</li></ol>	2
7	Snapchat	<ol style="list-style-type: none"><li>4. Password Configuration</li><li>5. Contact Details</li><li>6. Two factor authentication</li><li>7. Privacy features of snapchat.</li></ol>	2
8	Android Phone	<ol style="list-style-type: none"><li>1. Set a strong passcode ( consider disabling fingerprint or face login )</li><li>2. Audit app permissions</li><li>3. Enable automatic updates</li></ol>	2+2



		4. Enable find my Device 5. Keep Sensitive notifications of the lock screen	
9	Android Phone	1. Disable personalized ads 2. Give your google account a privacy check-up 3. Quickly block access to the camera or mic 4. Keep an eye on your clipboard 5. Use end to end encryption in messages	2+2
10	Awareness	1. How to be safe from scams and frauds	2

### Evaluation Pattern

#### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

#### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Practical 1	10
2	Practical 2	15
3	Viva	05

---

## Scheme of Courses for S.Y.B.Sc. I.T.

<b>Semester – III</b>				
<b>Course Code</b>	<b>Course Type</b>	<b>Course Title</b>	<b>Credits</b>	<b>No of lects/ week</b>
USIT301	Skill Enhancement Course	Python Programming	2	5
USIT302	Core Subject	Data Structures	2	5
USIT303	Core Subject	Computer Networks	2	5
USIT304	Core Subject	Operating Systems	2	5
USIT305	Core Subject	Applied Mathematics	2	5
USIT3P1	Skill Enhancement Course Practical	Python Programming Practical	2	3
USIT3P2	Core Subject Practical	Data Structures Practical	2	3
USIT3P3	Core Subject Practical	Computer Networks Practical	2	3
USIT3P4	Core Subject Practical	Operating Systems Practical	2	3
USIT3P5	Core Subject Practical	Mobile Programming Practical	2	3
<b>Total Credits</b>			<b>20</b>	

<b>Semester – IV</b>				
<b>Course Code</b>	<b>Course Type</b>	<b>Course Title</b>	<b>Credits</b>	<b>No of lects/ week</b>
USIT401	Skill Enhancement Course	Core Java	2	5
USIT402	Core Subject	Introduction to Embedded Systems	2	5
USIT403	Core Subject	Computer Oriented Statistical Techniques	2	5
USIT404	Core Subject	Software Engineering	2	5
USIT405	Core Subject	Computer Graphics and Animation	2	5
USIT4P1	Skill Enhancement Course Practical	Core Java Practical	2	3
USIT4P2	Core Subject Practical	Introduction to Embedded Systems Practical	2	3
USIT4P3	Core Subject Practical	Computer Oriented Statistical Techniques Practical	2	3
USIT4P4	Core Subject Practical	Software Engineering Practical	2	3
USIT4P5	Core Subject Practical	Computer Graphics and Animation Practical	2	3
<b>Total Credits</b>			<b>20</b>	

# SEMESTER III

## Syllabus for S. Y. B. Sc. I.T. Autonomous for the year 2023-24

Name of the Course	<b>Python Programming</b>
Course Code (refer to student handbook)	<b>USIT301</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>III</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	In the Python Programming course, the emphasis is on fostering employability, entrepreneurship, and skill development. Through hands-on projects and real-world applications, students gain practical programming expertise sought after by the job market. They learn to develop software solutions, enhancing their employability prospects in various industries. Additionally, the course encourages innovative thinking and problem-solving, nurturing an entrepreneurial mindset among students. They acquire coding proficiency, making them well-equipped to tackle challenges and explore entrepreneurial ventures in the technology domain. By honing their Python programming skills, students become valuable assets to potential employers and are empowered to create their own opportunities in the ever-evolving world of IT and business.

## Nomenclature: Python Programming

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Aware of the variables, expressions, looping and conditions used in Python programming.
2. Implement functions, strings, lists, tuples and directories
3. Create GUI forms and add widgets.
4. Use MySQL to store data.
5. Apply the programming skill set learnt here into various domains by having advance programming skill set of Python and usage of libraries.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
<b>I</b>	<b>Introduction</b>	The Python Programming Language, History, features, Installing Python, Running	<b>12</b>
	<b>Python program, Debugging</b>	Syntax Errors, Runtime Errors, Semantic Errors, Experimental Debugging, Formal and Natural Languages, The Difference Between Brackets, Braces, and Parentheses	
	<b>Variables and Expressions</b>	Values and Types, Variables, Variable Names and Keywords, Type conversion, Operators and Operands, Expressions, Interactive Mode and Script Mode, Order of Operations.	
	<b>Conditional Statements</b>	if, if-else, nested if –else	
	<b>Looping</b>	for, while, nested loops Control statements: Terminating loops, skipping specific conditions	
<b>II</b>	<b>Functions</b>	Function Calls, Type Conversion Functions, Math Functions, Composition, Adding New Functions, Definitions and Uses, Flow of Execution, Parameters and Arguments, Variables and Parameters Are Local, Stack Diagrams, Fruitful Functions and Void Functions, Why Functions? Importing with from, Return Values, Incremental Development, Composition, Boolean	<b>12</b>

		Functions, More Recursion, Leap of Faith, Checking Types	
	<b>Strings</b>	A String Is a Sequence, Traversal with a for Loop, String Slices, Strings Are Immutable, Searching, Looping and Counting, String Methods, The in Operator, String Comparison, String Operations.	
<b>III</b>	<b>Lists</b>	Values and Accessing Elements, Lists are mutable, traversing a List, Deleting elements from List, Built-in List Operators, Concatenation, Repetition, In Operator, Built-in List functions and methods	<b>12</b>
	<b>Tuples and Dictionaries</b>	Tuples, Accessing values in Tuples, Tuple Assignment, Tuples as return values, Variable-length argument tuples, Basic tuples operations, Concatenation, Repetition, in Operator, Iteration, Built-in Tuple Functions Creating a Dictionary, Accessing Values in a dictionary, Updating Dictionary, Deleting Elements from Dictionary, Properties of Dictionary keys, Operations in Dictionary, Built-In Dictionary Functions, Built-in Dictionary Methods	
	<b>Files</b>	Text Files, The File Object Attributes, Directories	
	<b>Exceptions</b>	Built-in Exceptions, Handling Exceptions, Exception with Arguments, User-defined Exceptions	
<b>IV</b>	<b>Regular Expressions</b>	Concept of regular expression, various types of regular expressions, using match function.	<b>12</b>
	<b>Classes and Objects</b>	Overview of OOP (Object Oriented Programming), Class Definition, Creating Objects, Instances as Arguments, Instances as return values, Built-in Class Attributes, Inheritance, Method Overriding, Data Encapsulation, Data Hiding	
	<b>Multithreaded Programming</b>	Thread Module, creating a thread, synchronizing threads, multithreaded priority queue	

	<b>Modules</b>	Importing module, Creating and exploring modules, Math module, Random module, Time module	
<b>V</b>	<b>Creating the GUI Form and Adding Widgets: Widgets</b>	Widgets: Button, Canvas, Checkbutton, Entry, Frame, Label, Listbox, Menubutton, Menu, Message, Radiobutton, Scale, Scrollbar, text, Toplevel, Spinbox, PanedWindow, LabelFrame, tkMessageBox. Handling Standard attributes and Properties of Widgets.	<b>12</b>
	<b>Layout Management</b>	Designing GUI applications with proper Layout Management features.	
	<b>Look and Feel Customization</b>	Enhancing Look and Feel of GUI using different appearances of widgets, Storing Data in Our MySQL	
	<b>Database via Our GUI</b>	Connecting to a MySQL database from Python, Configuring the MySQL connection, Designing the Python GUI database, Using the INSERT command, Using the UPDATE command, Using the DELETE command, Storing and retrieving data from MySQL database.	



### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Think Python	Allen Downey	O'Reilly	1st	2012
2	An Introduction to Computer Science using Python 3	Jason Montojo, Jennifer Campbell, Paul Gries	SPD	1st	2014
3	Python GUI Programming Cookbook	Burkhard A. Meier	Packt		2015
4	Introduction to Problem Solving with Python	E. Balagurusamy	TMH	1st	2016
5	Murach's Python programming	Joel Murach, Michael Urban	SPD	1st	2017
6	Object-oriented Programming in Python	Michael H. Goldwasser, David Letscher	Pearson Prentice Hall	1st	2008
7	Exploring Python	Budd	TMH	1st	2016

### Evaluation Pattern

#### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks)	20
Assignments	10
Attendance and active participation in classroom	10

**B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):**

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

**Guidelines for paper pattern for Semester End Evaluation:**

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
  3. Refer **appendix I** for Paper pattern.
-

Name of the Course	<b>Python Programming Practical</b>
Course Code (refer to student handbook)	<b>USIT3P1</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>III</b>
No of Credits	<b>2</b>
Nature	Practical
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Python Programming Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Demonstrate proficiency in Python programming concepts, syntax, and standard libraries through hands-on coding exercises and projects.(Proficiency in Python )
2. Apply Python programming skills to solve real-world problems, creating practical solutions that can be implemented in various domains.(Real-World Application)
3. Develop problem-solving and critical thinking skills by tackling challenging programming tasks and overcoming coding obstacles.(Problem-Solving Abilities)
4. Gain valuable hands-on experience through practical assignments, enhancing employability and making students more desirable to potential employers.(Practical Experience )
5. Foster an entrepreneurial mindset by encouraging creative thinking and innovation, empowering students to explore entrepreneurial ventures in the technology landscape.( Entrepreneurial Mindset)

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 1	Write the program for the following: a. Create a program that asks the user to enter their name and their age. Print out a message addressed to them that tells them the year that they will turn 100 years old. b. Enter the number from the user and depending on whether the number is even or odd, print out an appropriate message to the user. c. Write a program to generate the Fibonacci series. d. Write a function that reverses the user defined value. e. Write a function to check the input value is Armstrong and also write the function for Palindrome. f. Write a recursive function to print the factorial for a given number.	3
2	Practical 2	Write the program for the following: a. Write a function that takes a character (i.e. a string of length 1) and returns True if it is a vowel, False otherwise.	3

		<p>b. Define a function that computes the length of a given list or string.</p> <p>c. Define a procedure histogram() that takes a list of integers and prints a histogram to the screen. For example, histogram([4, 9, 7]) should print the following:</p> <pre>**** ***** *****</pre>	
3	<b>Practical 3</b>	<p>Write the program for the following:</p> <p>a. A pangram is a sentence that contains all the letters of the English alphabet at least once, for example: The quick brown fox jumps over the lazy dog. Your task here is to write a function to check a sentence to see if it is a pangram or not.</p> <p>b. Take a list, say for example this one: a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89] and write a program that prints out all the elements of the list that are less than 5.</p>	3
4	<b>Practical 4</b>	<p>Write the program for the following:</p> <p>a. Write a program that takes two lists and returns True if they have at least one common member.</p> <p>b. Write a Python program to print a specified list after removing the 0th, 2nd, 4<sup>th</sup> and 5th elements.</p> <p>c. Write a Python program to clone or copy a list</p>	3
5	<b>Practical 5</b>	<p>Write the program for the following:</p> <p>a. Write a Python script to sort (ascending and descending) a dictionary by value.</p> <p>b. Write a Python script to concatenate the following dictionaries to create a new one. Sample Dictionary : dic1={1:10, 2:20} dic2={3:30, 4:40} dic3={5:50,6:60} Expected Result : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}</p> <p>c. Write a Python program to sum all the items in a dictionary.</p>	3
6	<b>Practical 6</b>	<p>Write the program for the following:</p> <p>a. Write a Python program to read an entire text file.</p> <p>b. Write a Python program to append text to a file and display the text.</p> <p>c. Write a Python program to read last n lines of a file.</p>	3

7	<b>Practical 7</b>	<p>Write the program for the following:</p> <ol style="list-style-type: none"> <li>Design a class that store the information of student and display the same</li> <li>Implement the concept of inheritance using python</li> <li>Create a class called Numbers, which has a single class attribute called MULTIPLIER, and a constructor which takes the parameters x and y (these should all be numbers). <ol style="list-style-type: none"> <li>Write a method called add which returns the sum of the attributes x and y.</li> <li>Write a class method called multiply, which takes a single number parameter a and returns the product of a and MULTIPLIER.</li> <li>Write a static method called subtract, which takes two number parameters, b and c, and returns b - c.</li> <li>Write a method called value which returns a tuple containing the values of x and y. Make this method into a property, and write a setter and a deleter for manipulating the values of x and y.</li> </ol> </li> </ol>	3
8	<b>Practical 8</b>	<p>Write the program for the following:</p> <ol style="list-style-type: none"> <li>Open a new file in IDLE (“New Window” in the “File” menu) and save it as geometry.py in the directory where you keep the files you create for this course. Then copy the functions you wrote for calculating volumes and areas in the “Control Flow and Functions” exercise into this file and save it. Now open a new file and save it in the same directory. You should now be able to import your own module like this:import geometry Try and add print dir(geometry) to the file and run it. Now write a function pointyShapeVolume(x, y, squareBase) that calculates the volume of a square pyramid if squareBase is True and of a right circular cone if squareBase is False. x is the length of an edge on a square if squareBase is True and the radius of a circle when squareBase is False. y is the height of the object. First use squareBase to distinguish the cases. Use the circleArea and squareArea from the geometry module to calculate the base areas.</li> <li>Write a program to implement exception handling.</li> </ol>	3
9	<b>Practical 9</b>	<p>Write the program for the following:</p> <ol style="list-style-type: none"> <li>Try to configure the widget with various options like: bg=“red”, family=“times”,size=18</li> <li>Try to change the widget type and configuration options to experiment with other widget types like</li> </ol>	3

		Message, Button, Entry, Checkbutton, Radiobutton, Scale etc.	
<b>10</b>	<b>Practical 10</b>	Design the database applications for the following: a. Design a simple database application that stores the records and retrieve the same. b. Design a database application to search the specified record from the database. c. Design a database application to that allows the user to add, delete and modify the records.	<b>3</b>

This is sample Practical list. Course instructor may change the practical as per syllabus.

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Think Python	Allen Downey	O'Reilly	1st	2012
2	An Introduction to Computer Science using Python 3	Jason Montojo, Jennifer Campbell, Paul Gries	SPD	1st	2014

### Evaluation Pattern

#### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

#### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
<b>1</b>	<b>Program 1</b>	<b>10</b>
<b>2</b>	<b>Program 2</b>	<b>15</b>
<b>3</b>	<b>Viva</b>	<b>05</b>

Name of the Course	<b>Data Structures</b>
Course Code (refer to student handbook)	<b>USIT302</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>III</b>
No of Credits	<b>2</b>
Nature	Theory
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	In the Data Structures course, the focus is on enhancing employability, fostering entrepreneurship, and promoting skill development. Through a hands-on approach and practical implementations, students develop proficiency in various data structures, making them highly desirable to potential employers seeking skilled programmers and software developers. The course emphasizes problem-solving and algorithmic thinking, nurturing an entrepreneurial mindset among learners. By mastering data structures and their applications, students are equipped to tackle complex challenges in diverse industries, enhancing their employability prospects. Additionally, the acquired skills empower students to explore innovative solutions and venture into entrepreneurial endeavors, contributing to the technology-driven business landscape. Overall, this Data Structures course prepares students for success in their careers and entrepreneurial pursuits.



## Nomenclature: Data Structures

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Identify and distinguish data structure classification, data types, their complexities
2. Understand array, linked list, stack and queue.
3. Understand trees, various hashing techniques and graph for various applications
4. Compare various sorting and searching techniques.
5. Choose appropriate algorithm design techniques for solving problems.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	<b>Introduction</b>	Data and Information, Data Structure, Classification of Data Structures, Primitive Data Types, Abstract Data Types, Data structure vs. File Organization, Operations on Data Structure, Algorithm, Importance of Algorithm Analysis, Complexity of an Algorithm, Asymptotic Analysis and Notations, Big O Notation, Big Omega Notation, Big Theta Notation, Rate of Growth and Big O Notation.	12
	<b>Array</b>	Introduction, One Dimensional Array, Memory Representation of One Dimensional Array, Traversing, Insertion, Deletion, Searching, Sorting, Merging of Arrays, Multidimensional Arrays, Memory Representation of Two Dimensional Arrays, General Multi-Dimensional Arrays, Sparse Arrays, Sparse Matrix, Memory Representation of Special kind of Matrices, Advantages and Limitations of Arrays.	

<b>II</b>	<b>Linked List</b>	Linked List, One-way Linked List, Traversal of Linked List, Searching, Memory Allocation and Deallocation, Insertion in Linked List, Deletion from Linked List, Copying a List into Another List, Merging Two Linked Lists, Splitting a List into Two Lists, Reversing One way linked List, Circular Linked List, Applications of Circular Linked List, Two way Linked List, Traversing a Two way Linked List, Searching in a Two way linked List, Insertion of an element in Two way Linked List, Deleting a node from Two way Linked List, Header Linked List, Applications of the Linked list, Representation of Polynomials, Storage of Sparse Arrays, Implementing other Data Structures.	<b>12</b>
<b>III</b>	<b>Stack</b>	Introduction, Operations on the Stack Memory Representation of Stack, Array Representation of Stack, Applications of Stack, Evaluation of Arithmetic Expression, Matching Parenthesis, infix and postfix operations, Recursion.	<b>12</b>
	<b>Queue</b>	Introduction, Queue, Operations on the Queue, Memory Representation of Queue, Array representation of queue, Linked List Representation of Queue, Circular Queue, Some special kinds of queues, Dequeue, Priority Queue, Application of Priority Queue, Applications of Queues.	
<b>IV</b>	<b>Sorting and Searching Techniques</b>	Bubble, Selection, Insertion, Merge Sort. Searching: Sequential, Binary, Indexed Sequential Searches, Binary Search.	<b>12</b>
	<b>Tree</b>	Tree, Binary Tree, Properties of Binary Tree, Memory Representation of Binary Tree, Operations Performed on Binary Tree, Reconstruction of Binary Tree from its Traversals, Huffman Algorithm, Binary Search Tree, Operations on Binary Search Tree, Heap, Memory Representation of Heap, Operation on Heap, Heap Sort.	
	<b>Advanced Tree Structures</b>	Red Black Tree, Operations Performed on Red Black Tree, AVL Tree, Operations performed on AVL Tree, 2-3 Tree, B-Tree.	

<b>V</b>	<b>Hashing Techniques</b>	Hash function, Address calculation techniques, Common hashing functions Collision resolution, Linear probing, Quadratic, Double hashing, Bucket hashing, Deletion and rehashing	<b>12</b>
	<b>Graph</b>	Introduction, Graph, Graph Terminology, Memory Representation of Graph, Adjacency Matrix Representation of Graph, Adjacency List or Linked Representation of Graph, Operations Performed on Graph, Graph Traversal, Applications of the Graph, Reachability, Shortest Path Problems, Spanning Trees.	

**Learning Resources recommended:**

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1	A Simplified Approach to Data Structures	Lalit Goyal, Vishal Goyal, Pawan Kumar	SPD	1st	2014
2	An Introduction to Data Structure with Applications	Jean – Paul Tremblay and Paul Sorenson	Tata MacGraw Hill	2nd	2007
3	Data Structure and Algorithm	Maria Rukadikar	SPD	1st	2017
4	Schaum’s Outlines Data structure	Seymour Lipschutz	Tata McGraw Hill	2nd	2005
5	Data structure – A Pseudocode Approach with C	AM Tanenbaum, Y Langsam and MJ Augustein	Prentice Hall India	2nd	2006
6	Data structure and Algorithm Analysis in C	Weiss, Mark Allen	Addison Wesley	1st	2006

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
3. Refer **appendix I** for Paper pattern.

Name of the Course	<b>Data Structures Practical</b>
Course Code (refer to student handbook)	<b>USIT3P2</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>III</b>
No of Credits	<b>2</b>
Nature	Practical
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Data Structures Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Implement fundamental data structures efficiently.
2. Analyze and compare data structures for performance.
3. Solve real-world problems using data structures and algorithms.
4. Optimize code and troubleshoot errors in implementations.

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	<b>Programs using Arrays</b>	1. Implement the following: a. Write a program to store the elements in 1-D array and perform the operations like searching, sorting and reversing the elements. [Menu Driven] b. Read the two arrays from the user and merge them and display the elements in sorted order.[Menu Driven] c. Write a program to perform the Matrix addition, Multiplication and Transpose Operation. [Menu Driven]	3
2	<b>Programs on Linked List</b>	Implement the following for Linked List: a. Write a program to create a singly linked list and display the node elements in reverse order. b. Write a program to search the elements in the linked list and display the same c. Write a program to create a doubly linked list and sort the elements in the linked list.	3
3	<b>Programs on Stack</b>	3. Implement the following for Stack: a. Write a program to implement the concept of Stack with Push, Pop, Display and Exit operations. b. Write a program to convert an infix expression to postfix and prefix conversion. c. Write a program to implement the Tower of Hanoi problem.	3
4	<b>Programs on Queue</b>	Implement the following for Queue: a. Write a program to implement the concept of Queue with Insert, Delete, Display and Exit operations.	3

		b. Write a program to implement the concept of Circular Queue c. Write a program to implement the concept of Deque.	
5	<b>Programs on Sorting</b>	Implement the following sorting techniques: a. Write a program to implement bubble sort. b. Write a program to implement selection sort. c. Write a program to implement insertion sort.	3
6	<b>Programs on Searching Techniques</b>	Implement the following data structure techniques: a. Write a program to implement merge sort. b. Write a program to search the element using sequential search. c. Write a program to search the element using binary search.	3
7	<b>Programs on Tree</b>	Implement the following data structure techniques: a. Write a program to create the tree and display the elements. b. Write a program to construct the binary tree. c. Write a program for inorder, postorder and preorder traversal of tree	3
8	<b>Programs on Advanced Tree Structures</b>	Implement the following data structure techniques: a. Write a program to insert the element into maximum heap. b. Write a program to insert the element into minimum heap.	3
9	<b>Programs on Hashing Techniques</b>	Implement the following data structure techniques: a. Write a program to implement the collision technique. b. Write a program to implement the concept of linear probing.	3
10	<b>Programs on Graph</b>	Implement the following data structure techniques: a. Write a program to generate the adjacency matrix. b. Write a program for the shortest path diagram.	3

**Note :** 1. Learners may opt for any preferable languages from C/C++/Python/Java for implementation of practical programs.

2. This is sample Practical list. Course instructor may change the practical as per syllabus.

**Learning Resources recommended:**

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1	Data Structures and Algorithms Using Python	Rance Necaie	Wiley	First	2016
2	Data Structures Using C and C++	Langsam,Augenstein Tanenbaum	Pearson	First	2015

**Evaluation Pattern****A. Continuous Internal Evaluation (20 Marks):**

<b>Method</b>	<b>Marks</b>
Certified Journal	10
Attendance and active participation in Laboratory	10

**B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):**

<b>Question No</b>	<b>Unit</b>	<b>Marks</b>
1	Program 1	10
2	Program 2	15
3	Viva	05

---



Name of the Course	<b>Computer Networks</b>
Course Code (refer to student handbook)	<b>USIT303</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>III</b>
No of Credits	<b>2</b>
Nature	Theory
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Computer Networks

### Course Outcomes:

On the successful completion of this course, the learner will be able to:

1. Identify various data communication standards, topologies and terminologies
2. Describe how signals are used to transfer data and communication aspects between nodes.
3. Configure IP addresses using TCP/IP protocol suite
4. Understand the Transport layer and Network layer protocols
5. Use different application layer protocols.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
1	<b>Introduction</b>	Computer Network, Evolution of Computer Networks Different types of Computer Network, Difference between LAN, MAN and WAN, Hardware Devices used for Networking: Network Interface Card(NIC), Modem, Hub, Switch L1 and L2 switches, Comparison between switch and hub, Bridge, Router, Gateway. Standards and administration. Network Models: Protocol layering, TCP/IP protocol suite, The OSI model.	12
2	<b>Introduction to Physical layer</b>	Data and signals, periodic analog signals, digital signals, transmission impairment, data rate limits, performance.	12
	<b>Introduction to the Data Link Layer</b>	Link layer addressing, Data Link Layer Design Issues, Error detection and correction, block coding	
	<b>Wireless LANs</b>	Introduction, IEEE 802.11 project, Bluetooth, WiMAX, Cellular telephony, Satellite networks.	
3	<b>Network Layer</b>	IPv4 Addresses, IPv4 Protocol, ARP, ICMP, IPv6	12
	<b>Routing</b>	RIP, OSPF, BGP	
4	<b>Transport Layer</b>	UDP, TCP	12
5	<b>Application Layer</b>	WWW, HTTP, DNS, SMTP, POP3, MIME, IMAP, DHCP, TELNET, SSH, FTP	12

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	TCP/IP Protocol Suite	Behrouz A. Forouzan	Tata McGraw Hill	Fourth Edition	2010
2.	Data Communication and Networking	Behrouz A. Forouzan	Tata McGraw Hill	-----	-----
3.	Computer Networks	Andrew Tanenbaum	Pearson	Fifth	2013

### Evaluation Pattern

#### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks)	20
Assignments	10
Attendance and active participation in classroom	10

#### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

#### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
3. Refer **appendix I** for Paper pattern.

Name of the Course	<b>Computer Networks Practical</b>
Course Code (refer to student handbook)	<b>USIT3P3</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>III</b>
No of Credits	<b>2</b>
Nature	Theory
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Computer Networks Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Configure and manage network devices.
2. Implement and analyze network protocols.
3. Design and simulate network topologies.
4. Ensure network security and troubleshoot issues.
5. Collaborate on network projects and apply knowledge in real-world scenarios.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
1	Practical 1	Colour code for crimping LAN (Cat5/6/7) cable <ol style="list-style-type: none"><li>a. Study of Different color codes</li><li>b. Study of different connecting devices and their differences</li><li>c. Crimping LAN Cable</li></ol>	3
2	Practical 2	Configuring LAN setup <ol style="list-style-type: none"><li>a. Planning and Setting IP networks</li><li>b. Configuring subnet</li><li>c. Study of basic network command and Network configuration commands. ipconfig, netstat, ARP, ping, trace route etc.</li><li>d. Basic network troubleshooting.</li><li>e. Configuration of TCP/IP Protocols in Windows / Linux.</li><li>f. Implementation of Drive/file sharing and printer sharing.</li></ol>	3
3	Practical 3	IPv4 Addressing and Subnetting <ol style="list-style-type: none"><li>a. Given an IP address and network mask, determine other information about the IP address such as:<ul style="list-style-type: none"><li>● Network address</li><li>● Network broadcast address</li><li>● Total number of host bits</li><li>● Number of hosts</li></ul></li><li>b. Given an IP address and network mask, determine other information about the IP address such as:</li></ol>	3

		<ul style="list-style-type: none"> <li>• The subnet address of this subnet</li> <li>• The broadcast address of this subnet</li> <li>• The range of host addresses for this subnet</li> <li>• The maximum number of subnets for this subnet mask</li> <li>• The number of hosts for each subnet</li> <li>• The number of subnet bits</li> <li>• The number of this subnet</li> </ul>	
4	<b>Practical 4</b>	Designing and configuring a network topology a. Configure IP static routing	3
5	<b>Practical 5</b>	Configure IP routing using RIP.	3
6	<b>Practical 6</b>	Configuring Simple and multi-area OSPF.	3
7	<b>Practical 7</b>	Configuring server and client. a. Configure DHCP b. Configure DNS c. Configure HTTP d. Configure Telnet e. Configure FTP	3
8	<b>Practical 8</b>	Configure basic security features for networks	3
9	<b>Practical 9</b>	Packet capture and header analysis by wire-shark (TCP, UDP, IP etc.)	3
10	<b>Practical 10</b>	Planning and Design a corporate network for a given scenario.	3

This is sample Practical list. Course instructor may change the practical as per syllabus.

#### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	TCP/IP Protocol Suite	Behrouz A. Forouzan	Tata McGraw Hill	Fourth Edition	2010
2	Data Communication and Networking	Behrouz A. Forouzan	Tata McGraw Hill	-----	-----
3	<a href="https://www.cisco.com">https://www.cisco.com</a>				
4	<a href="https://www.computernetworkingnotes.com">https://www.computernetworkingnotes.com</a>				

## Evaluation Pattern

### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

---

Name of the Course	<b>Operating Systems</b>
Course Code (refer to student handbook)	<b>USIT304</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>III</b>
No of Credits	<b>2</b>
Nature	Theory
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	



## Nomenclature: Operating Systems

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Role of Operating System Computer System.
2. Use the different types of Operating System and their services.
3. Configure process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
4. Apply virtual memory concepts.
5. Effectively use and manage secondary memory.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	<b>Operating System Overview</b>	Objectives and Functions, Evolution, Achievements, Modern Operating Systems, Fault tolerance, OS design considerations for multiprocessor and multicore, overview of different operating systems	12
	<b>Processes</b>	Process Description and Control	
II	<b>Threads, Concurrency</b>	Mutual Exclusion and Synchronization	12
III	<b>Concurrency</b>	Deadlock and Starvation	12
	<b>Memory</b>	Memory Management, Virtual Memory	
IV	<b>Scheduling</b>	Uniprocessor Scheduling, Multiprocessor and Real-Time Scheduling	12
V	<b>IO and File Management</b>	I/O Management and Disk Scheduling, File Management, Operating System Security	12

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Operating Systems – Internals and Design Principles	Willaim Stallings	Pearson	9th	2009

2.	Operating System Concepts	Abraham Silberschatz, Peter B. Galvineg Gagne	Wiley	8th	
3.	Operating Systems	Godbole and Kahate	McGraw Hill	3rd	

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
3. Refer **appendix I** for Paper pattern.

Name of the Course	<b>Operating Systems Practical</b>
Course Code (refer to student handbook)	<b>USIT3P4</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>III</b>
No of Credits	<b>2</b>
Nature	Theory
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Operating Systems Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Install and configure operating systems.
2. Manage processes, threads, and system resources.
3. Implement memory management techniques.
4. Handle file system operations and data storage.
5. Troubleshoot and diagnose operating system issues.

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 1	Installation and Configuration of virtual machine a. Installation of virtual machine software. b. Installation of Windows OS c. Installation of Linux OS	3
2	Practical 2	Windows (DOS) Commands a. Date, time, prompt, md, cd, rd, path. b. Chkdsk, copy, xcopy, format, fidsk, cls, defrag, del, move. c. Diskcomp, diskcopy, diskpart, doskey, echo d. Edit, fc, find, rename, set, type, ver	3
3	Practical 3	Linux commands: a. pwd, cd, absolute and relative paths, ls, mkdir, rmdir b. file, touch, rm, cp, mv, rename, head, tail, cat, tac, more, less, strings, chmod c. ps, top, kill, pkill, bg, fg d. grep, locate, find, locate e. date, cal, uptime, w, whoami, finger, uname, man, df, du, free, whereis, which f. Compression: tar, gzip	3
4	Practical 4	Working with Linux Desktop and utilities a. The vi editor b. Graphics User Interface c. Working with Terminal d. Adjusting display resolution e. Using the browsers f. Configuring simple networking g. Creating users and shares	3
5	Practical 5	Installing utility software on Linux and Windows	3

<b>6</b>	<b>Practical 6</b>	Running C/C++/Python programs in Linux	<b>3</b>
<b>7</b>	<b>Practical 7</b>	Introduction to Linux Shell Scripting a. Basic operators b. Decision Making c. Looping d. Regular Expression e. Special variables and command Line arguments	<b>3</b>
<b>8</b>	<b>Practical 8</b>	Case study of Server OS Windows Server 2022 operating System - Architecture, Components, Services, Configuration	<b>3</b>
<b>9</b>	<b>Practical 9</b>	Case study of Android OS Architecture, Components, Services, Configuration	<b>3</b>
<b>10</b>	<b>Practical 10</b>	Case study of Cloud OS AWS, Azure, Google Cloud	<b>3</b>

This is sample Practical list. Course instructor may change the practical as per syllabus.

### Learning Resources recommended:

1. <https://ubuntu.com/download/desktop>
2. [https://sritsense.weebly.com/uploads/5/7/2/7/57272303/android\\_operating\\_system.pdf](https://sritsense.weebly.com/uploads/5/7/2/7/57272303/android_operating_system.pdf)

### Evaluation Pattern

#### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

#### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
<b>1</b>	<b>Program 1</b>	<b>10</b>
<b>2</b>	<b>Program 2</b>	<b>15</b>
<b>3</b>	<b>Viva</b>	<b>05</b>

Name of the Course	<b>Applied Mathematics</b>
Course Code (refer to student handbook)	<b>USIT305</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>III</b>
No of Credits	<b>2</b>
Nature	Theory
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Applied Mathematics

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Apply Knowledge of matrices to solve the problems.
2. Interpret the mathematical results in practical terms of complex numbers.
3. Solve and analyze the differential equations and its applications in related field of computers.
4. Understand various techniques of Laplace transform.
5. Solve multiple integrals.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	<b>Matrices</b>	Inverse of a matrix, Properties of matrices, Elementary Transformation, Rank of Matrix, Echelon or Normal Matrix, Linear equations, Linear dependence and linear independence of vectors, Linear transformation, Characteristics roots and characteristics vectors, Properties of characteristic vectors, Caley-Hamilton Theorem, Similarity of matrices, Reduction of matrix to a diagonal matrix which has elements as characteristics values.	12
	<b>Complex Numbers</b>	Complex number, Equality of complex numbers, Graphical representation of complex number(Argand's Diagram), Polar form of complex numbers, Polar form of $x+iy$ for different signs of $x,y$ , Exponential form of complex numbers, Mathematical operation with complex numbers and their representation on Argand's Diagram, Circular functions of complex angles, Definition of hyperbolic function, Relations between circular and hyperbolic functions, Inverse hyperbolic functions, Differentiation and Integration, Graphs of the hyperbolic functions, Logarithms of complex quality, $j(=i)$ as an operator(Electrical circuits)	
II	<b>Equation of the first order and of the first degree</b>	Separation of variables, Equations homogeneous in $x$ and $y$ , Non-homogeneous linear equations, Exact differential Equation, Integrating Factor, Linear Equation and equation reducible to this form, Method of substitution.	12

	<b>Differential equation of the first order of a degree higher than the first</b>	Introduction, Solvable for p (or the method of factors), Solve for y, Solve for x, Clairaut's form of the equation, Methods of Substitution	
	<b>Linear Differential Equations with Constant Coefficients</b>	Introduction, The Differential Operator, Linear Differential Equation $f(D)y = 0$ , Different cases depending on the nature of the root of the equation $f(D) = 0$ , Linear differential equation $f(D)y = X$ , The complimentary Function, The inverse operator $1/f(D)$ and the symbolic expiration for the particular integral $1/f(D)X$ ; the general methods, Particular integral : Short methods, Particular integral : Other methods, Differential equations reducible to the linear differential equations with constant coefficients.	
<b>III</b>	<b>The Laplace Transform</b>	Introduction, Definition of the Laplace Transform, Table of Elementary Laplace Transforms, Theorems on Important Properties of Laplace Transformation, First Shifting Theorem, Second Shifting Theorem, The Convolution Theorem, Laplace Transform of an Integral and Derivatives	<b>12</b>
	<b>Inverse Laplace Transform</b>	Shifting Theorem, Partial fraction Methods, Use of Convolution Theorem, Solution of Ordinary Linear Differential Equations with Constant Coefficients, Solution of Simultaneous Ordinary Differential Equations, Laplace Transformation of Special Function, Periodic Functions, Heaviside Unit Step Function, Dirac-delta Function(Unit Impulse Function)	
<b>IV</b>	<b>Multiple Integrals</b>	Double Integral, Change of the order of the integration, Double integral in polar coordinates, Triple integrals.	<b>12</b>
	<b>Applications of integration</b>	Areas, Volumes of solids	
<b>V</b>	<b>Beta and Gamma Functions</b>	Definitions, Properties and Problems, Duplication formula Differentiation Under the Integral Sign Error Functions	<b>12</b>



### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	A text book of Applied Mathematics Vol I	P. N. Wartikar and J. N. Wartikar	Pune Vidyarthi Griha		
2.	Applied Mathematics II	P. N. Wartikar and J. N. Wartikar	Pune Vidyarthi Griha		
3.	Higher Engineering Mathematics	Dr. B. S. Grewal	Khanna Publications		

### Evaluation Pattern

#### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks)	20
Assignments	10
Attendance and active participation in classroom	10

#### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

#### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
3. Refer **appendix I** for Paper pattern.

Name of the Course	<b>Mobile Programming Practical</b>
Course Code (refer to student handbook)	<b>USIT3P5</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>III</b>
No of Credits	<b>2</b>
Nature	Practical
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Mobile Programming Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Develop mobile applications for various platforms.
2. Utilize mobile development frameworks and tools effectively.
3. Implement user interfaces and user experience (UI/UX) design principles.
4. Integrate mobile applications with backend services and databases.
5. Test, debug, and deploy mobile applications on real devices.

### Curriculum:

The practicals will be based on HTML5, CSS, Flutter

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 0	Setting up Flutter, PhoneGAP Project and environment.	3
2	Practical 1	Program to demonstrate the features of Dart language.	3
3	Practical 2	Designing the mobile app to implement different widgets.	3
4	Practical 3	Designing the mobile app to implement different Layouts.	3
5	Practical 4	Designing the mobile app to implement Gestures.	3
6	Practical 5	Designing the mobile app to implement the theming and styling.	3
7	Practical 6	Designing the mobile app to implement the routing.	3
8	Practical 7	Designing the mobile app to implement the animation.	3
9	Practical 8	Designing the mobile app to implement the state management.	3
10	Practical 9	Designing the mobile app working with SQLite Database.	3
11	Practical 10	Designing the mobile app working with Firebase.	3

This is sample Practical list. Course instructor may change the practical as per syllabus.

**Learning Resources recommended:**

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1.	Flutter for Beginners	Alessandro Biessek	Packt Publishing		2019
2.	PhoneGap By Example	Andrey Kovalenko	Packt Publishing	1st	2015

**Evaluation Pattern**

**A. Continuous Internal Evaluation (20 Marks):**

<b>Method</b>	<b>Marks</b>
Certified Journal	10
Attendance and active participation in Laboratory	10

**B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):**

<b>Question No</b>	<b>Unit</b>	<b>Marks</b>
1	Program 1	10
2	Program 2	15
3	Viva	05

# SEMESTER IV

Name of the Course	<b>Core Java</b>
Course Code (refer to student handbook)	<b>USIT401</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>IV</b>
No of Credits	<b>2</b>
Nature	Theory
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Core Java

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Learn the architecture of Java
2. Identify data types, control flow, classes, inheritance, exceptions and event handling
3. Use object-oriented concepts for problem solving real-life applications
4. Build GUI programs
5. Create event driven programs using java.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	<b>Introduction</b>	History, Features of Java, Java Development Kit, Java Application Programming Interface, Java Virtual Machine, Java Program Structure.	12
	<b>Classes</b>	The Class Object and Its Attributes, Class Methods, Accessing A Method, Method Overloading, Instantiating Objects from A Class, Constructors, this keyword, super keyword, Types of Classes, Scope Rules, Access Modifier, constants, static members of a class, garbage collection.	
II	<b>Inheritance</b>	Derived Class Objects, Inheritance and Access Control, Default Base Class Constructors, this and super keywords. Abstract Classes and Interfaces, Abstract Classes, Abstract Methods,	
	<b>Interfaces</b>	What Is an Interface? How Is an Interface Different from An Abstract Class? Multiple Inheritance, Defining an Interface, Implementing Interfaces.	
III	<b>Exceptions</b>	Catching Java Exceptions, Catching Run-Time Exceptions, Handling Multiple Exceptions, The finally Clause, The throws Clause, Built-in Exceptions in java	12
	<b>Multithreading</b>	Thread Creations, Thread Life Cycle, Life Cycle Methods, Synchronization, wait() notify() notify all() methods	

	<b>Packages</b>	Introduction to predefined packages, User Defined Packages, Access specifier, Java Built-in packages, Array Class, String Class	
<b>IV</b>	<b>Introduction to JFC and Swing</b>	Features of the Java Foundation Classes, Swing API Components, JComponent Class, Containers and Panels, Labels, Buttons, RadioButton, Check Boxes, Text-Entry Components, Menus	<b>12</b>
	<b>Layouts</b>	Flow Layout, Grid Layout, Border Layout	
	<b>Event Handling</b>	Delegation Event Model, Events, Event classes, Event listener interfaces, Using delegation event model, adapter classes.	
<b>V</b>	<b>Advanced Swing Controls</b>	JScrollPane, Lists and Combo Boxes, Colors and File Choosers, Tables and Trees, JTabbedPane.	<b>12</b>
	<b>JDBC</b>	Introduction, JDBC Architecture, JDBC Drivers, java.sql package, Using Statement, PreparedStatement, CallableStatement, ResultSet	

#### Learning Resources recommended:

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1.	Core Java 8 for Beginners	Vaishali Shah, Sharnam Shah	SPD	1st	2015
2.	Java: The Complete Reference	Herbert Schildt	McGraw Hill	9th	2014
3.	Murach's beginning Java with Net Beans	Joel Murach , Michael Urban	SPD	1st	2016
4.	Core Java, Volume I: Fundamentals	Hortsman	Pearson	9th	2013
5.	Core Java, Volume II: Advanced Features	Gary Cornell and Hortsman	Pearson	8th	2008
6.	Core Java: An Integrated Approach	R. Nageswara Rao	DreamTech	1st	2008



## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks)	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

#### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
3. Refer **appendix I** for Paper pattern.

Name of the Course	<b>Core Java Practical</b>
Course Code (refer to student handbook)	<b>USIT4P1</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>IV</b>
No of Credits	<b>2</b>
Nature	Practical
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Core Java Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Implement Java programs to solve computational problems.
2. Demonstrate proficiency in object-oriented programming (OOP) concepts.
3. Utilize Java libraries and APIs for various functionalities.
4. Develop graphical user interfaces (GUI) using Java Swing or JavaFX.
5. Debug and troubleshoot Java code effectively.

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 1	OOPs concepts in Java – 1 a. Write a program to create a class and implement a default, overloaded and copy Constructor. b. Write a program to create a class and implement the concepts of Method Overloading c. Write a program to create a class and implement the concepts of Static methods	3
2	Practical 2	OOPs concepts in Java – 2 a. Write a program to implement the concepts of Inheritance and Method overriding b. Write a program to implement the concepts of Abstract classes and methods c. Write a program to implement the concept of interfaces	3
4	Practical 4	Multithreading: Write a java application to demonstrate 5 bouncing balls of different colors using threads.	3
5	Practical 5	JDBC a. Write a JDBC program that displays the data of a given table in a GUI Table. b. Write a JDBC program to Show the details of a specified product	3

		from a given table selected using Combobox. c. Write a GUI application to Navigate forward and reverse result set data.	
<b>6</b>	<b>Practical 6</b>	Swing a. Create a swing application that randomly changes color on button click. b. Create a Swing application to demonstrate use of TextArea using scrollpane to show content of text file in textarea selected using file chooser. c. Create a Swing application to demonstrate use of scrollpane to change its color selected using colour chooser.	<b>3</b>
<b>7</b>	<b>Practical 7</b>	Layouts: Write programs for the following layouts: a. Flow Layout b. Grid Layout c. Border Layout	<b>3</b>
<b>8</b>	<b>Practical 8</b>	Events: Write programs to demonstrate the following events: a. ActionEvent b. MouseEvent c. KeyEvent d. SelectionEvent e. FocusEvent	<b>3</b>
<b>9</b>	<b>Practical 9</b>	Demonstrate the use of Adapter Class in Event Handling.	<b>3</b>
<b>10</b>	<b>Practical 10</b>	Demonstrate the use of Anonymous Inner Class in Event Handling	<b>3</b>

This is sample Practical list. Course instructor may change the practical as per syllabus.

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Core Java 8 for Beginners	Vaishali Shah, Sharnam Shah	SPD	1st	2015
2	Java: The Complete Reference	Herbert Schildt	McGraw Hill	9th	2014
3	Murach's beginning Java with Net Beans	Joel Murach , Michael Urban	SPD	1st	2016
4	Core Java, Volume I: Fundamentals	Hortsman	Pearson	9th	2013
5	Core Java, Volume II: Advanced Features	Gary Cornell and Hortsman	Pearson	8th	2008
6	Core Java: An Integrated Approach	R. Nageswara Rao	DreamTech	1st	2008

### Evaluation Pattern

#### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

#### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

Name of the Course	<b>Introduction to Embedded Systems</b>
Course Code (refer to student handbook)	<b>USIT402</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>IV</b>
No of Credits	<b>2</b>
Nature	Theory
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Introduction to Embedded Systems

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Differentiate between general purpose and embedded systems
2. Discuss the characteristics and quality attributes of embedded systems.
3. Use different types of sensors appropriately.
4. Design and develop embedded systems

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	<b>PIC MICROCONTROLLER</b>	PIC MICROCONTROLLER: Architecture – memory organization – addressing modes – instruction set – PIC programming in Assembly & C –I/O port, Data Conversion, RAM & ROM Allocation, Timer programming	12
	<b>Advanced ARM Controllers</b>	Advanced ARM Controllers: Introduction to ARM and its Features, Architecture – memory organization – addressing modes – The ARM Programmer's model -Registers – Pipeline - Interrupts – Coprocessors – Interrupt Structure	
II	<b>Communication Protocol &amp; Implementation</b>	Communication Protocol & Implementation: Introduction to Communication Protocol, I2C - Interfacing with micro controller using bit-banking method, I2C devices – RTC, Memory, ADC-DAC, Port Expander, SPI (Serial Peripheral Interface), Bluetooth, Wi-Fi and RFID. Understanding Serial, Communication, Bluetooth Communication, SPI Interface ZigBee, Wi-Fi, I2C, Infrared, RFID, GSM, GPS, PDH/SDH/Ethernet	12
III	<b>Getting Started with Arduino</b>	Getting Started with Arduino: Introduction, Arduino Variants, Install the Drivers, Arduino IDE	12
	<b>Basic Functions</b>	Basic Functions: Overview, Structure, Digital I/O Functions, Analog	

		I/O Functions, Advanced I/O Functions, Timer Functions, Communication Functions, Interrupt Functions, Math Functions, Programming Language Reference	
<b>IV</b>	<b>Using Sensors with the Arduino</b>	Using Sensors with the Arduino: Light Sensitive Sensors, Temperature Sensors, Temperature and Humidity Sensor, Line-Tracking Sensor, Ultrasonic Sensors, Digital Infrared Motion Sensor, Joystick Module, Gas Sensor, Hall Sensor, Color Sensor, Digital Tilt Sensor, Triple Axis Acceleration Sensor, Analog Sound Sensor, Voice Recognition Module, Digital Vibration Sensor, Flame Sensor, Capacitive Touch Sensor	<b>12</b>
	<b>Electromechanical Control Using the Arduino</b>	Electromechanical Control Using the Arduino: DC Motor, Stepper Motor, Servo Motor	
<b>V</b>	<b>Wireless Control Using the Arduino</b>	Wireless Control Using the Arduino: Infrared Transmitter and Receiver, Wireless Radio Frequency, Bluetooth, GSM/GPRS, Wi-Fi	<b>12</b>
	<b>Case Studies</b>	Case Studies: <ul style="list-style-type: none"> <li>• Air Quality Monitor Using Arduino</li> <li>• A Fire-Fighting Robot Using Arduino</li> <li>• Intelligent Lock System Using Arduino</li> </ul>	

#### Learning Resources recommended:

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1	Programming Embedded Systems in C and C++	Michael Barr	O'Reilly	First	1999
2	Introduction to embedded systems	Shibu K V	Tata Mcgraw-Hill	First	2012
3	The 8051 Microcontroller and Embedded Systems	Muhammad Ali Mazidi	Pearson	Second	2011
4	Embedded Systems	Rajkamal	Tata Mcgraw-Hill		



## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

#### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
3. Refer **appendix I** for Paper pattern.

Name of the Course	<b>Introduction to Embedded Systems Practical</b>
Course Code (refer to student handbook)	<b>USIT4P2</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>IV</b>
No of Credits	<b>2</b>
Nature	Theory
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## **Nomenclature: Introduction to Embedded Systems Practical**

### **Course Outcomes:**

On the successful completion of this course, the learner will be able to

1. Design and implement basic embedded systems using microcontrollers.
2. Interface and control external devices and sensors with embedded systems.
3. Develop real-time embedded software for specific applications.
4. Understand and analyze the interaction between hardware and software in embedded systems.
5. Troubleshoot and debug embedded systems for optimal performance.

### **Curriculum:**

<b>Sr. No.</b>	<b>Title</b>	<b>Learning Points</b>	<b>No of Lectures</b>
<b>1</b>	<b>Practical 1</b>	<b>Introduction to Arduino Introduction to Arduino circuits and breadboarding Blinking of LEDs</b>	<b>3</b>
<b>2</b>	<b>Practical 2</b>	<b>Program using Light Sensitive Sensors</b>	<b>3</b>
<b>3</b>	<b>Practical 3</b>	<b>Program using temperature sensors</b>	<b>3</b>
<b>4</b>	<b>Practical 4</b>	<b>Programs using humidity sensors</b>	<b>3</b>
<b>5</b>	<b>Practical 5</b>	<b>Programs using Line tracking sensors</b>	<b>3</b>
<b>6</b>	<b>Practical 6</b>	<b>Programs using Ultrasonic Sensors</b>	<b>3</b>
<b>7</b>	<b>Practical 7</b>	<b>Programs using digital infrared motion sensors</b>	<b>3</b>
<b>8</b>	<b>Practical 8</b>	<b>Programs using gas sensors</b>	<b>3</b>
<b>9</b>	<b>Practical 9</b>	<b>Programs using servo motors</b>	<b>3</b>
<b>10</b>	<b>Practical 10</b>	<b>Programs making Joystick with Arduino</b>	<b>3</b>

This is sample Practical list. Course instructor may change the practical as per syllabus.

**Learning Resources recommended:**

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1	Programming Embedded Systems in C and C++	Michael Barr	O'Reilly	First	1999
2	Introduction to embedded systems	Shibu K V	Tata Mcgraw-Hill	First	2012

**Evaluation Pattern****A. Continuous Internal Evaluation (20 Marks):**

<b>Method</b>	<b>Marks</b>
Certified Journal	10
Attendance and active participation in Laboratory	10

**B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):**

<b>Question No</b>	<b>Unit</b>	<b>Marks</b>
<b>1</b>	<b>Program 1</b>	<b>10</b>
<b>2</b>	<b>Program 2</b>	<b>15</b>
<b>3</b>	<b>Viva</b>	<b>05</b>

Name of the Course	<b>Computer Oriented Statistical Techniques</b>
Course Code (refer to student handbook)	<b>USIT403</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>IV</b>
No of Credits	<b>2</b>
Nature	Theory
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Computer Oriented Statistical Techniques

### Course Outcomes:

On the successful completion of the course, students will be able:

1. To calculate and apply measures of central tendencies and measures of dispersion --grouped and ungrouped data cases
2. To calculate the moments, skewness and kurtosis by various methods.
3. To apply discrete and continuous probability distributions to various business problems.
4. To perform Test of Hypothesis as well as calculate confidence interval for a population parameter for single sample and two sample cases. Understand the concept of p-values
5. To apply simple linear regression and correlation model to real life examples.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	<b>Measures of Central Tendency</b>	Index, or Subscript, Notation, Summation Notation, Averages, or Measures of Central Tendency ,The Arithmetic Mean, The Weighted Arithmetic Mean ,Properties of the Arithmetic Mean, The Arithmetic Mean Computed from Grouped Data ,The Median ,The Mode, The Empirical Relation Between the Mean, Median, and Mode,The Geometric Mean G, The Harmonic Mean H ,The Relation Between the Arithmetic, Geometric, and Harmonic Means, The Root Mean Square, Quartiles, Deciles, and Percentiles, Software and Measures of Central Tendency.	12
	<b>Measures of Dispersion</b>	Dispersion, or Variation, The Range, The Mean Deviation, The Semi-Interquartile Range, The 10–90 Percentile Range, The Standard Deviation, The Variance, Short Methods for Computing the Standard Deviation, Properties of the Standard Deviation, Charlie’s Check, Sheppard’s Correction for Variance, Empirical Relations Between Measures of Dispersion, Absolute and Relative Dispersion; Coefficient of Variation, Standardized Variable; Standard Scores, Software and Measures of Dispersion.	
	<b>Introduction to R</b>	Basic syntax, data types, variables, operators, control statements, R-functions, R –Vectors, R –lists, R Arrays.	

<b>II</b>	<b>Moments, Skewness, and Kurtosis</b>	Moments , Moments for Grouped Data, Relations Between Moments, Computation of Moments for Grouped Data, Charlie's Check and Sheppard's Corrections, Moments in Dimensionless Form, Skewness, Kurtosis, Population Moments, Skewness and Kurtosis, Software Computation of Skewness and Kurtosis.	<b>12</b>
	<b>Elementary Probability Theory</b>	Definitions of Probability, Conditional Probability; Independent and Dependent Events, Mutually Exclusive Events, Probability Distributions, Mathematical Expectation, Relation Between Population, Sample Mean, and Variance, Combinatorial Analysis, Combinations, Stirling's Approximation to $n!$ , Relation of Probability to Point Set Theory, Euler or Venn Diagrams and Probability.	
	<b>Elementary Sampling Theory</b>	Sampling Theory, Random Samples and Random Numbers, Sampling With and Without Replacement, Sampling Distributions, Sampling Distribution of Means, Sampling Distribution of Proportions, Sampling Distributions of Differences and Sums, Standard Errors, Software Demonstration of Elementary Sampling Theory.	
<b>III</b>	<b>Statistical Estimation Theory</b>	Estimation of Parameters, Unbiased Estimates, Efficient Estimates, Point Estimates and Interval Estimates; Their Reliability, Confidence-Interval Estimates of Population Parameters, Probable Error.	<b>12</b>
	<b>Statistical Decision Theory</b>	Statistical Decisions, Statistical Hypotheses, Tests of Hypotheses and Significance, or Decision Rules, Type I and Type II Errors, Level of Significance, Tests Involving Normal Distributions, Two-Tailed and One-Tailed Tests, Special Tests, Operating-Characteristic Curves; the Power of a Test, p-Values for Hypotheses Tests, Control Charts, Tests Involving Sample Differences, Tests Involving Binomial Distributions.	
	<b>Statistics in R</b>	mean, median, mode, Normal Distribution , Binomial Distribution, Frequency Distribution in R.	

<b>IV</b>	<b>Small Sampling Theory</b>	Small Samples, Student's t Distribution, Confidence Intervals, Tests of Hypotheses and Significance, The Chi-Square Distribution, Confidence Intervals for Sigma , Degrees of Freedom, The F Distribution.	<b>12</b>
	<b>The Chi-Square Test</b>	Observed and Theoretical Frequencies, Definition of chi-square, Significance Tests, The Chi-Square Test for Goodness of Fit, Contingency Tables, Yates' Correction for Continuity, Simple Formulas for Computing chi-square, Coefficient of Contingency, Correlation of Attributes, Additive Property of chi-square.	
<b>V</b>	<b>Curve Fitting and the Method of Least Squares</b>	Relationship Between Variables, Curve Fitting, Equations of Approximating Curves, Freehand Method of Curve Fitting, The Straight Line, The Method of Least Squares, The Least-Squares Line, Nonlinear Relationships, The Least-Squares Parabola, Regression, Applications to Time Series, Problems Involving More Than Two Variables.	<b>12</b>
	<b>Correlation Theory</b>	Correlation and Regression, Linear Correlation, Measures of Correlation, The Least-Squares Regression Lines, Standard Error of Estimate, Explained and Unexplained Variation, Coefficient of Correlation, Remarks Concerning the Correlation Coefficient, Product-Moment Formula for the Linear Correlation Coefficient, Short Computational Formulas, Regression Lines and the Linear Correlation Coefficient, Correlation of Time Series, Correlation of Attributes, Sampling Theory of Correlation, Sampling Theory of Regression.	

**Learning Resources recommended:**

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1.	Statistics	Murray R. Spiegel, Larry J. Stephens.	McGraw – Hill International	4th	



2.	A Practical Approach using R	R.B. Patil, H.J. Dand and R. Bhavsar	SPD	1st	2017
3.	Fundamental of Mathematical Statistics	S.C. Gupta and V. K. Kapoor	Sultan Chand and Sons	11th Revised	2011
4.	Mathematical Statistics	J. N. Kapur and H.C. saxena	S. Chand	20th Revised	2005

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

### Guidelines for paper pattern for Semester End Evaluation:

- All questions will be compulsory.
- Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
- Refer **appendix I** for Paper pattern.

Name of the Course	<b>Computer Oriented Statistical Techniques Practical</b>
Course Code (refer to student handbook)	<b>USIT4P3</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>IV</b>
No of Credits	<b>2</b>
Nature	Practical
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Computer Oriented Statistical Techniques Practical

### Course Outcomes:

On the successful completion of the course, students will be able to:

1. Apply statistical methods to analyze and interpret data sets.
2. Use statistical software to perform data analysis efficiently.
3. Understand and implement various statistical techniques for hypothesis testing and regression analysis.
4. Present statistical findings visually through graphs and charts.
5. Apply statistical techniques to solve real-world problems in different domains.

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 1	Using R/Python execute the basic commands, array, list and frames.	3
2	Practical 2	Create a Matrix using R/Python and Perform the operations addition, inverse, transpose and multiplication operations.	3
3	Practical 3	Using R/Python Execute the statistical functions: mean, median, mode, quartiles, range, inter quartile range histogram	3
4	Practical 4	Using R/Python import the data from Excel / .CSV file and Perform the above functions.	3
5	Practical 5	Using R/Python import the data from Excel / .CSV file and Calculate the standard deviation, variance, co-variance.	3
6	Practical 6	Using R/Python import the data from Excel / .CSV file and draw the skewness.	3
7	Practical 7	Import the data from Excel / .CSV and perform the hypothesis testing.	3
8	Practical 8	Import the data from Excel / .CSV and perform the Chi-squared Test.	3
9	Practical 9	Using R/Python perform the binomial and normal distribution on the data.	3

<b>10</b>	<b>Practical 10</b>	Perform the Linear Regression using R/Python. Compute the Least squares means using R/Python. Compute the Linear Least Square Regression using R/Python	<b>3</b>
-----------	-------------------------	---	----------

This is sample Practical list. Course instructor may change the practical as per syllabus.

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	A Practical Approach using R	R.B. Patil, H.J. Dand and R. Bhavsar	SPD	1st	2017

### Evaluation Pattern

#### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

#### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
<b>1</b>	<b>Program 1</b>	<b>10</b>
<b>2</b>	<b>Program 2</b>	<b>15</b>
<b>3</b>	<b>Viva</b>	<b>05</b>

Name of the Course	<b>Software Engineering</b>
Course Code (refer to student handbook)	<b>USIT404</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>IV</b>
No of Credits	<b>2</b>
Nature	Theory
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Software Engineering

### Course Outcomes:

On the successful completion of the course, students will be able to:

1. Understand software engineering
2. Apply software engineering principles
3. Discuss various approaches to verification and validation of software including testing, measurements and estimation of software products
4. Create software using different software development models

### Curriculum:

Unit	Title	Learning Points	No of Lectures
<b>I</b>	<b>Introduction</b>	Introduction: What is software engineering? Software Development Life Cycle, Requirements Analysis, Software Design, Coding, Testing, Maintenance etc.	<b>12</b>
	<b>Software Requirements</b>	Software Requirements: Functional and Non-functional requirements, User Requirements, System Requirements, Interface Specification, Documentation of the software requirements.	
	<b>Software Processes</b>	Software Processes: Process and Project, Component Software Processes.	
	<b>Software Development Process Models</b>	Software Development Process Models. <ul style="list-style-type: none"><li>• Waterfall Model.</li><li>• Prototyping.</li><li>• Iterative Development.</li><li>• Rational Unified Process.</li><li>• The RAD Model</li><li>• Time boxing Model.</li></ul>	
	<b>Agile software development</b>	Agile software development: Agile methods, Plan-driven and agile development, Extreme programming, Agile project management, Scaling agile methods.	
<b>II</b>	<b>Socio-technical system</b>	Socio-technical system: Essential characteristics of socio technical	<b>12</b>

		systems, Emergent System Properties, Systems Engineering, Components of system such as organization, people and computers, Dealing Legacy Systems.	
	<b>Critical system</b>	Critical system: Types of critical system, A simple safety critical system, Dependability of a system, Availability and Reliability, Safety and Security of Software systems.	
	<b>Requirements Engineering Processes</b>	Requirements Engineering Processes: Feasibility study, Requirements elicitation and analysis, Requirements Validations, Requirements Management.	
	<b>System Models</b>	System Models: Models and its types, Context Models, Behavioural Models, Data Models, Object Models, Structured Methods.	
<b>III</b>	<b>Architectural Design</b>	Architectural Design: Architectural Design Decisions, System Organisation, Modular Decomposition Styles, Control Styles, Reference Architectures.	<b>12</b>
	<b>User Interface Design</b>	User Interface Design: Need of UI design, Design issues, The UI design Process, User analysis, User Interface Prototyping, Interface Evaluation.	
	<b>Project Management</b>	Project Management Software Project Management, Management activities, Project Planning, Project Scheduling, Risk Management.	
	<b>Quality Management</b>	Quality Management: Process and Product Quality, Quality assurance and Standards, Quality Planning, Quality Control, Software Measurement and Metrics.	
<b>IV</b>	<b>Verification and Validation</b>	Verification and Validation: Planning Verification and Validation, Software Inspections, Automated Static Analysis, Verification and Formal Methods.	<b>12</b>
	<b>Software Testing</b>	Software Testing: System Testing, Component Testing, Test Case Design, Test Automation.	

	<b>Software Measurement</b>	Software Measurement: Size-Oriented Metrics, Function-Oriented Metrics, Extended Function Point Metrics	
	<b>Software Cost Estimation</b>	Software Cost Estimation: Software Productivity, Estimation Techniques, Algorithmic Cost Modelling, Project Duration and Staffing	
<b>V</b>	<b>Process Improvement</b>	Process and product quality, Process Classification, Process Measurement, Process Analysis and Modeling, Process Change, The CMMI Process Improvement Framework.	<b>12</b>
	<b>Service Oriented Software Engineering</b>	Service Oriented Software Engineering: Services as reusable components, Service Engineering, Software Development with Services.	
	<b>Software reuse</b>	Software reuse: The reuse landscape, Application frameworks, Software product lines, COTS product reuse.	
	<b>Distributed software engineering</b>	Distributed software engineering: Distributed systems issues, Client–server computing, Architectural patterns for distributed systems, Software as a service	

#### Learning Resources recommended:

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1	Software Engineering,	Ian Somerville	Pearson Education.	Ninth	
2	Software Engineering	Pankaj Jalote	Narosa Publication		
3	Software engineering, a practitioner's approach	Roger Pressman	Tata Mcgraw-hill	Seventh	
4	Software Engineering principles and practice	WS Jawadekar	Tata Mcgraw-hill		



5	Software Engineering- A Concise Study	S.A Kelkar	PHI India.		
6	Software Engineering Concept and Applications	Subhajit Datta	Oxford Higher Education		
7	Software Design	D.Budgen	Pearson education	2nd	
8	Software Engineering	KL James	PHI	EEE	2009

### Evaluation Pattern

#### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

#### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

#### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
3. Refer **appendix I** for Paper pattern.

Name of the Course	<b>Software Engineering Practical</b>
Course Code (refer to student handbook)	<b>USIT4P4</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>IV</b>
No of Credits	<b>2</b>
Nature	Practical
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## **Nomenclature: Software Engineering Practical**

### **Course Outcomes:**

On the successful completion of the course, students will be able to:

1. Apply software development methodologies to manage and execute projects effectively.
2. Design and implement software systems following industry best practices and standards.
3. Collaborate in teams to develop and maintain software products.
4. Utilize software testing techniques to ensure the quality and reliability of software.

### **Curriculum:**

<b>Sr. No.</b>	<b>Title</b>	<b>Learning Points</b>	<b>No of Lectures</b>
<b>1</b>	<b>Practical 1</b>	<b>Study and implementation of class diagrams.</b>	<b>3</b>
<b>2</b>	<b>Practical 2</b>	<b>Study and implementation of Use Case Diagrams.</b>	<b>3</b>
<b>3</b>	<b>Practical 3</b>	<b>Study and implementation of Entity Relationship Diagrams.</b>	<b>3</b>
<b>4</b>	<b>Practical 4</b>	<b>Study and implementation of Sequence Diagrams.</b>	<b>3</b>
<b>5</b>	<b>Practical 5</b>	<b>Study and implementation of State Transition Diagrams.</b>	<b>3</b>
<b>6</b>	<b>Practical 6</b>	<b>Study and implementation of Data Flow Diagrams.</b>	<b>3</b>
<b>7</b>	<b>Practical 7</b>	<b>Study and implementation of Collaboration Diagrams.</b>	<b>3</b>
<b>8</b>	<b>Practical 8</b>	<b>Study and implementation of Activity Diagrams.</b>	<b>3</b>
<b>9</b>	<b>Practical 9</b>	<b>Study and implementation of Component Diagrams.</b>	<b>3</b>
<b>10</b>	<b>Practical 10</b>	<b>Study and implementation of Deployment Diagrams.</b>	<b>3</b>

This is sample Practical list. Course instructor may change the practical as per syllabus.

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Object - Oriented Modeling and Design	Michael Blaha, James Rumbaugh	Pearson		2011
2	Learning UML 2. 0	Kim Hamilton, Russ, Miles	O'Reilly Media		2006
3	The unified modeling language user guide	Grady Booch, James Rumbaugh, Ivar Jacobson	Addison-Wesley		2005
4	UML A Beginners Guide	Jason T. Roff	McGraw Hill Professional		2003

### Evaluation Pattern

#### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

#### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

Name of the Course	<b>Computer Graphics and Animation</b>
Course Code (refer to student handbook)	<b>USIT405</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>IV</b>
No of Credits	<b>2</b>
Nature	Theory
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Computer Graphics and Animation

### Course Outcomes:

On the successful completion of the course, students will be able to:

1. Understand the basics of computer graphics, different graphics systems and applications of computer graphics
2. Compare various algorithms for scan conversion and filling of basic objects
3. Use of geometric transformations on graphics objects and their application in composite form.
4. Extract scene with different clipping methods and its transformation to graphics display device.
5. Explore projections and visible surface detection techniques for display of 3D scene on 2D screen.
6. Render projected objects to naturalize the scene in 2D view and use of illumination models
7. Understand the core concepts and mathematical foundations of computer graphics
8. Know the fundamental computer graphics algorithms and data structures
9. Understand an overview of different modeling approaches and methods
10. Apply basic shading and texture mapping techniques
11. Understand light interaction with 3D scenes
12. Explain the applications, areas, and graphic pipeline, display and hardcopy technologies.
13. Apply and compare the algorithms for drawing 2D images also explain aliasing, anti-aliasing and half toning techniques.
14. Discuss OpenGL application programming Interface and apply it for 2D & 3D computer graphics.
15. Analyze and apply clipping algorithms and transformation on 2D images.
16. Solve the problems on viewing transformations and explain the projection and hidden surface removal algorithms.
17. Apply basic ray tracing algorithm, shading, shadows, curves and surfaces and also solve the problems of curves.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	<b>Introduction to Computer Graphics</b>	Introduction to Computer Graphics: Overview of Computer Graphics, Computer Graphics Application and Software, Description of some graphics devices, Input Devices for	12

		Operator Interaction, Active and Passive Graphics Devices, Display Technologies, Storage Tube Graphics Displays, Calligraphic Refresh Graphics Displays, Raster Refresh (Raster-Scan) Graphics Displays, Cathode Ray Tube Basics, Color CRT Raster Scan Basics, Video Basics, The Video Controller, Random-Scan Display Processor, LCD displays.	
	<b>Scan conversion</b>	Scan conversion – Digital Differential Analyzer (DDA) algorithm, Bresenhams’ Line drawing algorithm. Bresenhams’ method of Circle drawing, Midpoint Circle Algorithm, Midpoint Ellipse Algorithm, Mid-point criteria, Problems of Aliasing, end-point ordering and clipping lines, Scan Converting Circles, Clipping Lines algorithms–Cyrus-Beck, Cohen-Sutherland and Liang-Barsky, Clipping Polygons, problem with multiple components.	
<b>II</b>	<b>Two-Dimensional Transformations</b>	Two-Dimensional Transformations: Transformations and Matrices, Transformation Conventions, 2D Transformations, Homogeneous Coordinates and Matrix Representation of 2D Transformations, Translations and Homogeneous Coordinates, Rotation, Reflection, Scaling, Combined Transformation, Transformation of Points, Transformation of The Unit Square, Solid Body Transformations, Rotation About an Arbitrary Point, Reflection through an Arbitrary Line, A Geometric Interpretation of Homogeneous Coordinates, The Window-to-Viewport Transformations.	<b>12</b>
	<b>Three-Dimensional Transformations</b>	Three-Dimensional Transformations: Three-Dimensional Scaling, Three-Dimensional Shearing, Three-Dimensional Rotation, Three-Dimensional Reflection, Three-	

		Dimensional Translation, Multiple Transformation, Rotation about an Arbitrary Axis in Space, Reflection through an Arbitrary Plane, Matrix Representation of 3D Transformations, Composition of 3D Transformations, Affine and Perspective Geometry, Perspective Transformations, Techniques for Generating Perspective Views, Vanishing Points, the Perspective Geometry and camera models, Orthographic Projections, Axonometric Projections, Oblique Projections, View volumes for projections.	
<b>III</b>	<b>Viewing in 3D</b>	Viewing in 3D Stages in 3D viewing, Canonical View Volume (CVV), Specifying an Arbitrary 3D View, Examples of 3D Viewing, The Mathematics of Planar Geometric Projections, Combined transformation matrices for projections and viewing, Coordinate Systems and matrices, camera model and viewing pyramid.	<b>12</b>
	<b>Light</b>	Light: Radiometry, Transport, Equation, Photometry	
	<b>Color</b>	Color: Colorimetry, Color Spaces, Chromatic Adaptation, Color Appearance	
<b>IV</b>	<b>Visible-Surface Determination</b>	Visible-Surface Determination: Techniques for efficient Visible-Surface Algorithms, Categories of algorithms, Back face removal, The z-Buffer Algorithm, Scan-line method, Painter's algorithms (depth sorting), Area sub-division method, BSP trees, Visible-Surface Ray Tracing, comparison of the methods.	<b>12</b>
	<b>Plane Curves and Surfaces</b>	Plane Curves and Surfaces: Curve Representation, Nonparametric Curves, Parametric Curves, Parametric Representation of a Circle, Parametric Representation of an	



		Ellipse, Parametric Representation of a Parabola, Parametric Representation of a Hyperbola, Representation of Space Curves, Cubic Splines, , Bezier Curves, B-spline Curves, B-spline Curve Fit, B-spline Curve Subdivision, Parametric Cubic Curves, Quadric Surfaces. Bezier Surfaces.	
<b>V</b>	<b>Computer Animation</b>	Computer Animation: Principles of Animation, Key framing, Deformations, Character Animation, Physics-Based Animation, Procedural Techniques, Groups of Objects.	<b>12</b>
	<b>Image Manipulation and Storage</b>	Image Manipulation and Storage: What is an Image? Digital image file formats, Image compression standard – JPEG, Image Processing - Digital image enhancement, contrast stretching, Histogram Equalization, smoothing and median Filtering.	

**Learning Resources recommended:**

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1	Computer Graphics - Principles and Practice	J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes	Pearson	2 <sup>nd</sup>	
2	Steve Marschner, Peter Shirley	Fundamentals of Computer Graphics	CRC press	4th	2016
3	Computer Graphics	Hearn, Baker	Pearson	2nd	
4	Principles of Interactive Computer Graphics	William M. Newman and Robert F. Sproull	TMH	2nd	

5	Mathematical Elements for CG	D. F. Rogers, J. A. Adams	TMH	2nd	
---	------------------------------	---------------------------	-----	-----	--

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
3. Refer **appendix I** for Paper pattern.

Name of the Course	<b>Computer Graphics and Animation Practical</b>
Course Code (refer to student handbook)	<b>USIT4P5</b>
Class	<b>S.Y.B.Sc.IT</b>
Semester	<b>IV</b>
No of Credits	<b>2</b>
Nature	Practical
Type (applicable to NEP only)	Core
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Computer Graphics and Animation Practical

### Course Outcomes:

On the successful completion of the course, students will be able to:

1. Develop interactive computer graphics applications using graphics libraries and APIs.
2. Implement 2D and 3D transformations, projections, and rendering techniques.
3. Create visually appealing animations and special effects.
4. Understand and apply principles of computer graphics to solve graphical challenges.
5. Collaborate in teams to design and showcase computer graphic projects.

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 1	Solve the following: a. Study and enlist the basic functions used for graphics in C / C++ / Python language. Give an example for each of them. b. Draw a co-ordinate axis at the center of the screen.	3
2	Practical 2	Solve the following: a. Divide your screen into four region, draw circle, rectangle, ellipse and half ellipse in each region with appropriate message. b. Draw a simple hut on the screen.	3
3	Practical 3	Draw the following basic shapes in the center of the screen : i. Circle ii. Rectangle iii. Square iv. Concentric Circles v. Ellipse vi. Line	3
4	Practical 4	Solve the following: a. Develop the program for DDA Line drawing algorithm. b. Develop the program for Bresenham's Line drawing algorithm.	3
5	Practical 5	Solve the following: a. Develop the program for the mid-point circle drawing algorithm. b. Develop the program for the mid-point ellipse drawing algorithm.	3
6	Practical 6	Solve the following: a. Write a program to implement 2D scaling. b. Write a program to perform 2D translation	3
7	Practical 7	Solve the following:	3

		a. Perform 2D Rotation on a given object. b. Program to create a house like figure and perform the following operations. i. Scaling about the origin followed by translation. ii. Scaling with reference to an arbitrary point. iii. Reflect about the line $y=mx+c$	
<b>8</b>	<b>Practical 8</b>	Solve the following: a. Write a program to implement Cohen-Sutherland clipping. b. Write a program to implement Liang - Barsky Line Clipping Algorithm	<b>3</b>
<b>9</b>	<b>Practical 9</b>	Solve the following: a. Write a program to fill a circle using Flood Fill Algorithm. b. Write a program to fill a circle using Boundary Fill Algorithm.	<b>3</b>
<b>10</b>	<b>Practical 10</b>	Solve the following: a. Develop a simple text screen saver using graphics functions. b. Perform smiling face animation using graphic functions. c. Draw the moving car on the screen.	<b>3</b>

This is sample Practical list. Course instructor may change the practical as per syllabus.

#### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Computer Graphics - Principles and Practice	J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes	Pearson Education	Second Edition	
2	Steve Marschner, Peter Shirley	Fundamentals of Computer Graphics	CRC press	Fourth Edition	2016
3	Computer Graphics	Hearn, Baker	Pearson Education	Second	
4	Principles of Interactive Computer Graphics	William M. Newman and Robert F. Sproull	Tata McGraw Hill	Second	

## Evaluation Pattern

### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

---

# APPENDIX I

## PAPER PATTERN

### 1. Internal Evaluation

Test: 1 Class test of 20 marks. (Can be taken online/offline)

<b>Q</b>	<b>Attempt <i>any four</i> of the following:</b>	<b>20</b>
a.		
b.		
c.		
d.		
e.		
f.		

### 2. External Examination: (60 marks) (Currently for SY and TY)

	<b>All questions are compulsory</b>	
<b>Q1</b>	<b>(Based on Unit 1) Attempt <i>any three</i> of the following:</b>	<b>12</b>
a.		
b.		
c.		
d.		
e.		
f.		
<b>Q2</b>	<b>(Based on Unit 2) Attempt <i>any three</i> of the following:</b>	<b>12</b>
<b>Q3</b>	<b>(Based on Unit 3) Attempt <i>any three</i> of the following:</b>	<b>12</b>
<b>Q4</b>	<b>(Based on Unit 4) Attempt <i>any three</i> of the following:</b>	<b>12</b>
<b>Q5</b>	<b>(Based on Unit 5) Attempt <i>any three</i> of the following:</b>	<b>12</b>

**3. External Examination: (60 marks) (For FY)**

	<b>All questions are compulsory</b>	
<b>Q1</b>	<b>(Based on Unit 1) Attempt <i>any three</i> of the following:</b>	<b>15</b>
<b>a.</b>		
<b>b.</b>		
<b>c.</b>		
<b>d.</b>		
<b>e.</b>		
<b>f.</b>		
<b>Q2</b>	<b>(Based on Unit 2) Attempt <i>any three</i> of the following:</b>	<b>15</b>
<b>Q3</b>	<b>(Based on Unit 3) Attempt <i>any three</i> of the following:</b>	<b>15</b>
<b>Q4</b>	<b>(Based on Units 1,2 &amp; 3) Attempt <i>any three</i> of the following:</b>	<b>15</b>
	<b>(Set 2 Sub questions on each unit)</b>	



## Scheme of Courses for T.Y.B.Sc. I.T.

<b>Semester – V</b>				
<b>Course Code</b>	<b>Course Type</b>	<b>Course Title</b>	<b>Credits</b>	<b>No of lects/ week</b>
USIT501	Skill Enhancement Course	Software Project Management	2	5
USIT502	Skill Enhancement Course	Internet of Things	2	5
USIT503	Skill Enhancement Course	Advanced Web Programming	2	5
USIT504	Discipline Specific Elective (Any One)	Artificial Intelligence	2	5
USIT505		Linux System Administration	2	5
USIT506	Discipline Specific Elective (Any One)	Enterprise Java	2	3
USIT507		Next Generation Technologies	2	3
USIT5P1	Skill Enhancement Course Practical	Project Dissertation	2	3
USIT5P2	Skill Enhancement Course Practical	Internet of Things Practical	2	3
USIT5P3	Skill Enhancement Course Practical	Advanced Web Programming Practical	2	3
USIT5P4	Discipline Specific Elective Practical (Any One)*	Artificial Intelligence Practical	2	3
USIT5P5		Linux Administration Practical	2	3
USIT5P6	Discipline Specific Elective Practical (Any One)*	Enterprise Java Practical	2	3
USIT5P7		Next Generation Technologies Practical	2	3
<b>Total Credits</b>			<b>20</b>	

<b>Semester – VI</b>				
<b>Course Code</b>	<b>Course Type</b>	<b>Course Title</b>	<b>Credits</b>	<b>No of lects/ week</b>
USIT601	Skill Enhancement Course	Software Quality Assurance	2	5
USIT602	Skill Enhancement Course	Security in Computing	2	5
USIT603	Skill Enhancement Course	Business Intelligence	2	5
USIT604	Discipline Specific Elective (Any One)	Principles of Geographic Information Systems	2	5
USIT605		Enterprise Networking		5
USIT606	Discipline Specific Elective (Any One)	IT Service Management	2	3
USIT607		Cyber Laws		3
USIT6P1	Skill Enhancement Course Practical	Project Implementation	2	3
USIT6P2	Skill Enhancement Course Practical	Security in Computing Practical	2	3
USIT6P3	Skill Enhancement Course Practical	Business Intelligence Practical	2	3
USIT6P4	Discipline Specific Elective Practical (Any One)*	Principles of Geographic Information Systems Practical	2	3
USIT6P5		Enterprise Networking Practical		
USIT6P6	Skill Enhancement Course Practical	Advanced Mobile Programming	2	3
<b>Total Credits</b>			<b>20</b>	

**\*The choice of Practical course is based on the theory Course. For Semester V, USIT504, USIT505, USIT506 and USIT507, the practical courses are USIT5P4, USIT5P5, USIT5P6, USIT5P7. For Semester VI, USIT604, USIT605 the practical courses are USIT6P4, USIT6P5 respectively. Practical Course USIT6P6 is compulsory.**

# SEMESTER V

## Syllabus for T.Y.B.Sc.IT Autonomous from the year 2023-24

Name of the Course	<b>Software Project Management</b>
Course Code (refer to student handbook)	<b>USIT501</b>
Class	<b>T.Y.B.Sc.IT</b>
Semester	<b>V</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Core</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<p><b>Employability:</b> The demand for skilled software project managers is growing rapidly. As organizations continue to embrace technology and undertake software development projects, the need for professionals who can effectively manage these projects, teams, and resources becomes crucial. <b>Entrepreneurship:</b> Entrepreneurs in the software industry need strong project management skills to bring their ideas to life. They must be able to define project scopes, create realistic schedules, allocate resources, and manage risks effectively.</p> <p><b>Skill Development:</b> Software project management skills encompass defining project goals, creating work breakdown structures, estimating effort and resources, and scheduling tasks. Acquiring these skills helps individuals effectively plan and execute software projects.</p>

## Nomenclature: Software Project Management

### Course Outcomes:

On the successful completion of this course, the learner will be able to-

1. Proper understanding of Project Management Principles
2. Knowledge of Software Development Life Cycle (SDLC)
3. Create Project Plans
4. Team Management and Leadership Skills
5. Risk Assessment and Mitigation Problem Solutions
6. Thorough communication and Stakeholder Management

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	Introduction to Software Project Management:	Introduction, Why is Software Project Management Important? What is a Project? Software Projects versus Other Types of Project, Contract Management and Technical Project Management, Activities Covered by Software Project Management, Plans, Methods and Methodologies, Some Ways of Categorizing Software Projects, Project Charter, Stakeholders, Setting Objectives, The Business Case, Project Success and Failure, What is Management? Management Control, Project Management Life Cycle, Traditional versus Modern Project Management Practices.	12
	Project Evaluation and Programme Management:	Introduction, Business Case, Project Portfolio Management, Evaluation of Individual Projects, Cost-benefit Evaluation Techniques, Risk Evaluation, Programme Management, Managing the Allocation of Resources within Programmes, Strategic Programme Management, Creating a Programme, Aids to Programme Management, Some Reservations about Programme Management, Benefits Management.	
	An Overview of Project Planning:	Introduction to Step Wise Project Planning, Step 0: Select Project, Step 1: Identify Project Scope and Objectives, Step 2: Identify Project Infrastructure, Step 3: Analyze Project Characteristics, Step 4: Identify Project Products and Activities, Step5: Estimate Effort for Each Activity, Step 6: Identify Activity Risks, Step 7: Allocate Resources, Step 8: Review/Publicize Plan, Steps 9 and 10: Execute Plan/Lower Levels of Planning	

II	Selection of an Appropriate Project Approach:	Introduction, Build or Buy? Choosing Methodologies and Technologies, Software Processes and Process Models, Choice of Process Models, Structure versus Speed of Delivery, The Waterfall Model, The Spiral Model, Software Prototyping, Other Ways of Categorizing Prototypes, Incremental Delivery, Atern/Dynamic Systems Development Method, Rapid Application Development, Agile Methods, Extreme Programming (XP), Scrum, Lean Software Development, Managing Iterative Processes, Selecting the Most Appropriate Process Model.	12
	Software Effort Estimation:	Introduction, Where are the Estimates Done? Problems with Over- and Under-Estimates, The Basis for Software Estimating, Software Effort Estimation Techniques, Bottom-up Estimating, The Top-down Approach and Parametric Models, Expert Judgement, Estimating by Analogy, Albrecht Function Point Analysis, Function Points Mark II, COSMIC Full Function Points, COCOMO II: A Parametric Productivity Model, Cost Estimation, Staffing Pattern, Effect of Schedule Compression, Capers Jones Estimating Rules of Thumb.	
III	Activity Planning:	Introduction, Objectives of Activity Planning, When to Plan, Project Schedules, Projects and Activities, Sequencing and Scheduling Activities, Network Planning Models, Formulating a Network Model, Adding the Time Dimension, The Forward Pass, Backward Pass, Identifying the Critical Path, Activity Float, Shortening the Project Duration, Identifying Critical Activities, Activity-on-Arrow Networks.	12
	Risk Management:	Introduction, Risk, Categories of Risk, Risk Management Approaches, A Framework for Dealing with Risk, Risk Identification, Risk Assessment, Risk Planning, Risk Management, Evaluating Risks to the Schedule, Boehm's Top 10 Risks and Countermeasures, Applying the PERT Technique, Monte Carlo Simulation, Critical Chain Concepts.	
	Resource Allocation:	Introduction, Nature of Resources, Identifying Resource Requirements, Scheduling Resources, Creating Critical Paths, Counting the Cost, Being Specific, Publishing the Resource Schedule, Cost Schedules, Scheduling Sequence.	

IV	Monitoring and Control:	Introduction, Creating the Framework, Collecting the Data, Review, Visualizing Progress, Cost Monitoring, Earned Value Analysis, Prioritizing Monitoring, Getting the Project, Back to Target, Change Control, Software Configuration Management(SCM).	12
	Managing Contracts:	Introduction, Types of Contract, Stages in Contract Placement, Typical Terms of a Contract, Contract Management, Acceptance.	
	Managing People in Software Environments:	Introduction, Understanding Behaviour, Organizational Behavior: A Background, Selecting the Right Person for the Job, Instruction in the Best Methods, Motivation, The Oldham–Hackman Job Characteristics Model, Stress, Stress Management, Health and Safety, Some Ethical and Professional Concerns.	
V	Working in Teams:	Introduction, becoming a Team, Decision Making, Organization and Team Structures, Coordination Dependencies, Dispersed and Virtual Teams, Communication Genres, Communication Plans, Leadership.	12
	Software Quality:	Introduction, The Place of Software Quality in Project Planning, Importance of Software Quality, Defining Software Quality, Software Quality Models, ISO 9126, Product and Process Metrics, Product versus Process Quality Management, Quality Management Systems, Process Capability Models, Techniques to Help Enhance Software Quality, Testing, Software Reliability, Quality Plans.	
	Project Closeout:	Introduction, Reasons for Project Closure, Project Closure Process, Performing a Financial Closure, Project Closeout Report.	

#### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Software Project Management	Bob Hughes, Ike Cotterell, Rajib Mall	TMH	6th	2018
2.	Project Management and Tools & Technologies –An overview	Shailesh Mehta	SPD	1st	2017
3.	Software Project Management	Walker Royce	Pearson		2005

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks)	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

#### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numerical based questions, single line answers, etc., will contain internal options.
3. Refer **appendix I** for Paper pattern.



Name of the Course	<b>Project Dissertation Practical</b>
Course Code (refer to student handbook)	<b>USIT5P1</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>V</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	<b>Skill Enhancement Course</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

**Nomenclature:** Project Dissertation Practical

**Course Outcomes:**

On the successful completion of this course, the learner will be able to

1. Understand the current state of the project management profession
2. Apply project management tools and techniques
3. Understand project management terminology.
4. Explore the appropriate methods to initiate, plan, execute, control and close projects

**Refer Appendix II**

**Evaluation Pattern**

**A. Continuous Internal Evaluation (20 Marks):**

<b>Method</b>	<b>Marks</b>
Project Documentation	10
Attendance and active participation in Laboratory	10

**B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):**

<b>Question No</b>	<b>Unit</b>	<b>Marks</b>
<b>1</b>	Project idea	<b>05</b>
<b>2</b>	Diagrams	<b>05</b>
<b>3</b>	Database and Normalization	<b>05</b>
<b>4</b>	Presentation	<b>05</b>
<b>5</b>	Documentation	<b>05</b>
<b>6</b>	Viva	<b>05</b>

Name of the Course	<b>Internet of Things</b>
Course Code (refer to student handbook)	<b>USIT502</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>V</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Skill Enhancement Course</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<p><b>Employability:</b>  <b>Increasing Demand:</b> The IoT has created a surge in demand for skilled professionals who can design, develop, and manage IoT systems. Industries such as manufacturing, healthcare, transportation, and smart cities require IoT experts to implement and maintain their connected devices.</p> <p><b>Entrepreneurship:</b>  <b>Innovation Opportunities:</b> IoT provides a fertile ground for entrepreneurs to develop innovative products and services. Entrepreneurs can leverage IoT technologies to create smart devices, automation solutions, and data-driven platforms.</p> <p><b>Start-up Ecosystem:</b> The IoT has fueled the growth of start-up ecosystems worldwide. Entrepreneurs can access resources such as funding, mentorship, and collaborative spaces focused on IoT innovation.</p>

## Nomenclature: Internet of Things

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand IoT Technology.
2. Gain Knowledge of designing principles.
3. Familiar with IoT Sensors and Actuators.
4. Design and Develop IoT Systems
5. Knowledge of IoT Security and Privacy
6. Know Ethical and Social Implications of IoT
7. Think from prototyping to manufacturing of IoT.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	The Internet of Things: An Overview:	The Flavour of the Internet of Things, The “Internet” of “Things”, The Technology of the Internet of Things, Enchanted Objects, Who is Making the Internet of Things?	12
	Design Principles for Connected Devices:	Calm and Ambient Technology, Magic as Metaphor, Privacy, Keeping Secrets, Whose Data Is It Anyway? Web Thinking for Connected Devices, SmallPieces, Loosely Joined, First-Class Citizens on The Internet, Graceful Degradation, Affordances.	
	Internet Principles:	Internet Communications: An Overview, IP, TCP, The IP Protocol Suite (TCP/IP), UDP, IP Addresses, DNS, Static IP Address Assignment, Dynamic IP Address Assignment, IPv6, MAC Addresses, TCP and UDP Ports, An Example: HTTP Ports, Other Common Ports, Application Layer Protocols, HTTP, HTTPS: Encrypted HTTP, Other Application Layer Protocols.	
II	Thinking About Prototyping:	Sketching, Familiarity, Costs versus Ease of Prototyping, Prototypes and Production, Changing Embedded Platform, Physical Prototypes and Mass Personalisation, climbing into the Cloud, Open Source versus Closed Source, Why Closed? Why Open? Mixing Open and Closed Source, Closed Source for Mass Market Projects, Tapping into the Community. Electronics, Sensors, Actuators, Scaling Up the Electronics, Embedded Computing Basics, Microcontrollers, System-on-Chips, Choosing Your Platform, Arduino, developing on the Arduino, Some	
	Prototyping Embedded Devices:		

		Notes on the Hardware, Openness, Raspberry Pi, Cases and Extension Boards, Developing on the Raspberry Pi, Some Notes on the Hardware, Openness.	
III	Prototyping the Physical Design:	Preparation, Sketch, Iterate, and Explore, Non Digital Methods, Laser Cutting, Choosing a Laser Cutter, Software, Hinges and Joints, 3D Printing, Types of 3D Printing, Software, CNC Milling, Repurposing/Recycling.	12
	Prototyping Online Components:	Getting Started with an API, Mashing Up APIs, Scraping, Legalities, writing a New API, Clockodillo, Security, implementing the API, Using Curl to Test, Going Further, Real-Time Reactions, Polling, Comet, Other Protocols, MQ Telemetry Transport, Extensible Messaging and Presence Protocol, Constrained Application Protocol.	
IV	Techniques for Writing Embedded Code:	Memory Management, Types of Memory, Making the Most of Your RAM, Performance and Battery Life, Libraries, Debugging.	12
	Business Models:	A Short History of Business Models, Space and Time, From Craft to Mass Production, The Long Tail of the Internet, Learning from History, The Business Model Canvas, Who Is the Business Model For? Models, Make Thing, Sell Thing, Subscriptions, Customisation, be a Key Resource, Provide Infrastructure: Sensor Networks, take a Percentage, Funding an Internet of Things Startup, Hobby Projects and Open Source, Venture Capital, Government Funding, Crowd funding, Lean Startups.	
V	Moving to Manufacture:	What Are You Producing? Designing Kits, Designing Printed circuit boards, Software Choices, The Design Process, Manufacturing Printed Circuit Boards, Etching Boards, Milling Boards. Assembly, Testing Mass-Producing the Case and Other Fixtures, Certification, Costs, Scaling Up Software, Deployment, correctness and Maintainability, Security, Performance, User Community.	12
	Ethics:	Characterizing the Internet of Things, Privacy, Control, Disrupting Control, Crowdsourcing, Environment, Physical Thing, Electronics, Internet Service, Solutions, The Internet of Things as Part of the Solution, Cautious Optimism, The Open Internet of Things Definition.	

**Learning Resources recommended:**

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Designing the Internet of Things	Adrian McEwen, Hakim Cassimally	WILEY	First	2014
2.	Internet of Things – Architecture and Design	Raj Kamal	McGraw Hill	First	2017
3.	Getting Started with the Internet of Things	Cuno Pfister	O'Reilly	Sixth	2018
4.	Getting Started with Raspberry Pi	Matt Richardson and Shawn Wallace	SPD	Third	2016

**Evaluation Pattern****A. Continuous Internal Evaluation (40 Marks):**

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

**B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):**

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

**Guidelines for paper pattern for Semester End Evaluation:**

- All questions will be compulsory.
- Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
- Refer **appendix I** for Paper pattern.

Name of the Course	<b>Internet of Things Practical</b>
Course Code (refer to student handbook)	<b>USIT5P2</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>V</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	<b>Skill Enhancement Course</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

## Nomenclature: Internet of Things Practical

### Course Outcomes

Learner should be able to

1. Prepare and make use of embedded development platforms.
2. Interface various sensors and actuators with a development platform.
3. Design and Develop IoT Systems.

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 0	Starting Raspbian OS, Familiarising with Raspberry Pi Components and interface, Connecting to ethernet, Monitor, USB.	3
2	Practical 1	Displaying different LED patterns with Raspberry Pi.	3
3	Practical 2	Displaying Time over 4-Digit 7-Segment Display using Raspberry Pi	3
4	Practical 3	Raspberry Pi Based Oscilloscope	3
5	Practical 4	Controlling Raspberry Pi with WhatsApp.	3
6	Practical 5	Setting up Wireless Access Point using Raspberry Pi	3
7	Practical 6	Fingerprint Sensor interfacing with Raspberry Pi	3
8	Practical 7	Raspberry Pi GPS Module Interfacing	3
9	Practical 8	IoT based Web Controlled Home Automation using Raspberry Pi	3
10	Practical 9	Visitor Monitoring with Raspberry Pi and Pi Camera	3
11	Practical 10	Interfacing Raspberry Pi with RFID.	3
12	Practical 11	Building Google Assistant with Raspberry Pi.	3
13	Practical 12	Installing Windows 10 IoT Core on Raspberry Pi	3

This is sample Practical list. Course instructor may change the practical as per syllabus.



**Learning Resources recommended:**

<b>Sr. No.</b>	<b>Title</b>	<b>Author/s</b>	<b>Publisher</b>	<b>Edition</b>	<b>Year</b>
1.	Getting Started with the Internet of Things	Cuno Pfister	O'Reilly	Sixth	2018
2.	Getting Started with Raspberry Pi	Matt Richardson and Shawn Wallace	SPD	Third	2016

**Evaluation Pattern****A. Continuous Internal Evaluation (20 Marks):**

<b>Method</b>	<b>Marks</b>
Certified Journal	10
Attendance and active participation in Laboratory	10

**B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):**

<b>Question No</b>	<b>Unit</b>	<b>Marks</b>
1	Program 1	10
2	Program 2	15
3	Viva	05

Name of the Course	<b>Advanced Web Programming</b>
Course Code (refer to student handbook)	<b>USIT503</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>V</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Skill Enhancement Course</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<p><b>Employability:</b> There is a continuous demand for professionals with advanced web programming skills due to the increasing reliance on web-based applications and services. Employers seek individuals who can develop complex and interactive web solutions using advanced programming languages and frameworks.</p> <p><b>Specialized Roles:</b> Advanced web programming opens up opportunities for specialized roles such as full-stack developers, front-end developers, back-end developers, and web application architects. <b>Entrepreneurship:</b> Advanced web programming skills empower entrepreneurs to create innovative and customized web solutions for their businesses or clients. By leveraging cutting-edge technologies and frameworks, entrepreneurs can develop unique web applications tailored to specific needs.</p>

## Nomenclature: Advanced Web Programming

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Be proficient in Advanced Web Technologies.
2. Be Advanced Front-End Developer Skills.
3. Be Back-End Developer.
4. Be a Full-Stack Web Developer.
5. Integration Third-Party APIs and Services.
6. Test and Debug Web Applications.
7. Do Web Application Security.
8. Have Performance Optimization and Web Analytics.
9. Perform Collaboration and Project Management Skills.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	Introducing .NET:	The .NET Framework, C#, VB, and the .NET Languages, The Common Language Runtime, The .NET Class Library.	12
	The C# Language:	C# Language Basics, Variables and Data Types, Variable Operations, Object-Based Manipulation, Conditional Logic, Loops, Methods.	
	Types, Objects, and Namespaces:	The Basics About Classes, Building a Basic Class, Value Types and Reference Types, Understanding Namespaces and Assemblies, Advanced Class Programming.	
II	Web Form Fundamentals:	Writing Code, Using the Code-Behind Class, Adding Event Handlers, Understanding the Anatomy of an ASP.NET Application, Introducing Server Controls, Using the Page Class, Using Application Events, Configuring an ASP.NET Application.	12
	Form Controls:	Stepping Up to Web Controls, Web Control Classes, List Controls, Table Controls, Web Control Events and AutoPostBack, Validation, Understanding Validation, Using the Validation Controls, Rich Controls, The Calendar, The AdRotator, Pages with Multiple Views, User Controls and Graphics, User Controls, Dynamic Graphics, The Chart Control, Website Navigation: Site Maps, URL Mapping and Routing, The SiteMapPath Control, The TreeView Control, The Menu Control.	

III	Error Handling, Logging, and Tracing:	Avoiding Common Errors, Understanding Exception Handling, Handling Exceptions, Throwing Your Own Exceptions, Using Page Tracing	12
	State Management:	Understanding the Problem of State, Using View State, Transferring Information Between Pages, Using Cookies, Managing Session State, Configuring Session State, Using Application State, Comparing State Management Options	
	Styles, Themes, and Master pages:	Styles, Themes, Master Page Basics, Advanced Master Pages,	
IV	ADO.NET Fundamentals:	Understanding Databases, Configuring Your Database, Understanding SQL Basics, Understanding the Data Provider Model, Using Direct Data Access, Using Disconnected Data Access.	12
	Data Binding:	Introducing Data Binding, Using Single-Value Data Binding, Using Repeated-Value Data Binding, Working with Data Source Controls,	
	The Data Controls:	The GridView, Formatting the GridView, selecting a GridView Row, Editing with the GridView, Sorting and Paging the GridView, Using GridView Templates, The DetailsView and FormView	
V	XML:	XML Explained, The XML Classes, XML Validation, XML Display and Transforms.	12
	Security Fundamentals:	Understanding Security Requirements, Authentication and Authorization, Forms Authentication, Windows Authentication.	
	ASP.NET AJAX:	Understanding Ajax, Using Partial Refreshes, Using Progress Notification, Implementing Timed Refreshes, Working with the ASP.NET AJAX Control Toolkit.	

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Beginning ASP.NET 4.5 in C#	Matthew MacDonald	Apress		2012
2.	C# 2015	Anne Bohem and Joel Murach	Murach	Third	2016
3.	Murach's ASP.NET 4.6 Web Programming in C#2015	Mary Delamater and Anne Bohem	SPD	Sixth	2016
4.	ASP.NET 4.0 programming	J. Kanjilal	Tata McGraw-Hill		2011
5.	Programming ASP.NET	D.Esposito	Microsoft Press (Dreamtech)		2011
6.	Beginning Visual C# 2010	K. Watson, C. Nagel, J.H Padderson, J.D. Reid, M.Skinner	Wrox (Wiley)		2010

### Evaluation Pattern

#### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

**B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):**

<b>Question No</b>	<b>Unit</b>	<b>Marks</b>
<b>1</b>	<b>I</b>	<b>12</b>
<b>2</b>	<b>II</b>	<b>12</b>
<b>3</b>	<b>III</b>	<b>12</b>
<b>4</b>	<b>IV</b>	<b>12</b>
<b>5</b>	<b>V</b>	<b>12</b>

**Guidelines for paper pattern for Semester End Evaluation:**

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
  3. Refer **appendix I** for Paper pattern.
-

Name of the Course	<b>Advanced Web Programming Practical</b>
Course Code (refer to student handbook)	<b>USIT5P3</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>V</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	<b>Skill Enhancement Course</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<p><b>employability/ entrepreneurship/ skill development</b></p> <p><b>Entrepreneurship:</b></p> <p><b>E-commerce and Online Services: With the growing dominance of e-commerce and online services, entrepreneurs with advanced web programming skills can build feature-rich online stores, payment gateways, booking platforms, and other web-based services. This opens up opportunities to monetize digital products and create scalable online businesses.</b></p>

## Nomenclature: Advanced Web Programming Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Be proficient in Advanced Web Technologies.
2. Be Advanced Front-End Developer Skills.
3. Be Back-End Developer.
4. Be a Full-Stack Web Developer.
5. Integration Third-Party APIs and Services.
6. Test and Debug Web Applications.
7. Do Web Application Security.
8. Have Performance Optimization and Web Analytics.
9. Perform Collaboration and Project Management Skills.

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 1: Working with basic C# and ASP .NET	<ol style="list-style-type: none"><li>a. Create an application that obtains four int values from the user and displays the product.</li><li>b. Create an application to demonstrate string operations.</li><li>c. Create an application that receives the (Student Id, Student Name, Course Name, Date of Birth) information from a set of students. The application should also display the information of all the students once the data entered.</li><li>d. Create an application to demonstrate following operations<ol style="list-style-type: none"><li>i. Generate Fibonacci series.</li><li>ii. Test for prime numbers.</li><li>iii. Test for vowels.</li><li>iv. Use of foreach loop with arrays</li><li>v. Reverse a number and find sum of digits of a number.</li></ol></li></ol>	3
2	Practical 2: Working with Object Oriented C# and ASP .NET	<ol style="list-style-type: none"><li>a. Create simple application to perform following operations<ol style="list-style-type: none"><li>i. Finding factorial Value</li><li>ii. Money Conversion</li><li>iii. Quadratic Equation</li><li>iv. Temperature Conversion</li></ol></li><li>b. Create simple application to demonstrate use of following concepts<ol style="list-style-type: none"><li>i. Function Overloading</li><li>ii. Inheritance (all types)</li><li>iii. Constructor overloading</li><li>iv. Interfaces</li></ol></li><li>c. Create simple application to demonstrate use of</li></ol>	3



		<p>following concepts</p> <p>i. Using Delegates and events ii. Exception handling</p>	
3	Practical 3: Working with Web Forms and Controls	<p>a. Create a simple web page with various server controls to demonstrate setting and use of their properties. (Example : AutoPostBack)</p> <p>b. Demonstrate the use of Calendar control to perform following operations.</p> <p>a) Display messages in a calendar control b) Display vacation in a calendar control</p> <p>c) Selected day in a calendar control using style d) Difference between two calendar dates</p> <p>c. Demonstrate the use of Treeview control perform following operations.</p> <p>a) Treeview control and datalist b) Treeview operations</p>	3
4	Practical 4: Working with Form Controls	<p>a. Create a Registration form to demonstrate use of various Validation controls.</p> <p>b. Create Web Form to demonstrate use of Adrotator Control.</p> <p>c. Create Web Form to demonstrate use User Control.</p>	3
5	Practical 5: Working with Navigation, Beautification and Master page.	<p>a. Create Web Form to demonstrate use of Website Navigation controls and Site Map.</p> <p>b. Create a web application to demonstrate use of Master Page with applying Styles and Themes for page beautification.</p> <p>c. Create a web application to demonstrate various states of ASP.NET Pages.</p>	3
6	Practical 6: Working with Database	<p>a. Create a web application bind data in a multiline textbox by querying in another textbox.</p> <p>b. Create a web application to display records by using database.</p> <p>c. Demonstrate the use of Datalist link control.</p>	3
7	Practical 7: Working with Database	<p>a. Create a web application to display Databinding using dropdownlist control.</p> <p>b. Create a web application for to display the phone no of an author using database.</p> <p>c. Create a web application for inserting and deleting record from a database. (Using Execute-Non Query).</p>	3

8	Practical 8: Working with data controls	a. Create a web application to demonstrate various uses and properties of SqlDataSource. b. Create a web application to demonstrate data binding using DetailsView and FormView Control. c. Create a web application to display Using Disconnected Data Access and Databinding using GridView.	3
9	Practical 9 Working with GridView control	a. Create a web application to demonstrate use of GridView control template and GridView hyperlink. b. Create a web application to demonstrate use of GridView button column and GridView events. c. Create a web application to demonstrate GridView paging and Creating own table format using GridView.	3
10	Practical 10 Working with AJAX and XML	a. Create a web application to demonstrate reading and writing operations with XML. b. Create a web application to demonstrate Form Security and Windows Security with proper Authentication and Authorization properties. c. Create a web application to demonstrate use of various Ajax controls.	3
11	Practical 11 Programs to create and use DLL	Programs to create and use DLL	3

This is sample Practical list. Course instructor may change the practical as per syllabus.

#### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Beginning ASP.NET 4.5 in C#	Matthew MacDonald	Apress		2012
2.	C# 2015	Anne Bohem and Joel Murach	Murach	Third	2016
3.	Murach's ASP.NET 4.6 Web Programming in C#2015	Mary Delamater and Anne Bohem	SPD	Sixth	2016
4.	ASP.NET 4.0 programming	J. Kanjilal	Tata McGraw-Hill		2011

## Evaluation Pattern

### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

---

Name of the Course	<b>Artificial Intelligence</b>
Course Code (refer to student handbook)	<b>USIT504</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>V</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Discipline Specific Elective</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

## Nomenclature: Artificial Intelligence

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Know History of AI
2. Solve Problems and Design Algorithms.
3. Understand Machine Learning Techniques
4. Basics of AI Programming.
5. Have Knowledge of Ethical and Social Implications.
6. Aware about Current Trends and Future Directions.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	<b>Introduction:</b>	What is Artificial Intelligence? Foundations of AI, history, the state of art AI today.	12
	<b>Intelligent Agents:</b>	agents and environment, good behavior, nature of environment, the structure of agents.	
II	<b>Solving Problems by Searching:</b>	Problem solving agents, examples problems, searching for solutions, uninformed search, informed search strategies, heuristic functions.	12
	<b>Beyond Classical Search:</b>	local search algorithms, searching with non-deterministic action, searching with partial observations, online search agents and unknown environments.	
III	<b>Adversarial Search:</b>	Games, optimal decisions in games, alpha-beta pruning, stochastic games, partially observable games, state-of-the-art game programs.	12
	<b>Logical Agents:</b>	Knowledge base agents, The Wumpus world, logic, propositional logic, propositional theorem proving, effective propositional model checking, agents based on propositional logic.	
IV	<b>First Order Logic:</b>	Syntax and semantics, using First Order Logic, Knowledge engineering in First Order Logic.	12
	<b>Inference in First Order Logic:</b>	propositional vs. First Order, unification and lifting, forward and backward chaining, resolution.	

V	<b>Planning:</b>	Definition of Classical Planning, Algorithms for planning as state space search, planning graphs, other classical planning approaches, analysis of planning approaches, Time, Schedules and resources, hierarchical planning, Planning and Acting in Nondeterministic Domains, multiagent planning,	12
	<b>Knowledge Representation:</b>	Categories and Objects, events, mental events and objects, reasoning systems for categories, reasoning with default information, Internet shopping world	

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Artificial Intelligence: A Modern Approach	Stuart Russel and Peter Norvig	Pearson	3rd	2015
2	A First Course in Artificial Intelligence	Deepak Khemani	TMH	First	2017
3	Artificial Intelligence: A Rational Approach	Rahul Deva	Shroff publishers	1st	2018
4	Artificial Intelligence	Elaine Rich, Kevin Knight and Shivashankar Nair	TMH	3rd	2009
5	Artificial Intelligence & Soft Computing for Beginners	Anandita Das Bhattacharjee	SPD	1st	2013

### Evaluation Pattern

#### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks)	20
Assignments	10
Attendance and active participation in classroom	10

**B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):**

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

**Guidelines for paper pattern for Semester End Evaluation:**

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
  3. Refer **appendix I** for Paper pattern.
-

Name of the Course	<b>Artificial Intelligence Practical</b>
Course Code (refer to student handbook)	<b>USIT5P4</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>V</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	<b>Discipline Specific Elective</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>



## Nomenclature: Artificial Intelligence Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

- Understand AI Fundamentals
- Have Problem Solving and Algorithm Design Skills
- Generate Machine Learning Techniques
- Develop AI Programming
- Solve Ethical and Social Implications
- Have AI System Development Skills
- Focus on Current Trends and Future Directions

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 1	a Write a program to implement depth first search algorithm. b Write a program to implement breadth first search algorithm.	3
2	Practical 2	a .Write a program to simulate 4-Queen / N-Queen problem. b Write a program to solve tower of Hanoi problem.	3
3	Practical 3	a. Write a program to implement alpha beta search. b Write a program for Hill climbing problem.	3
4	Practical 4	a Write a program to implement A* algorithm. b Write a program to implement AO* algorithm.	3
5	Practical 5	a Write a program to solve water jug problem. b Design the simulation of tic – tac – toe game using min-max algorithm.	3
6	Practical 6	a Write a program to solve Missionaries and Cannibals problem. b Design an application to simulate number puzzle problem.	3

7	Practical 7	a Write a program to shuffle Deck of cards. b Solve traveling salesman problem using artificial intelligence technique.	3
8	Practical 8	a Solve the block of World problem. b Solve constraint satisfaction problem	3
9	Practical 9	a Derive the expressions based on Associative law b Derive the expressions based on Distributive law	3
10	Practical 10	a Write a program to derive the predicate. (for e.g.: Sachin is batsman , batsman is cricketer) - > Sachin is Cricketer. b Write a program which contains three predicates: male, female, parent. Make rules for following family relations: father, mother, grandfather, grandmother, brother, sister, uncle, aunt, nephew and niece, cousin. Question: i. Draw Family Tree. ii. Define: Clauses, Facts, Predicates and Rules with conjunction and disjunction	3

This is sample Practical list. Course instructor may change the practical as per syllabus.

#### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Artificial Intelligence: A Modern Approach	Stuart Russel and Peter Norvig	Pearson	3rd	2015
2	A First Course in Artificial Intelligence	Deepak Khemani	TMH	First	2017
3	Artificial Intelligence: A Rational Approach	Rahul Deva	Shroff publishers	1st	2018

## Evaluation Pattern

### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

Name of the Course	<b>Linux System Administration</b>
Course Code (refer to student handbook)	USIT505
Class	T.Y.BSc.IT
Semester	V
No of Credits	2
Nature	Theory
Type (applicable to NEP only)	Discipline Specific Elective
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Linux System Administration

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand Linux Operating System
2. Gain knowledge of Installation and Configuration
3. Do System Maintenance and Updates
4. Have User and Group Management
5. Do Networking and Security Tasks
6. Perform System Monitoring and Performance Tuning
7. Troubleshoot and Resolve Problems.

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	Introduction to Red Hat Enterprise Linux:	Linux, Open Source and Red Hat, Origins of Linux, Distributions, Duties of Linux System Administrator.	12
	Command Line:	Working with the Bash Shell, Getting the Best of Bash, Useful Bash Key Sequences, Working with Bash History, Performing Basic File System Management Tasks, Working with Directories, Piping and Redirection, Finding Files	
	System Administration Tasks:	Performing Job Management Tasks, System and Process Monitoring and Management, Managing Processes with ps, Sending Signals to Processes with the kill Command, using top to Show Current System Activity, Managing Process Niceness, Scheduling Jobs, Mounting Devices, Working with Links, Creating Backups, Managing Printers, Setting Up System	
	Managing Software:	Understanding RPM, Understanding Meta Package Handlers, Creating Your Own Repositories, Managing Repositories, Installing Software with Yum, Querying Software, Extracting Files from RPM Packages	

II	Configuring and Managing Storage:	Understanding Partitions and Logical Volumes, Creating Partitions, Creating File Systems, File Systems Overview, Creating File Systems, Changing File System Properties, Checking the File System Integrity, Mounting File Systems Automatically Through fstab, Working with Logical Volumes, Creating Logical Volumes, Resizing Logical Volumes, Working with Snapshots, Replacing Failing Storage Devices, Creating Swap Space, Working with Encrypted Volumes	12
	Connecting to the Network:	Understanding NetworkManager, Working with Services and Runlevels, Configuring the Network with NetworkManager, Working with system-config-network, NetworkManager Configuration Files, Network Service Scripts, Networking from the Command Line, Troubleshooting Networking, Setting Up IPv6, Configuring SSH, Enabling the SSH Server, Using the SSH Client, Using PuTTY on Windows Machines, Configuring Key-Based SSH Authentication, Using Graphical Applications with SSH, Using SSH Port Forwarding, Configuring VNC Server Access	
	Working with Users, Groups, and Permissions:	Permissions: Managing Users and Groups, Commands for User Management, Managing Passwords, Modifying and Deleting User Accounts, Configuration Files, Creating Groups, Using Graphical Tools for User, and Group Management, Using External Authentication Sources, the Authentication Process, sssd, nsswitch, Pluggable Authentication Modules, Managing Permissions, the Role of Ownership, Basic Permissions: Read, Write, and Execute, Advanced Permissions, Working with Access Control Lists, Setting Default Permissions with umask, Working with Attributes	
III	Securing Server with iptables:	Understanding Firewalls, Setting Up a Firewall with system-config-firewall, Allowing Services, Trusted Interfaces, Masquerading, Configuration Files, Setting Up a Firewall with iptables, Tables, Chains, and Rules, Composition of Rule, Configuration Example, Advanced iptables Configuration, Configuring Logging, The Limit Module, Configuring NAT	12
	Setting Up Cryptographic Services:	Introducing SSL, Proof of Authenticity: The Certificate Authority, Managing Certificates with openssl, Creating a Signing Request, Working with GNU Privacy Guard, Creating GPG Keys, Key Transfer, Managing GPG Keys,	

		Encrypting Files with GPG, GPG Signing, Signing RPM Files	
	Configuring Server for File Sharing:	What is NFS? Advantages and Disadvantages of NFS, Configuring NFS4, Setting Up NFSv4, Mounting an NFS Share, Making NFS Mounts Persistent, Configuring Automount, Configuring Samba, Setting Up a Samba File Server, Samba Advanced Authentication Options, Accessing Samba Shares, Offering FTP Services.	
IV	Configuring DNS and DHCP:	Introduction to DNS, The DNS Hierarchy, DNS Server Types, The DNS Lookup Process, DNS Zone Types, Setting Up a DNS Server, Setting Up a Cache-Only Name Server, Setting Up a Primary Name Server, Setting Up a Secondary Name Server, Understanding DHCP, Setting Up a DHCP Server	12
	Setting Up a Mail Server:	Using the Message Transfer Agent, the Mail Delivery Agent, the Mail User Agent, Setting Up Postfix as an SMTP Server, Working with Mutt, Basic Configuration, Internet Configuration, Configuring Dovecot for POP and IMAP	
	Configuring Apache on Red Hat Enterprise Linux:	Configuring the Apache Web Server, creating a Basic Website, Understanding the Apache Configuration Files, Apache Log Files, Working with Virtual Hosts, Securing the Web Server with TLS Certificates, Configuring Authentication, Setting Up Authentication with .htpasswd, Configuring LDAP Authentication, Setting Up MySQL	
V	Introducing Bash Shell Scripting:	Introduction, Elements of a Good Shell Script, Executing the Script, Working with Variables and Input, Understanding Variables, Variables, Subshells, and Sourcing, Working with Script Arguments, Asking for Input, Using Command Substitution, Substitution Operators, Changing Variable Content with Pattern Matching, Performing Calculations, Using Control Structures, Using if...then...else, Using case, Using while, Using until, Using for, Configuring booting with GRUB.	12
	High-Availability Clustering:	High-Availability Clustering, The Workings of High Availability, High-Availability Requirements, Red Hat High-Availability Add-on Software, Components, Configuring Cluster-Based Services, Setting Up Bonding, Setting Up Shared Storage, Installing the Red Hat High Availability Add-On, Building the	

		Initial State of the Cluster, Configuring Additional Cluster Properties, Configuring a Quorum Disk, Setting Up Fencing, Creating Resources and Services, Troubleshooting a Nonoperational Cluster, Configuring GFS2 File Systems	
	Setting Up an Installation Server:	Configuring a Network Server as an Installation Server, Setting Up a TFTP and DHCP Server for PXE Boot, Installing the TFTP Server, Configuring DHCP for PXE Boot, Creating the TFTP PXE Server Content, creating a Kickstart File, Using a Kickstart File to Perform an Automated, Installation, Modifying the Kickstart File with, system-config-kickstart, Making Manual Modifications to the Kickstart File	

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Red Hat Enterprise Linux 6 Administration	Sander van Vugt	John Wiley and Sons		2013
2	Red hat Linux Networking and System Administration	Terry Collings and Kurt Wall	Wiley	3rd	
3	Linux Administration: A Beginner's Guide	Wale Soyinka	TMH	Fifth Edition	



## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

#### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
  3. Refer **appendix I** for Paper pattern.
-

Name of the Course	<b>Linux Administration Practical</b>
Course Code (refer to student handbook)	<b>USIT5P5</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>V</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	<b>Discipline Specific Elective</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

## Nomenclature: Linux Administration Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand Linux Operating System
2. Gain knowledge of Installation and Configuration
3. Do System Maintenance and Updates
4. Have User and Group Management
5. Do Networking and Security Tasks
6. Perform System Monitoring and Performance Tuning
7. Troubleshoot and Resolve Problems.

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 0	Installation of RHEL 6.X	3
2	Practical 1	Graphical User Interface and Command Line Interface and Processes a Exploring the Graphical Desktop b The Command Line Interface c Managing Processes	3
3	Practical 2	Storage Devices and Links, Backup and Repository b Working with Storage Devices and Links a Making a Backup b Creating a Repository	3
4	Practical 3	Working with RPMsm Storage and Networking a Using Query Options b Extracting Files From RPMs c Configuring and Managing Storage d Connecting to the Network	3
5	Practical 4	Working with Users, Groups, and Permissions	3
6	Practical 5	Firewall and Cryptographic services a Securing Server with iptables b Setting Up Cryptographic Services	3
7	Practical 6	Configuring Server for File Sharing a Configuring NFS Server and Client b Configuring Samba c Configuring FTP	3

8	Practical 7	DNS, DHCP and Mail Server a Configuring DNS b Configuring DHCP c Setting Up a Mail Server	3
9	Practical 8	Web Server a Configuring Apache on Red Hat Enterprise Linux b Writing a Script to Monitor Activity on the Apache Web Server c Using the select Command	3
10	Practical 9	Shell Scripts and High-Availability Clustering a Writing Shell Scripts b Configuring Booting with GRUB c Configuring High Availability Clustering	3
11	Practical 10	Setting Up an Installation Server a Configuring Network Server as an Installation Server b Setting Up a TFTP and DHCP Server for PXE Boot	

This is sample Practical list. Course instructor may change the practical as per syllabus.

#### Learning Resources recommended:

Sr. No	Title	Author/s	Publisher	Edition	Year
1	Red Hat Enterprise Linux 6 Administration	Sander van Vugt	John Wiley and Sons		2013
2	Red hat Linux Networking and System Administration	Terry Collings and Kurt Wall	Wiley	3rd	
3	Linux Administration: A Beginner's Guide	Wale Soyinka	TMH	Fifth Edition	

## Evaluation Pattern

### A. Internal Evaluation

#### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

#### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

---

Name of the Course	<b>Enterprise Java</b>
Course Code (refer to student handbook)	USIT506
Class	T.Y.B.Sc.IT
Semester	V
No of Credits	2
Nature	<b>Theory</b>
Type (applicable to NEP only)	Discipline Specific Elective
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	

## Nomenclature: Enterprise Java

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand Java Enterprise Edition (Java EE)
2. Develop enterprise-level applications
3. Master Java EE APIs
4. Implement database connectivity
5. Build user interfaces
6. Deploy and manage Java EE applications

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	Understanding Java EE	What is an Enterprise Application? What is java enterprise edition? Java EE Technologies, Java EE evolution, Glassfish server	12
	Java EE Architecture, Server and Containers	Types of System Architecture, Java EE Server, Java EE Containers.	
	Introduction to Java Servlets	The Need for Dynamic Content, Java Servlet Technology, Why Servlets? What can Servlets do?	
	Servlet API and Lifecycle	Java Servlet API, The Servlet Skeleton, The Servlet Life Cycle, A Simple Welcome Servlet	
	Working with Servlets	Getting Started, Using Annotations Instead of Deployment Descriptor.	
	Working with Databases	What Is JDBC? JDBC Architecture, Accessing Database, The Servlet GUI and Database Example.	
II	Request Dispatcher	Requestdispatcher Interface, Methods of Requestdispatcher, Requestdispatcher Application.	12
	COOKIES	Kinds of Cookies, Where Cookies Are Used? Creating Cookies Using Servlet, Dynamically Changing the Colors of A Page	

	SESSION	What Are Sessions? Lifecycle of Http Session, Session Tracking With Servlet API, A Servlet Session Example	
	Working with Files	Uploading Files, Creating an Upload File Application, Downloading Files, Creating a Download File Application.	
	Working with Non-Blocking I/O	Creating a Non-Blocking Read Application, Creating The Web Application, Creating Java Class, Creating Servlets, Retrieving The File, Creating index.jsp	
III	Introduction To Java Server Pages	Why use Java Server Pages? Disadvantages Of JSP, JSP v\vs Servlets, Life Cycle of a JSP Page, How does a JSP function? How does JSP execute? About Java Server Pages	
	Getting Started With Java Server Pages	Comments, JSP Document, JSP Elements, JSP GUI Example.	
	Action Elements	Including other Files, Forwarding JSP Page to Another Page, Passing Parameters for other Actions, Loading a Javabean.	
	Implicit Objects, Scope and El Expressions	Implicit Objects, Character Quoting Conventions, Unified Expression Language [Unified El], Expression Language.	
	Java Server Pages Standard Tag Libraries	What is wrong in using JSP Scriptlet Tags? How JSTL Fixes JSP Scriptlet's Shortcomings? Disadvantages Of JSTL, Tag Libraries.	
IV	Introduction To Enterprise Javabeans	Enterprise Bean Architecture, Benefits of Enterprise Bean, Types of Enterprise Bean, Accessing Enterprise Beans, Enterprise Bean Application, Packaging Enterprise Beans	<b>12</b>
	Working with Session Beans	When to use Session Beans? Types of Session Beans, Remote and Local Interfaces, Accessing Interfaces, Lifecycle of Enterprise Beans, Packaging Enterprise Beans, Example of Stateful Session Bean, Example of Stateless Session Bean, Example of Singleton Session Beans.	
	Working with Message Driven Beans	Lifecycle of a Message Driven Bean, Uses of Message Driven Beans, The Message Driven Beans Example.	



	Interceptors	Request and Interceptor, Defining An Interceptor, AroundInvoke Method, Applying Interceptor, Adding An Interceptor To An Enterprise Bean, Build and Run the Web Application.	
	Java Naming and Directory Interface	What is Naming Service? What is Directory Service? What is Java Naming and Directory interface? Basic Lookup, JNDI Namespace in Java EE, Resources and JNDI, Datasource Resource Definition in Java EE.	
V	Persistence, Object/Relational Mapping And JPA	What is Persistence? Persistence in Java, Current Persistence Standards in Java, Why another Persistence Standards? Object/Relational Mapping	12
	Introduction to Java Persistence API	The Java Persistence API, JPA, ORM, Database and the application, Architecture of JPA, How JPA Works? JPA Specifications.	
	Writing JPA Application	Application Requirement Specifications, Software Requirements, The Application Development Approach, Creating Database and Tables in Mysql, creating a Web Application, Adding the Required Library Files, creating a Javabeen Class, Creating Persistence Unit [Persistence.Xml], Creating JSPS, The JPA Application Structure, Running the JPA Application.	
	Introduction to Hibernate	What is Hibernate? Why Hibernate? Hibernate, Database and The Application, Components of Hibernate, Architecture of Hibernate, How Hibernate Works?	
	Writing Hibernate Application	Application Requirement Specifications, Software Requirements, The Application Development Approach, Creating Database and Tables in Mysql, creating a Web Application, Adding the Required Library Files, creating a Javabeen Class, Creating Hibernate Configuration File, Adding a Mapping Class,Creating JSPS, Running The Hibernate Application.	

### Learning Resources recommended:

Sr.No.	Title	Author/s	Publisher	Edition	Year
1	Java EE 7 For Beginners	Sharanam Shah, Vaishali Shah	SPD	First	2017
2	Java EE 8 Cookbook: Build reliable applications with the most robust and mature technology for enterprise development	Elder Moraes	Packt	First	2018
3	Advanced Java Programming	Uttam Kumar Roy	Oxford Press		2015

### Evaluation Pattern

#### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

#### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

#### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
3. Refer **appendix I** for Paper pattern.

Name of the Course	<b>Enterprise Java Practical</b>
Course Code (refer to student handbook)	<b>USIT5P6</b>
Class	<b>T.Y.B.Sc.IT</b>
Semester	<b>V</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	Discipline Specific Elective
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

## Nomenclature: Enterprise Java Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

- Understand Java Enterprise Edition (Java EE)
- Develop enterprise-level applications
- Master Java EE APIs
- Implement database connectivity
- Build user interfaces
- Deploy and manage Java EE applications

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	<b>Practical 1:</b> Implement the following Simple Servlet applications.	a. Create a simple calculator application using servlet. b. Create a servlet for a login page. If the username and password are correct then it says message "Hello <username>" else a message "login failed" c. Create a registration servlet in Java using JDBC. Accept the details such as Username, Password, Email, and Country from the user using HTML Form and store the registration details in the database.	3+3
II	<b>Practical 2:</b> Implement the following Servlet applications with Cookies and Sessions.	a. Using Request Dispatcher Interface create a Servlet which will validate the password entered by the user, if the user has entered "Servlet" as password, then he will be forwarded to Welcome Servlet else the user will stay on the index.html page and an error message will be displayed. b. Create a servlet that uses Cookies to store the number of times a user has visited servlet. c. Create a servlet demonstrating the use of session creation and destruction. Also check whether the user has visited this page first time or has visited earlier also using sessions.	3+3

II	<b>Practical 3:</b> Implement the Servlet IO and File applications.	a. Create a Servlet application to upload and download a file. b. Develop Simple Servlet Question Answer Application using Database. c. Create simple Servlet application to demonstrate Non-Blocking Read Operation.	3+3
III	<b>Practical 4:</b> Implement the following JSP applications.	a. Develop a simple JSP application to display values obtained from the use of intrinsic objects of various types. b. Develop a simple JSP application to pass values from one page to another with validations. (Name-txt, age-txt, hobbies-checkbox, email-txt, gender-radio button). c. Create a registration and login JSP application to register and authenticate the user based on username and password using JDBC.	3+3+3
III	<b>Practical 5:</b> Implement the following JSP JSTL and EL Applications.	a. Create an html page with fields, eno, name, age, desg, salary. Now on submit this data to a JSP page which will update the employee table of database with matching eno. b. Create a JSP page to demonstrate the use of Expression language. c. Create a JSP application to demonstrate the use of JSTL.	3+3+3
IV	<b>Practical 6:</b> Implement the following EJB Applications.	a. Create a Currency Converter application using EJB. b. Develop a Simple Room Reservation System Application Using EJB. c. Develop simple shopping cart application using EJB [Stateful Session Bean].	3+3
IV	<b>Practical 7:</b> Implement the following EJB applications with different types of Beans.	a. Develop simple EJB application to demonstrate Servlet Hit count using Singleton Session Beans. b. Develop simple visitor Statistics application using Message Driven Bean [Stateless Session Bean]. c. Develop simple Marks Entry Application to demonstrate accessing Database using EJB.	3+3+3
V	<b>Practical 8:</b> Implement the following JPA applications.	a. Develop a simple Inventory Application Using JPA. b. Develop a Guestbook Application Using JPA. c. Create simple JPA application to store and retrieve Book details.	3+3

V	<b>Practical 9:</b> Implement the following JPA applications with ORM and Hibernate.	a. Develop a JPA Application to demonstrate use of ORM associations. b. Develop a Hibernate application to store Feedback of Website Visitor in MySQL Database. c. Develop a Hibernate application to store and retrieve employee details in MySQL Database.	3+3+3
V	<b>Practical 10:</b> Implement the following Hibernate applications.	a. Develop an application to demonstrate Hibernate One- To -One Mapping Using Annotation. b. Develop Hibernate application to enter and retrieve course details with ORM Mapping. c. Develop a five page web application site using any two or three Java EE Technologies.	3+3+3

This is sample Practical list. Course instructor may change the practical as per syllabus.

#### Learning Resources recommended:

Sr.No.	Title	Author/s	Publisher	Edition	Year
1	Java EE 7 For Beginners	Sharanam Shah, Vaishali Shah	SPD	First	2017
2	Java EE 8 Cookbook: Build reliable applications with the most robust and mature technology for enterprise development	Elder Moraes	Packt	First	2018

## Evaluation Pattern

### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

---

Name of the Course	<b>Next Generation Technologies</b>
Course Code (refer to student handbook)	<b>USIT507</b>
Class	<b>T.Y.B.Sc.IT</b>
Semester	<b>V</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>



## Nomenclature: Next Generation Technologies

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand emerging technologies
2. Analyze the impact of next-generation technologies
3. Explore applications of emerging technologies
4. Identify business opportunities
5. Stay updated on emerging trends

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	Big Data:	Getting Started, Big Data, Facts About Big Data, Big Data Sources, Three Vs of Big Data, Volume, Variety, Velocity, Usage of Big Data, Visibility, Discover and Analyze Information, Segmentation and Customizations, Aiding Decision Making, Innovation, Big Data Challenges, Policies and Procedures, Access to Data, Technology and Techniques, Legacy Systems and Big Data, Structure of Big Data, Data Storage, Data Processing, Big Data Technologies	12
	NoSQL:	SQL, NoSQL, Definition, A Brief History of NoSQL, ACID vs. BASE, CAP Theorem (Brewer's Theorem), The BASE, NoSQL Advantages and Disadvantages, Advantages of NoSQL, Disadvantages of NoSQL, SQL vs. NoSQL Databases, Categories of NoSQL Databases	
	Introducing MongoDB:	History, MongoDB Design Philosophy, Speed, Scalability, and Agility, Non-Relational Approach, JSON-Based Document Store, Performance vs. Features, Running the Database Anywhere, SQL Comparison	
II	The MongoDB Data Model:	The Data Model, JSON and BSON, The Identifier (_id), Capped Collection, Polymorphic Schemas, Object-Oriented Programming, Schema Evolution	12
	Using MongoDB Shell:	Basic Querying, Create and Insert, Explicitly Creating Collections, Inserting Documents Using Loop, Inserting by Explicitly Specifying _id, Update, Delete, Read, Using Indexes, Stepping Beyond the Basics, Using Conditional Operators, Regular Expressions, MapReduce, aggregate(), Designing an Application's Data Model, Relational Data Modeling and Normalization, MongoDB Document Data Model Approach	

	MongoDB Architecture :	Core Processes, mongod, mongo, mongos, MongoDB Tools, Standalone Deployment, Replication, Master/Slave Replication, Replica Set, Implementing Advanced Clustering with Replica Sets, Sharding, Sharding Components, Data Distribution Process, Data Balancing Process, Operations, Implementing Sharding, Controlling Collection Distribution (Tag-Based Sharding), Points to Remember When Importing Data in a Sharded Environment, Monitoring for Sharding, Monitoring the Config Servers, Production Cluster Architecture, Scenario 1, Scenario 2, Scenario 3, Scenario 4	
III	MongoDB Storage Engine:	Data Storage Engine, Data File (Relevant for MMAPv1), Namespace (.ns File), Data File (Relevant for WiredTiger), Reads and Writes, How Data Is Written Using Journaling, GridFS – The MongoDB File System, The Rationale of GridFS, GridFS under the Hood, Using GridFS, Indexing, Types of Indexes, Behaviors and Limitations	12
	MongoDB Use Cases:	Use Case 1 -Performance Monitoring, Schema Design, Operations, Sharding, Managing the Data, Use Case 2 – Social Networking, Schema Design, Operations, Sharding	
	MongoDB Limitations:	MongoDB Space Is Too Large (Applicable for MMAPv1), Memory Issues (Applicable for Storage Engine MMAPv1), 32-bit vs. 64-bit, BSON Documents, Namespaces Limits, Indexes Limit, Capped Collections Limit - Maximum Number of Documents in a Capped Collection, Sharding Limitations, Shard Early to Avoid Any Issues, Shard Key Can't Be Updated, Shard Collection Limit, Select the Correct Shard Key, Security Limitations, No Authentication by Default, Traffic to and from MongoDB Isn't Encrypted, Write and Read Limitations, Case-Sensitive Queries, Type-Sensitive Fields, No JOIN, Transactions, MongoDB Not Applicable Range	
	MongoDB Best Practices:	Deployment, Hardware Suggestions from the MongoDB Site, Few Points to be Noted, Coding, Application Response Time Optimization, Data Safety, Administration, Replication Lag, Sharding, Monitoring	
IV	The End of Disk? SSD and In-Memory Databases:	The End of Disk?, Solid State Disk, The Economics of Disk, SSD-Enabled Databases, In-Memory Databases, TimesTen, Redis, SAP HANA, VoltDB, Oracle 12c "in-Memory Database, Berkeley Analytics Data Stack and Spark, Spark Architecture	12

	jQuery:	Introduction, Traversing the DOM, DOM Manipulation with jQuery, Events, Ajax with jQuery, jQuery Plug-ins, jQuery Image Slider	
V	JSON:	Introduction, JSON Grammar, JSON Values, JSON Tokens, Syntax, JSON vs XML, Data Types, Objects, Arrays, Creating JSON, JSON Object, Parsing JSON, Persisting JSON, Data Interchange, JSON PHP, JSON HTML, JSONP	12

### Learning Resources recommended:

Sr.No.	Title	Author/s	Publisher	Edition	Year
1	Practical MongoDB	Shakuntala Gupta Edward Navin Sabharwal	Apress		
2	Beginning jQuery	Jack Franklin Russ Ferguson	Apress	Second	
3	Next Generation Databases	Guy Harrison	Apress		
4	Beginning JSON	Ben Smith	Apress		

### Evaluation Pattern

#### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

**B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):**

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

**Guidelines for paper pattern for Semester End Evaluation:**

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
  3. Refer **appendix I** for Paper pattern.
-

Name of the Course	<b>Next Generation Technologies Practical</b>
Course Code (refer to student handbook)	<b>USIT5P7</b>
Class	<b>T.Y.B.Sc.IT</b>
Semester	<b>V</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	<b>Elective</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

## Nomenclature: Next Generation Technologies Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Explore and experiment with emerging technologies and innovations.
2. Develop practical applications utilizing cutting-edge technologies.
3. Analyze the potential impact of next-gen technologies on various industries.
4. Collaborate on projects to address real-world challenges using these technologies.
5. Demonstrate adaptability and readiness for future advancements in technology.

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 1 MongoDB Basics	a Write a MongoDB query to create and drop database. b Write a MongoDB query to create, display and drop collection c Write a MongoDB query to insert, query, update and delete a document.	3
2	Practical 2	Simple Queries with MongoDB	3
3	Practical 3 Implementing Aggregation	a Write a MongoDB query to use sum, avg, min and max expression. b Write a MongoDB query to use push and addToSet expression. c Write a MongoDB query to use first and last expression.	3+3
4	Practical 4 Replication, Backup and Restore	a Write a MongoDB query to create Replica of existing database. b Write a MongoDB query to create a backup of existing database. c Write a MongoDB query to restore database from the backup.	3+3
5	Practical 5 Java and MongoDB	a Connecting Java with MongoDB and inserting, retrieving, updating and deleting.	3
6	Practical 6 PHP and MongoDB	a Connecting PHP with MongoDB and inserting, retrieving, updating and deleting.	3

7	Practical 7 Python and MongoDB	a Connecting Python with MongoDB and inserting, retrieving, updating and deleting.	3
8	Practical 8 Programs on Basic jQuery	a jQuery Basic, jQuery Events b jQuery Selectors, jQuery Hide and Show effects c jQuery fading effects, jQuery Sliding effects	3
9	Practical 9 jQuery Advanced	a jQuery Animation effects, jQuery Chaining b jQuery Callback, jQuery Get and Set Contents c jQuery Insert Content, jQuery Remove Elements and Attribute	3
10	Practical 10 JSON	a Creating JSON b Parsing JSON c Persisting JSON	3
11	Practical 11 Create a JSON file and import it to MongoDB	a Export MongoDB to JSON. b Write a MongoDB query to delete JSON object from MongoDB	3

This is sample Practical list. Course instructor may change the practical as per syllabus.

### Learning Resources recommended:

### Evaluation Pattern

#### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

#### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

# SEMESTER VI



Name of the Course	<b>Software Quality Assurance</b>
Course Code (refer to student handbook)	<b>USIT601</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>VI</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Skill Enhancement Course</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	employability/ entrepreneurship/ skill development

## Nomenclature: Software Quality Assurance

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand of Software Quality Assurance (SQA) Principles
2. Gain Knowledge of SQA Processes and Methodologies
3. Achieve Proficiency in Test Planning and Test Design
4. Compete in Test Execution and Defect Management
5. Solve Problems and Analytical Skills.
6. Aware about Emerging Trends and Technologies

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	Introduction to Quality:	Historical Perspective of Quality, What is Quality? (Is it a fact or perception?), Definitions of Quality, Core Components of Quality, Quality View, Financial Aspect of Quality, Customers, Suppliers and Processes, Total Quality Management (TQM), Quality Principles of Total Quality Management, Quality Management Through Statistical Process Control, Quality Management Through Cultural Changes, Continual (Continuous) Improvement Cycle, Quality in Different Areas, Benchmarking and Metrics, Problem Solving Techniques, Problem Solving Software Tools.	12
	Software Quality:	Introduction, Constraints of Software Product Quality Assessment, Customer is a King, Quality and Productivity Relationship, Requirements of a Product, Organisation Culture, Characteristics of Software, Software Development Process, Types of Products, Schemes of Criticality Definitions, Problematic Areas of Software Development Life Cycle, Software Quality Management, Why Software Has Defects? Processes Related to Software Quality, Quality Management System Structure, Pillars of Quality Management System, Important Aspects of Quality Management.	
II	Fundamentals of testing:	Introduction, Necessity of testing, What is testing? Fundamental test process, The psychology of testing, Historical Perspective of Testing, Definitions of Testing, Approaches to Testing, Testing During Development Life Cycle, Requirement Traceability Matrix, Essentials of Software Testing, Workbench, Important Features of Testing Process, Misconceptions About Testing,	12

		Principles of Software Testing, Salient Features of Good Testing, Test Policy, Test Strategy or Test Approach, Test Planning, Testing Process and Number of Defects Found in Testing, Test Team Efficiency, Mutation Testing, Challenges in Testing, Test Team Approach, Process Problems Faced by Testing, Cost Aspect of Testing, Establishing Testing Policy, Methods, Structured Approach to Testing, Categories of Defect, Defect, Error, or Mistake in Software, Developing Test Strategy, Developing Testing Methodologies (Test Plan), Testing Process, Attitude Towards Testing (Common People Issues), Test Methodologies/Approaches, People Challenges in Software Testing, Raising Management Awareness for Testing, Skills Required by Tester, Testing throughout the software life cycle, Software development models, Test levels, Test types, the targets of testing, Maintenance testing	
III	Unit Testing: Boundary Value Testing:	Normal Boundary Value Testing, Robust Boundary Value Testing, Worst-Case Boundary Value Testing, Special Value Testing, Examples, Random Testing, Guidelines for Boundary Value Testing	12
	Equivalence Class Testing:	Testing: Equivalence Classes, Traditional Equivalence Class Testing, Improved Equivalence Class Testing, Edge Testing, Guidelines and Observations.	
	Table-Based Testing:	Decision Tables, Decision Table Techniques, Cause-and-Effect Graphing, Guidelines and Observations	
	Path Testing:	Program Graphs, DD-Paths, Test Coverage Metrics, Basis Path Testing, Guidelines and Observations	
	Data Flow Testing:	Define/Use Testing, Slice-Based Testing, Program Slicing Tools.	
IV	Software Verification and Validation:	Introduction, Verification, Verification Workbench, Methods of Verification, Types of reviews on the basis of Stage Phase, Entities involved in verification, Reviews in testing lifecycle, Coverage in Verification, Concerns of Verification, Validation, Validation Workbench, Levels of Validation, Coverage in Validation, Acceptance Testing, Management of Verification and Validation, Software development verification and validation activities.	
	V-test Model:	Introduction, V-model for software, testing during Proposal stage, Testing during requirement stage, Testing during test planning phase, Testing during	

		design phase, Testing during coding, VV Model, Critical Roles and Responsibilities.	
V	Levels of Testing:	Introduction, Proposal Testing, Requirement Testing, Design Testing, Code Review, Unit Testing, Module Testing, Integration Testing, Big-Bang Testing, Sandwich Testing, Critical Path First, Sub System Testing, System Testing, Testing Stages.	12
	Special Tests:	Introduction, GUI testing, Compatibility Testing, Security Testing, Performance Testing, Volume Testing, Stress Testing, Recovery Testing, Installation Testing, Requirement Testing, Regression Testing, Error Handling Testing, Manual Support Testing, Intersystem Testing, Control Testing, Smoke Testing, Adhoc Testing, Parallel Testing, Execution Testing, Operations Testing, Compliance Testing, Usability Testing, Decision Table Testing, Documentation Testing, Training testing, Rapid Testing, Control flow graph, Generating tests on the basis of Combinatorial Designs, State Graph, Risk Associated with New Technologies, Process maturity level of Technology, Testing Adequacy of Control in New technology usage, Object Oriented Application Testing, Testing of Internal Controls, COTS Testing, Client Server Testing, Web Application Testing, Mobile Application Testing, eBusiness eCommerce Testing, Agile Development Testing, Data Warehousing Testing.	

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Software Testing and Continuous Quality Improvement	William E. Lewis	CRC Press	Third	2016
2.	Software Testing: Principles, Techniques and Tools	M. G. Limaye	TMH		2017
3.	Foundations of Software Testing	Dorothy Graham, Erik van Veenendaal, Isabel Evans, Rex Black	Cengage Learning	3rd	
4.	Software Testing: A Craftsman's Approach	Paul C. Jorgenson	CRC Press	4th	2017

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

#### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
  3. Refer **Appendix I** for Paper pattern.
-

Name of the Course	<b>Project Implementation Practical</b>
Course Code (refer to student handbook)	<b>USIT6P1</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>VI</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	<b>Skill Enhancement Course</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

## **Nomenclature:** Project Implementation

### **Course Outcomes:**

On the successful completion of this course, the learner will be able to

1. Understand the current state of the project management profession
2. Apply project management tools and techniques
3. Understand project management terminology.
4. Explore the appropriate methods to initiate, plan, execute, control and close projects

### **Refer Appendix II**

### **Evaluation Pattern**

#### **A. Continuous Internal Evaluation (20 Marks):**

<b>Method</b>	<b>Marks</b>
Project Documentation	10
Attendance and active participation in Laboratory	10

#### **B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):**

<b>Question No</b>	<b>Unit</b>	<b>Marks</b>
1	Documentation & Content Presentation	05
2	Problem Definition, Solutions Provided, Charts, Diagram, Planning and Methodology	05
3	Presentation Skills	05
4	Working and Functionality Testing	05
5	Successful Execution	05
6	Viva	05

Name of the Course	<b>Security in Computing</b>
Course Code (refer to student handbook)	<b>USIT602</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>VI</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	



## Nomenclature: Security in Computing

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand Computer Security Concepts
2. Gain Knowledge of Security Policies and Standards
3. Proficiency in Secure System Design
4. Understand the concepts of Cryptography
5. Have awareness of Malware and Intrusion Detection
6. Understand Incident Response and Disaster Recovery
7. Do Ethical and Legal Considerations

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	Information Security Overview:	The Importance of Information Protection, The Evolution of Information Security, Justifying Security Investment, Security Methodology, How to Build a Security Program, The Impossible Job, The Weakest Link, Strategy and Tactics, Business Processes vs. Technical Controls.	12
	Risk Analysis:	Threat Definition, Types of Attacks, Risk Analysis.	
	Secure Design Principles:	The CIA Triad and Other Models, Defense Models, Zones of Trust, Best Practices for Network Defense.	
II	Authentication and Authorization:	Authentication, Authorization	12
	Encryption:	A Brief History of Encryption, Symmetric-Key Cryptography, Public Key Cryptography, Public Key Infrastructure.	
	Storage Security:	Storage Security Evolution, Modern Storage Security, Risk Remediation, Best Practices.	
	Database Security:	General Database Security Concepts, Understanding Database Security Layers, Understanding Database-Level Security, Using Application Security, Database Backup	

		and Recovery, Keeping Your Servers Up to Date, Database Auditing and Monitoring	
III	Secure Network Design:	Introduction to Secure Network Design, Performance, Availability, Security	12
	Network Device Security:	Switch and Router Basics, Network Hardening.	
	Firewalls:	Overview, The Evolution of Firewalls, Core Firewall Functions, Additional Firewall Capabilities, Firewall Design.	
	Wireless Network Security:	Radio Frequency Security Basics, Data-Link Layer Wireless Security Features, Flaws, and Threats, Wireless Vulnerabilities and Mitigations, Wireless Network Hardening Practices and Recommendations, Wireless Intrusion Detection and Prevention, Wireless Network Positioning and Secure Gateways.	
IV	Intrusion Detection and Prevention Systems:	IDS Concepts, IDS Types and Detection Models, IDS Features, IDS Deployment Considerations, Security Information and Event Management (SIEM).	12
	Voice over IP (VoIP) and PBX Security:	Background, VoIP Components, VoIP Vulnerabilities and Countermeasures, PBX, TEM: Telecom Expense Management.	
	Operating System Security Models:	Operating System Models, Classic Security Models, Reference Monitor, Trustworthy Computing, International Standards for Operating System Security.	
V	Virtual Machines and Cloud Computing:	Virtual Machines, Cloud Computing.	12
	Secure Application Design:	Secure Development Lifecycle, Application Security Practices, Web Application Security, Client Application Security, Remote Administration Security.	
	Physical Security:	Classification of Assets, Physical Vulnerability Assessment, Choosing Site Location for Security, Securing Assets: Locks and Entry Controls, Physical Intrusion Detection.	

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	The Complete Reference: Information Security	Mark Rhodes-Ousley	McGraw-Hill	2nd	2013
2.	Essential Cybersecurity Science	Josiah Dykstra	O'Reilly	Fifth	2017
3.	Principles of Computer Security: CompTIA Security+ and Beyond	Wm.Arthur Conklin, Greg White	McGraw Hill	Second	2010

### Evaluation Pattern

#### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

#### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

#### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
3. Refer **appendix I** for Paper pattern.

Name of the Course	<b>Security in Computing Practical</b>
Course Code (refer to student handbook)	<b>USIT6P2</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>VI</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

## Nomenclature: Security in Computing Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand Computer Security Concepts
2. Gain Knowledge of Security Policies and Standards
3. Proficiency in Secure System Design
4. Understand the concepts of Cryptography
5. Have awareness of Malware and Intrusion Detection
6. Understand Incident Response and Disaster Recovery
7. Do Ethical and Legal Considerations

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 1	Configure Routers a OSPF MD5 authentication. b NTP. c to log messages to the syslog server. d to support SSH connections.	3
2	Practical 2	Configure AAA Authentication a Configure a local user account on Router and configure authenticate on the console and vty lines using local AAA b Verify local AAA authentication from the Router console and the PC-A client	3+3
3	Practical 3	Configuring Extended ACLs a Configure, Apply and Verify an Extended Numbered ACL	3+3
4	Practical 4	Configure IP ACLs to Mitigate Attacks and IPV6 ACLs a Verify connectivity among devices before firewall configuration. b Use ACLs to ensure remote access to the routers is available only from management station PC-C. c Configure ACLs on to mitigate attacks. d Configuring IPv6 ACLs	3+3
5	Practical 5	Configuring a Zone-Based Policy Firewall	3

6	Practical 6	Configure IOS Intrusion Prevention System (IPS) Using the CLI a Enable IOS IPS. b Modify an IPS signature.	3
7	Practical 7	Layer 2 Security a Assign the Central switch as the root bridge. b Secure spanning-tree parameters to prevent STP manipulation attacks. c Enable port security to prevent CAM table overflow attacks.	3
8	Practical 8	Layer 2 VLAN Security	3
9	Practical 9	Configure and Verify a Site-to-Site IPsec VPN Using CLI	3
10	Practical 10	Configuring ASA Basic Settings and Firewall Using CLI a Configure basic ASA settings and interface security levels using CLI b Configure routing, address translation, and inspection policy using CLI c Configure DHCP, AAA, and SSH d Configure a DMZ, Static NAT, and ACLs	3

This is sample Practical list. Course instructor may change the practical as per syllabus.

#### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	The Complete Reference: Information Security	Mark hodes-Ousley	McGraw-Hill	2nd	2013
2.	Essential Cybersecurity Science	Josiah Dykstra	O'Reilly	Fifth	2017

## Evaluation Pattern

### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

---

Name of the Course	<b>Business Intelligence</b>
Course Code (refer to student handbook)	<b>USIT603</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>VI</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	<b>Skill Enhancement Course</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>



## Nomenclature: Business Intelligence

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand Business Intelligence Concepts
2. Have Knowledge of Data Warehousing
3. Be Proficient in Data Analysis
4. Understand Data Visualization:
5. Do Data-driven Decision Making
6. Make Critical Thinking and Problem-Solving

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	Business intelligence:	Effective and timely decisions, Data, information and knowledge, The role of mathematical models, Business intelligence architectures, Ethics and business intelligence	12
	Decision support systems:	Definition of system, Representation of the decision-making process, Evolution of information systems, Definition of decision support system, Development of a decision support system	
II	Mathematical models for decision making:	Structure of mathematical models, Development of a model, Classes of models	12
	Data mining:	Definition of data mining, Representation of input data ,Data mining process, Analysis methodologies	
	Data preparation:	Data validation, Data transformation, Data reduction	
III	Classification:	Classification problems, Evaluation of classification models, Bayesian methods, Logistic regression, Neural networks, Support vector machines	
	Clustering:	Clustering methods, Partition methods, Hierarchical methods, Evaluation of clustering models	

IV	Business intelligence applications: Marketing models:	Relational marketing, Sales force management,	12
	Logistic and production models:	Supply chain optimization, Optimization models for logistics planning, Revenue management systems.	
	Data envelopment analysis:	Efficiency measures, Efficient frontier, The CCR model, Identification of good operating practices	
V	Knowledge Management:	Introduction to Knowledge Management, Organizational Learning and Transformation, Knowledge Management Activities, Approaches to Knowledge Management, Information Technology (IT) In Knowledge Management, Knowledge Management Systems Implementation, Roles of People in Knowledge Management	12
	Artificial Intelligence and Expert Systems:	Concepts and Definitions of Artificial Intelligence, Artificial Intelligence Versus Natural Intelligence, Basic Concepts of Expert Systems, Applications of Expert Systems, Structure of Expert Systems, Knowledge Engineering, Development of Expert Systems	

#### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Business Intelligence: Data Mining and Optimization for Decision Making	Carlo Vercellis	Wiley	First	2009
2.	Decision support and Business Intelligence Systems	Efraim Turban, Ramesh Sharda, Dursun Delen	Pearson	Ninth	2011
3.	Fundamental of Business Intelligence	Grossmann W, Rinderle-Ma	Springer	First	2015

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

#### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
  3. Refer **appendix I** for Paper pattern.
-

Name of the Course	<b>Business Intelligence Practical</b>
Course Code (refer to student handbook)	<b>USIT6P3</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>VI</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

## Nomenclature: Business Intelligence Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understanding Business Intelligence
2. Have Knowledge of Data Warehousing
3. Be Proficient in Data Analysis
4. Understand Data Visualization:
5. Do Data-driven Decision Making
6. Make Critical Thinking and Problem-Solving

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 1	Import the legacy data from different sources such as ( Excel , SqlServer, Oracle etc.) and load in the target system. ( You can download sample database such as Adventureworks, Northwind, foodmart etc.)	3
2	Practical 2	Perform the Extraction Transformation and Loading (ETL) process to construct the database in the Sqlserver.	3+3
3	Practical 3	a. Create the Data staging area for the selected database. b. Create the cube with suitable dimension and fact tables based on ROLAP, MOLAP and HOLAP model.	3+3
4	Practical 4	a. Create the ETL map and setup the schedule for execution. b. Execute the MDX queries to extract the data from the datawarehouse.	3+3
5	Practical 5	a. Import the datawarehouse data in Microsoft Excel and create the Pivot table and Pivot Chart. b. Import the cube in Microsoft Excel and create the Pivot table and Pivot Chart to perform data analysis.	3

6	Practical 6	Apply the what – if Analysis for data visualization. Design and generate necessary reports based on the data warehouse data.	3
7	Practical 7	Perform the data classification using classification algorithm.	3
8	Practical 8	Perform the data clustering using clustering algorithm.	3
9	Practical 9	Perform the Linear regression on the given data warehouse data.	3
10	Practical 10	Perform the logistic regression on the given data warehouse data.	3

This is sample Practical list. Course instructor may change the practical as per syllabus.

#### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Business Intelligence: Data Mining and Optimization for Decision Making	Carlo Vercellis	Wiley	First	2009
2.	Decision support and Business Intelligence Systems	Efraim Turban, Ramesh Sharda, Dursun Delen	Pearson	Ninth	2011
3.	Fundamental of Business Intelligence	Grossmann W, Rinderle-Ma	Springer	First	2015

## Evaluation Pattern

### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

---

Name of the Course	<b>Principles of Geographic Information Systems</b>
Course Code (refer to student handbook)	<b>USIT604</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>VI</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>



## Nomenclature: Principles of Geographic Information Systems

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand the fundamentals of Geographic Information Systems (GIS)
2. Proficiency in GIS software
3. Data acquisition and manipulation
4. Do Spatial analysis and modeling
5. Perform Cartographic visualization
6. Do Data querying and attribute management
7. Do Project design and implementation
8. Have Ethical and legal considerations

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	A Gentle Introduction to GIS: The nature of GIS:	Some fundamental observations, Defining GIS, GISystems, GIScience and GIApplications, Spatial data and Geoinformation.	12
	The real world and representations of it:	Models and modelling, Maps, Databases, Spatial databases and spatial analysis	
	Models and Representations of the real world Geographic Phenomena	Defining geographic phenomena, types of geographic phenomena, Geographic fields, Geographic objects, Boundaries	
	Computer Representations of Geographic Information:	Regular tessellations, irregular tessellations, Vector representations, Topology and Spatial relationships, Scale and Resolution, Representation of Geographic fields, Representation of Geographic objects, Organizing and Managing Spatial Data The Temporal Dimension	
II	Data Management and Processing Systems Hardware and Software Trends Geographic Information Systems:	GIS Software, GIS Architecture and functionality, Spatial Data Infrastructure (SDI)	12

	Stages of Spatial Data handling:	Spatial data handling and preparation, Spatial Data Storage and maintenance, Spatial Query and Analysis, Spatial Data Presentation.	
	Database management Systems:	Reasons for using a DBMS, Alternatives for data management, The relational data model, Querying the relational database.	
	GIS and Spatial Databases:	Linking GIS and DBMS, Spatial database functionality.	
III	Spatial Referencing and Positioning Spatial Referencing:	Reference surfaces for mapping, Coordinate Systems, Map Projections, Coordinate Transformations	12
	Data Entry and Preparation Spatial Data Input:	Direct spatial data capture, Indirect spatial data capture, Obtaining spatial data elsewhere	
	Data Quality:	Accuracy and Positioning, Positional accuracy, Attribute accuracy, temporal accuracy, Lineage, Completeness, Logical consistency	
	Data Preparation:	Data checks and repairs, Combining data from multiple sources	
	Point Data Transformation:	Interpolating discrete data, Interpolating continuous data	
IV	Spatial Data Analysis Classification of analytical GIS Capabilities Retrieval, classification and measurement:	Measurement, Spatial selection queries, Classification	12
	Overlay functions:	Vector overlay operators, Raster overlay operators	
	Neighbourhood functions:	Proximity computations, Computation of diffusion, Flow computation, Raster based surface analysis	
	Analysis:	Network analysis, interpolation, terrain modeling	
	GIS and Application models:	GPS, Open GIS Standards, GIS Applications and Advances	

	Error Propagation in spatial data processing:	How Errors propagate, Quantifying error propagation	
V	Data Visualization GIS and Maps, The Visualization Process Visualization Strategies: Present or explore? The cartographic toolbox	What kind of data do I have? How can I map my data?	12
	How to map?	How to map qualitative data, How to map quantitative data, How to map the terrain elevation, How to map time series, Map Cosmetics, Map Dissemination	

#### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Principles of Geographic Information Systems An Introductory Text Book	Editors: Otto Huisman and Rolf A.	The International Institute of Geoinformation Science and Earth Observation	Fourth	2009
2.	Principles of Geographic Information Systems	P.A Burrough and R.A.McDonnell	Oxford University Press	Third	1999
3.	Fundamentals of Spatial Information Systems,	R.Laurini and D. Thompson,	Academic Press		1994
4.	Fundamentals of Geographic Information Systems	Michael N.Demers	Wiley Publications	Fourth	2009
5.	Introduction to Geographic Information Systems	Chang Kang-tsung (Karl),	McGrawHill	Any above 3 <sup>rd</sup>	2013 7 <sup>th</sup> Edition

				Edition	
6.	GIS Fundamentals: A First Text on Geographic Information Systems	Paul Bolsatd	XanEdu Publishing Inc	5 <sup>th</sup> Edition	

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

### Guidelines for paper pattern for Semester End Evaluation:

- All questions will be compulsory.
- Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
- Refer **appendix I** for Paper pattern.

Name of the Course	<b>Principles of Geographic Information Systems Practical</b>
Course Code (refer to student handbook)	<b>USIT6P4</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>VI</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

## Nomenclature: Principles of Geographic Information Systems Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand the fundamentals of Geographic Information Systems (GIS)
2. Proficiency in GIS software
3. Data acquisition and manipulation
4. Do Spatial analysis and modeling
5. Perform Cartographic visualization
6. Do Data querying and attribute management
7. Do Project design and implementation
8. Have Ethical and legal considerations

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 1	Familiarizing Quantum GIS: Installation of QGIS, datasets for both Vector and Raster data, Maps.	3
2	Practical 2	Creating and Managing Vector Data: Adding vector layers, setting properties, formatting, calculating line lengths and statistics	3
3	Practical 3	Exploring and Managing Raster data: Adding raster layers, raster styling and analysis, raster mosaicking and clipping	3
4	Practical 4	Working with attributes, terrain Data	3
5	Practical 5	Working with Projections and WMS Data	3
6	Practical 6	Georeferencing Topo Sheets and Scanned Maps Georeferencing Aerial Imagery Digitizing Map Data	3
7	Practical 7	Managing Data Tables and Saptial data Sets: Table joins, spatial joins, points in polygon analysis, performing spatial queries	3
8	Practical 8	Advanced GIS Operations 1: Nearest Neighbor Analysis, Sampling Raster Data using Points or Polygons, Interpolating Point Data	3

9	Practical 9	Advance GIS Operations 2: Batch Processing using Processing Framework Automating Complex Workflows using Processing Modeler Automating Map Creation with Print Composer Atlas	3
10	Practical 10	Validating Map data	3

This is sample Practical list. Course instructor may change the practical as per syllabus.

### Learning Resources recommended:

1. <https://www.esri.com>
2. <https://www.gislounge.com>

### Evaluation Pattern

#### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

#### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

Name of the Course	<b>Enterprise Networking</b>
Course Code (refer to student handbook)	<b>USIT605</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>VI</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>



## Nomenclature: Enterprise Networking

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand Networking Fundamentals
2. Design and Configure Networks
3. Implement Network Services
4. Troubleshoot Network Issues
5. Implement Network Security

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	General Network Design:	Network Design Methodology, Architectures for the Enterprise, Borderless Networks Architecture, Collaboration and Video Architecture, Data Center and Virtualization Architecture, Design Lifecycle: Plan, Build, Manage Plan Phase Build Phase Manage Phase Prepare, Plan, Design, Implement, Operate, and Optimize Phases Prepare Phase Plan Phase Design Phase Implement Phase Operate Phase Optimize Phase Summary of PPDIOO Phases Project Deliverables Design Methodology Identifying Customer Design Requirements Characterizing the Existing Network Steps in Gathering Information Network Audit Tools Network Checklist Designing the Network Topology and Solutions Top-Down Approach Pilot and Prototype Tests Design Document	12
	Network Design Models:	Hierarchical Network Models Benefits of the Hierarchical Model, Hierarchical Network Design, Core Layer, Distribution Layer, Access Layer, Hierarchical Model Examples, Hub-and-Spoke, Design Collapsed Core, Design Enterprise Architecture Model, Enterprise Campus Module, Enterprise Edge Area, E-Commerce Module, Internet Connectivity Module, VPN/Remote Access, Enterprise WAN, Service Provider Edge Module, Remote Modules, Enterprise Branch Module, Enterprise Data Center Module, Enterprise Teleworker Module, High Availability Network Services, Workstation-to-Router	

		Redundancy and LAN, High Availability Protocols, ARP Explicit Configuration, RDP, RIP, HSRP, VRRP, GLBP, Server Redundancy, Route Redundancy, Load Balancing, Increasing Availability, Link Media Redundancy	
II	Enterprise LAN Design:	LAN Media, Ethernet Design Rules, 100Mbps Fast Ethernet Design Rules, Gigabit Ethernet Design Rules, 1000BASE-LX Long-Wavelength Gigabit Ethernet, 1000BASE-SX Short-Wavelength Gigabit Ethernet, 1000BASE-CX Gigabit Ethernet over Coaxial Cable, 1000BASE-T Gigabit Ethernet over UTP 86, 10 Gigabit Ethernet Design Rules, 10GE Media Types, EtherChannel, Comparison of Campus Media LAN Hardware, Repeaters, Hubs, Bridges, Switches, Routers, Layer 3 Switches, Campus LAN Design and Best Practices Best Practices for Hierarchical Layers, Access Layer Best Practices, Distribution Layer Best Practices, Core Layer Best Practices, STP Design Considerations, STP Toolkit, PortFast, UplinkFast, BackboneFast, Loop Guard, Root Guard, BPDU Guard, BPDU Filter, VLAN and Trunk Considerations, Unidirectional Link Detection (UDLD) Protocol, Large-Building LANs, Enterprise Campus LANs, Edge Distribution, Medium-Size LANs, Small and Remote Site LANs, Server Farm Module, Server Connectivity Options, Enterprise Data Center Infrastructure, Campus LAN QoS Considerations, Multicast Traffic Considerations, CGMP, IGMP Snooping.	12
	Data Center Design:	Enterprise DC Architecture, Data Center Foundation Components, Data Center Topology Components, Data Center Network Programmability, SDN, Controllers, APIs, ACI, Challenges in the DC, Data Center Facility Aspects, Data Center Space, Data Center Power, Data Center Cooling, Data Center Heat, Data Center Cabling, Enterprise DC Infrastructure, Data Center Storage, Data Center Reference Architecture, Defining the DC Access Layer, Defining the DC Aggregation Layer, Defining the DC Core Layer, Security in the DC, Fabric Extenders, Virtualization Overview, Challenges, Defining Virtualization and Benefits, Virtualization Risks, Types of Virtualization, Virtualization Technologies, VSS, VRF, vPC, Device Contexts, Server Virtualization, Server Scaling, Virtual Switching, Network Virtualization Design Considerations, Access Control, Path Isolation, Services Edge, Data Center Interconnect, DCI Use Cases, DCI Transport Options, DCI L2 Considerations, Load Balancing in the DC, Application Load Balancing, Network Load Balancing.	

III	Wireless LAN Design:	Wireless LAN Technologies, WLAN Standards, ISM and UNII Frequencies, Summary of WLAN Standards, Service Set Identifier, WLAN Layer 2 Access Method, WLAN Security, Unauthorized Access, WLAN Security Design Approach, IEEE 802.1X-2001 Port-Based Authentication, Dynamic WEP Keys and LEAP, Controlling WLAN Access to Servers, WLAN Authentication, Authentication Options, WLAN Controller Components, WLC Interface Types, AP Controller Equipment Scaling, Roaming and Mobility Groups, Intracontroller Roaming, Layer 2 Intercontroller Roaming, Layer 3 Intercontroller Roaming, Mobility Groups, WLAN Design, Controller Redundancy Design: Deterministic vs. Dynamic, N+1 WLC Redundancy, N+N WLC Redundancy, N+N+1 WLC Redundancy, Radio Management and Radio Groups, RF Groups, RF Site Survey, Using EoIP Tunnels for Guest Services, Wireless Mesh for Outdoor Wireless, Mesh Design Recommendations, Campus Design Considerations, Power over Ethernet (PoE), Wireless and Quality of Service (QoS), Branch Design Considerations, Local MAC, REAP, Hybrid REAP, Branch Office Controller Options.	12
	WAN Technologies and the Enterprise Edge:	WAN and Enterprise Edge Overview, Definition of WAN, WAN Edge Module, Enterprise Edge Modules, WAN Transport Technologies, ISDN, ISDN BRI Service, ISDN PRI Service, Digital Subscriber Line, Cable, Wireless, Frame Relay, Time-Division Multiplexing, Metro Ethernet, SONET/SDH, Multiprotocol Label Switching (MPLS), Dark Fiber, Dense Wavelength-Division Multiplexing, Ordering WAN Technology and Contracts, WAN and Edge Design Methodologies, Response Time, Throughput, Reliability, Bandwidth Considerations, WAN Link Categories, Optimizing Bandwidth Using QoS, Queuing, Traffic Shaping and Policing, Classification, Congestion Management, Priority Queuing, Custom Queuing, Weighted Fair Queuing, Class-Based Weighted Fair Queuing, Low-Latency Queuing, Traffic Shaping and Policing, Link Efficiency, Window Size, DMZ Connectivity, Segmenting DMZs, DMZ Services, Internet Connectivity, Centralized Internet (Branch) vs. Direct Internet (Branch), High Availability for the Internet Edge, VPN Network Design.	
	WAN Design	Traditional WAN Technologies Hub-and-Spoke Topology Full-Mesh Topology Partial-Mesh Topology Point-to-Point Topology Remote Site Connectivity Enterprise VPN vs. Service Provider VPN Enterprise Managed VPN:IPsec IPsec Direct Encapsulation	

		<p>Generic Routing Encapsulation IPsec DMVPN IPsec Virtual Tunnel Interface Design GETVPN Service Provider-Managed Offerings ,Metro Ethernet Service Provider VPNs:L2 vs. L3 ,Virtual Private Wire Services VPWS L2 VPN Considerations ,Virtual Private LAN Services VPLS L2 VPN Considerations ,MPLS, MPLS Layer 3 Design Overview MPLS L3 VPN Considerations ,VPN Benefits WAN Backup Design WAN Backup over the Internet Enterprise WAN Architecture Cisco Enterprise MAN/WAN Enterprise WAN/MAN Architecture Comparison ,Enterprise WAN Components Comparing Hardware and Software Enterprise Branch Architecture Branch Design Branch Connectivity Redundancy for Branches Single WAN Carrier vs. Dual WAN Carriers Single MPLS Carrier Site ,Dual MPLS Carriers Hybrid WAN: L3 VPN with IPsec VPN ,Internet for Branches Flat Layer 2 vs. Collapsed Core ,Enterprise Branch Profiles Small Branch Design Medium Branch Design Large Branch Design Enterprise Teleworker Design ,ISRs for Teleworkers</p>	
IV	IPv4,IPv6, ICMPv6	<p>Internet Protocol Version 4 Design,IPv4 Header ToS IPv4 Fragmentation IPv4 Addressing ,IPv4 Address Classes Class A Addresses Class B Addresses ,Class C Addresses Class D Addresses Class E Addresses ,IPv4 Address Types IPv4 Private Addresses NAT ,IPv4 Address Subnets Mask Nomenclature IP Address Subnet Design Example Determining the Network Portion of an IP Address Variable-Length Subnet Masks, Loopback Addresses IP Telephony Networks ,IPv4 Addressing Design Goal of IPv4 Address Design , Plan for Future Use of IPv4 Addresses , Performing Route Summarization , Plan for a Hierarchical IP Address Network , Private and Public IP Address and NAT Guidelines , Steps for Creating an IPv4 Address Plan Case Study: IP Address Subnet Allocation , Address Assignment and Name Resolution , Recommended Practices of IP Address Assignment , BOOTP DHCP DNS , Internet Protocol Version 6 Design, IPv6 Header IPv6 Address Representation IPv4-Compatible IPv6 Addresses IPv6 Prefix Representation IPv6 Address Scope Types and Address Allocations IPv6 Address Allocations IPv6 Unicast Address Global Unicast Addresses Link-Local Addresses , Unique Local IPv6 Address Global Aggregatable IPv6 Address , IPv4-Compatible IPv6 Address IPv6 Anycast Addresses , IPv6 Multicast Addresses IPv6 Mechanisms, IPv6 Neighbor Discovery Protocol IPv6 Name Resolution ,Path MTU Discovery IPv6 ddress-Assignment Strategies , Manual</p>	12

		<p>Configuration SLAAC of Link-Local Address , SLAAC of Globally Unique IPv6 Address DHCPv6 , DHCPv6 Lite IPv6 Security IPv6 Routing Protocols RIPng OSPFv3 , BGP4 Multiprotocol Extensions (MP-BGP) for IPv6 , IPv6 Addressing Design , Planning for Addressing with IPv6 , Route Summarization with IPv6 IPv6 Private Addressing IPv6 for the Enterprise IPv6 Address Allocation , Partly Linked IPv4 Address into IPv6, Whole IPv4 Address Linked into IPv6</p> <p>IPv6 Addresses Allocated Per Location and/or Type , IPv4-to-IPv6 Transition Mechanisms and Deployment Models , Dual-Stack Mechanism IPv6 over IPv4 Tunnels , Protocol Translation Mechanisms IPv6 Deployment Models , Dual-Stack Model Hybrid Model Service Block Model ,IPv6 Deployment Model Comparison IPv6 Comparison with IPv4 ,OSPF, BGP, Route Manipulation, and IP Multicast,OSPFv2 OSPFv2 Metric OSPFv2 Adjacencies and Hello Timers , OSPFv2 Areas OSPF Area Design Considerations OSPF Router Types OSPF DRs LSA Types Autonomous System External Path Types OSPF Stub Area Types Stub Areas Totally Stubby Areas , NSSAs Virtual Links OSPFv2 Router Authentication , OSPFv2 Summary OSPFv3 OSPFv3 Changes from OSPFv2, OSPFv3 Areas and Router Types OSPFv3 LSAs OSPFv3 Summary BGP BGP Neighbors eBGP iBGP Route Reflectors Confederations BGP Administrative Distance, BGP Attributes, Weight, and the BGP Decision Process BGP Path Attributes Next-Hop Attribute Local Preference Attribute Origin Attribute Autonomous System Path Attribute MED Attribute Community Attribute Atomic Aggregate and Aggregator Attributes Weight BGP Decision Process, BGP Summary, Route Manipulation PBR Route Summarization Route Redistribution Default Metric OSPF Redistribution Route Filtering Transit Traffic Routing Protocols on the Hierarchical Network Infrastructure IP Multicast Review, Multicast Addresses Layer 3 to Layer 2 Mapping IGMP, IGMPv1 IGMPv2 IGMPv3 CGMP IGMP Snooping, Sparse Versus Dense Multicast Multicast Source and Shared Trees PIM PIM-SM PIM DR Auto-RP PIMv2 Bootstrap Router, DVMRP IPv6 Multicast Addresses</p>	
V	Managing Security	<p>Network Security Overview Security Legislation Security Threats Reconnaissance and Port Scanning Vulnerability Scanners Unauthorized Access Security Risks Targets Loss of Availability Integrity Violations and Confidentiality Breaches , Security Policy and Process Security Policy Defined , Basic Approach of a Security Policy Purpose of Security Policies, Security Policy</p>	12

		<p>Components Risk Assessment , Risk Index Continuous Security Integrating Security Mechanisms into Network Design Trust and Identity Management ,Trust Domains of Trust Identity Passwords Tokens Certificates ,Network Access Control Secure Services Encryption Fundamentals Encryption Keys VPN Protocols , Transmission Confidentiality Data Integrity Threat Defense , Physical Security Infrastructure Protection Security Management Solutions Security Solution Network Security Platforms , Trust and Identity Technologies Firewall Fundamentals ,Types of Firewalls Next-Gen Firewalls NAT Placement , Firewall Guidelines Firewall ACLs , Identity and Access Control Deployments Detecting and Mitigating Threats IPS/IDS Fundamentals IPS/IDS Guidelines , Threat Detection and Mitigation Technologies, Threat-Detection and Threat-Mitigation Solutions, FirePOWER IPS Security Management Applications, Security Platform Solutions Security Management Network Integrating Security into Network Devices IOS Security, ISR G2 Security Hardware Options Securing the Enterprise, Implementing Security in the Campus Implementing Security in the Data Center Implementing Security in the Enterprise Edge Network Management Protocols, Simple Network Management Protocol SNMP Components, MIB SNMP Message Versions SNMPv1 SNMPv2 SNMPv3, Other Network Management Technologies RMON, RMON2 NetFlow Compared to RMON and SNMP, CDP LLDP Syslog</p>	
--	--	--	--

#### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	CCDA200-310 Official Cert Guide	ANTHONY BRUNO, CCIE No. 2738 STEVE JORDAN, CCIE No. 11293	Cisco Press		
2.	Network Warrior	Gary A Donabue	O Reilly	2nd	2011

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
  3. Refer **appendix I** for Paper pattern.
-



Name of the Course	<b>Enterprise Networking Practical</b>
Course Code (refer to student handbook)	<b>USIT6P5</b>
Class	<b>T.Y.B.Sc.IT</b>
Semester	<b>VI</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
	<b>Discipline Specific Elective Practical (Any One)*</b>
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>



## Nomenclature: Enterprise Networking Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand Networking Fundamentals
2. Design and Configure Networks
3. Implement Network Services
4. Troubleshoot Network Issues
5. Implement Network Security

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 1	Configuring OSPF – I a Single-Area OSPF Link Costs and Interface Priorities b Multi-Area OSPF with Stub Areas and Authentication	3
2	Practical 2	Configuring OSPF – II a OSPF Virtual Links and Area Summarization b OSPF over Frame Relay	3+3
3	Practical 3	Redistribution and Administrative Distances a Redistribution Between RIP and OSPF b Manipulating Administrative Distances	3+3
4	Practical 4	BGP a Configuring BGP with Default Routing b Using the AS_PATH Attribute c BGP Route Reflectors and Route Filters	3+3
5	Practical 5	IPv6 a Configuring OSPF for IPv6 b Configuring 6to4 Tunnels	3
6	Practical 6	VLANs and EtherChannel a Static VLANs, VLAN Trunking, and VTP Domains and Modes b Configuring EtherChannel	3
7	Practical 7	Spanning Tree Protocol a Spanning Tree Protocol (STP) Default Behavior b Modifying Default Spanning Tree Behavior	3
8	Practical 8	VLAN and Spanning Tree a Per-VLAN Spanning Tree Behavior b Multiple Spanning Tree	3

9	Practical 9	Internal VLAN Routing a Inter-VLAN Routing with an External Router b Inter-VLAN Routing with an Internal Route Processor	3
10	Practical 10	Configure NAT Services	3

This is sample Practical list. Course instructor may change the practical as per syllabus.

### Learning Resources recommended:

Sr.No.	Title	Author/s	Publisher	Edition	Year
1.	CCDA200-310 Official Cert Guide	ANTHONY BRUNO, CCIE No. 2738 STEVE JORDAN, CCIE No. 11293	Cisco Press		
2.	Network Warrior	Gary A Donabue	O Reilly	2nd	2011

### Evaluation Pattern

#### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

#### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

Name of the Course	<b>IT Service Management</b>
Course Code (refer to student handbook)	<b>USIT606</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>VI</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

## Nomenclature: IT Service Management

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand IT Service Management principles
2. Have knowledge of ITSM processes
3. Be proficient in ITSM tools and technologies
4. Understand Service strategy and design
5. Know ITSM governance and continuous improvement
6. Be familiar with ITSM implementation and project management
7. Do Incident and problem management

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	IT Service Management	Introduction, What is service management? What are services? Business Process, Principles of Service management: Specialisation and Coordination, The agency principle, Encapsulation, Principles of systems, The service Life Cycle, Functions and processes across the life cycle.	12
	Service Strategy Principles:	Value creation, Service Assets, Service Provider Service Structures, Service Strategy Principles. Service Strategy: Define the market, Develop the offerings, Develop Strategic Assets, Prepare for execution.	
	Service Strategy:	Define the market, Develop the offerings, Develop Strategic Assets, Prepare for execution.	
	Challenges, Critical Success factors and risks:	Complexity, Coordination and Control, Preserving value, Effectiveness in measurement, Risks.	
II	Service Design:	Fundamentals	12

	Service Design Principles:	Goals, Balanced Design, Identifying Service requirements, identifying and documenting business requirements and drivers, Design activities, Design aspects, Subsequent design activities, Design constraints, Service oriented architecture, Business Service Management, Service Design Models	
	Service Design Processes:	Service Catalogue Management, Service Level Management, Capacity Management, Availability Management, IT Service Continuity Management, Information Security Management, Supplier Management	
	Challenges, Critical Success factors and risks:	Challenges, Critical Success factors, Risks, Service Transition under difficult Conditions.	
III	Service Transition:	Fundamentals	12
	Service Transition Principles:	Principles Supporting Service Transition, Policies for Service Transition	
	Service Transition Processes:	Transition planning and support, Change Management, Service Asses Configuration Management, Service and Deployment Management, Service Validation and Testing, Evaluation, Knowledge Management.	
	Challenges, Critical Success factors and risks:	Challenges, Critical Success factors, Risks, Service Transition under difficult Conditions.	
IV	Service Operation:	Fundamentals	12
	Service Operation Principles:	Functions, groups, teams, departments and divisions, achieving balance in service operations, Providing service, Operation staff involvement in service design and service transition, Operational Health, Communication, Documentation	
	Service Operation Processes:	Event Management, Incident Management, Request fulfilment, Problem Management, Access Management, Operational activities of processes covered in other lifecycle phases.	
	Challenges, Critical Success factors and risks:	Challenges, Critical Success factors, Risks	

V	Continual Service Improvement(CSI) Principles:	CSI Approach, CSI and organizational change, Ownership, CSI register, External and Internal drivers, Service level management, Knowledge management, The Deming cycle, Service Measurement, IT governance, Frameworks, models, standards and quality Systems, CSI inputs and outputs.	12
	CSI Process:	The seven-step improvement process.	
	CSI Methods nad Techniques:	Methods and techniques, Assessments, benchmarking, Service Measurement, Metrics, Return on Investment, Service reporting, CSI and other service management processes,	
	Organising for CSI:	Organisational development, Functions, roles, Customer Engagement, Responsibility model - RACI, Competence and training.	
	Technology considerations:	Tools to support CSI activities.	
	Implementing CSI:	Critical Considerations for implementing CSI, The start, Governance, CSI and organisational change, Communication Strategy and Plan	

#### Learning Resources recommended:

Sr.No.	Title	Author/s	Publisher	Edition	Year
1.	ITIL v3 Foundation Complete Certification Kit				2009
2.	ITIL v3 Service Strategy		OGC/TSO		
3.	ITIL v3 Service Transition		OGC/TSO		
4.	ITIL v3 Service Operation		OGC/TSO		
5.	ITIL Continual Service Improvement		TSO	2011	2011

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
  3. Refer **appendix I** for Paper pattern.
-

Name of the Course	<b>Cyber Laws</b>
Course Code (refer to student handbook)	<b>USIT607</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>VI</b>
No of Credits	<b>2</b>
Nature	<b>Theory</b>
Type (applicable to NEP only)	
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>



## Nomenclature: Cyber Laws

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Understand Cyber Law concepts
2. Get awareness of legal issues in cyberspace
3. Understand digital rights and freedoms
4. Be familiar with cybercrime legislation
5. Have knowledge of Legal considerations for cybersecurity
6. Perform Ethical and policy implications of cyber law

### Curriculum:

Unit	Title	Learning Points	No of Lectures
I	Power of Arrest Without Warrant Under the IT Act, 2000:	A Critique, Crimes of this Millennium, Section 80 of the IT Act, 2000 – A Weapon or a Farce? Forgetting the Line Between Cognizable and Non-Cognizable Offences, Necessity of Arrest without Warrant from Any Place, Public or Otherwise, Check and Balances Against Arbitrary Arrests, Arrest for “About to Commit” an Offence Under the IT Act: A Tribute to Draco, Arrest, But NO Punishment!	12
	Cyber Crime and Criminal Justice: Penalties, Adjudication and Appeals Under the IT Act, 2000:	Concept of “Cyber Crime “ and the IT Act , 2000, Hacking, Teenage Web Vandals, Cyber Fraud and Cyber Cheating, Virus on the Internet, Defamation, Harassment and E-mail Abuse, Cyber Pornography, Other IT Act Offences, Monetary Penalties, Adjudication and Appeals Under IT Act , 2000, Network Service Providers, Jurisdiction and Cyber Crime, Nature of Cyber Criminality, Strategies to Tackle Cyber Crime and Trends, Criminal Justice in India and Implications on Cyber Crime.	
II	Contracts in the Infotech World:	Contracts in the Infotech World, Click-Wrap and Shrink-Wrap Contract: Status under the Indian Contract Act, 1872, Contract Formation Under the Indian Contract Act,1872, Contract Formation on the Internet, Terms and Conditions of Contracts.	12

	Jurisdiction in the Cyber World:	Questioning the Jurisdiction and Validity of the Present Law of Jurisdiction, Civil Law of Jurisdiction in India, Cause of Action, Jurisdiction and the Information Technology Act, 2000, Foreign Judgements in India, Place of Cause of Action in Contractual and IPR Disputes, Exclusion Clauses in Contracts, Abuse of Exclusion Clauses, Objection of Lack of Jurisdiction, Misuse of the Law of Jurisdiction, Legal Principles on Jurisdiction in the United State of America, Jurisdiction Disputes w.r.t. the Internet in the United State of America.	
III	Battling Cyber Squatters and Copyright Protection in the Cyber World:	Concept of Domain Name and Reply to Cyber Squatters, Meta-Tagging, Legislative and Other Innovative Moves Against Cyber Squatting, The Battle Between Freedom and Control on the Internet, Works in Which Copyright Subsists and meaning of Copyright, Copyright Ownership and Assignment, License of Copyright, Copyright Terms and Respect for Foreign Works, Copyright Infringement, Remedies and Offences, Copyright Protection of Content on the Internet; Copyright Notice, Disclaimer and Acknowledgement, Downloading for Viewing Content on the Internet, Hyper-Linking and Framing, Liability of ISPs for Copyright Violation in the Cyber World: Legal Developments in the US, Napster and its Cousins: A Revolution on the Internet but a Crisis for Copyright Owners, Computer Software Piracy.	12
IV	E-Commerce Taxation: Real Problems in the Virtual World:	A Tug of War on the Concept of 'Permanent Establishment', Finding the PE in Cross Border E-Commerce, The United Nations Model Tax Treaty, The Law of Double Taxation Avoidance Agreements and Taxable Jurisdiction Over Non-Residents, Under the Income Tax Act, 1961, Tax Agents of Non-Residents under the Income Tax Act, 1961 and the Relevance to E-Commerce, Source versus Residence and Classification between Business Income and Royalty, The Impact of the Internet on Customer Duties, Taxation Policies in India: At a Glance.	12

	Digital Signature, Certifying Authorities and E-Governance:	Digital Signatures, Digital Signature Certificate, Certifying Authorities and Liability in the Event of Digital Signature Compromise, E-Governance in India: A Warning to Babudom!	
V	The Indian Evidence Act of 1872 v. Information Technology Act, 2000:	Status of Electronic Records as Evidence, Proof and Management of Electronic Records; Relevancy, Admissibility and Probative Value of E-Evidence, Proving Digital Signatures, Proof of Electronic Agreements, Proving Electronic Messages, Other Amendments in the Indian Evidence Act by the IT Act, Amendments to the Bankers Books Evidence Act, 1891 and Reserve Bank of India Act, 1934.	12
	Protection of Cyber Consumers in India:	Are Cyber Consumers Covered Under the Consumer Protection Act? Goods and Services, Consumer Complaint, Defect in Goods and Deficiency in Services, Restrictive and Unfair Trade Practices, Instances of Unfair Trade Practices, Reliefs Under CPA, Beware Consumers, Consumer Foras, Jurisdiction and Implications on cyber Consumers in India, Applicability of CPA to Manufacturers, Distributors, Retailers and Service Providers Based in Foreign Lands Whose Goods are Sold or Services Provided to a Consumer in India. Amendments in Indian IT Act 2000	

### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Cyber Law Simplified	Vivek Sood	TMH Education		2001
2.	Cybersecurity Law	Jeff Kosseff	Wiley		2017

## Evaluation Pattern

### A. Continuous Internal Evaluation (40 Marks):

Method	Marks
Unit Test (MCQ / Descriptive – Based on Theory and/or Problems Online/Offline – 1 unit test of 20 marks	20
Assignments	10
Attendance and active participation in classroom	10

### B. Semester End Evaluation (Paper Pattern) (60 Marks – 2 hours):

Question No	Unit	Marks
1	I	12
2	II	12
3	III	12
4	IV	12
5	V	12

### Guidelines for paper pattern for Semester End Evaluation:

1. All questions will be compulsory.
  2. Descriptive type of questions, programming-based questions, problem solving / numericals based questions, single line answers, etc., will contain internal options.
  3. Refer **appendix I** for Paper pattern.
-

Name of the Course	<b>Advanced Mobile Programming Practical</b>
Course Code (refer to student handbook)	<b>USIT6P6</b>
Class	<b>T.Y.BSc.IT</b>
Semester	<b>IV</b>
No of Credits	<b>2</b>
Nature	<b>Practical</b>
Type (applicable to NEP only)	
Highlight revision specific to employability/ entrepreneurship/ skill development (if any) 100 words	<b>employability/ entrepreneurship/ skill development</b>

## Nomenclature: Advanced Mobile Programming Practical

### Course Outcomes:

On the successful completion of this course, the learner will be able to

1. Build enterprise level mobile applications with Kotlin on Android.
2. Understand both the basic and advanced concepts of Kotlin.
3. Understand why use Kotlin over Java.
4. Install and configure Android Studio.
5. Explain and use key Android programming concepts.

### Curriculum:

Sr. No.	Title	Learning Points	No of Lectures
1	Practical 1: Introduction to Android, Introduction to Android Studio IDE, Application Fundamentals:	Creating a Project, Android Components, Activities, Services, Content Providers, Broadcast Receivers, Interface overview, Creating Android Virtual device, USB debugging mode, Android Application Overview. Simple "Hello World" program.	3
2	Practical 2: Programming Resources	Android Resources: (Color, Theme, String, Drawable, Dimension, Image)	3+3
3	Practical 3: Programming Activities and fragments	Activity Life Cycle, Activity methods, Multiple Activities, Life Cycle of fragments and multiple fragments.	3+3
4	Practical 4: Programs related to different Layouts	Coordinate, Linear, Relative, Table, Absolute, Frame, List View, Grid View.	3+3
5	Practical 5: Programming UI elements	AppBar, Fragments, UI Components	3
6	Practical 6:	Programming menus, dialog, dialog fragments	3

7	Practical 7: Programs on Intents, Events, Listeners and Adapters	The Android Intent Class, Using Events and Event Listeners	3
8	Practical 8	Programs on Services, notification and broadcast receivers	3
9	Practical 9	Database Programming with SQLite	3
10	Practical 10	Programming threads, handles and asynchronized programs	3
11	Practical 11	Programming Media API and Telephone API	3
12	Practical 12	Programming Security and permissions	3
13	Practical 13	Programming Network Communications and Services (JSON)	3

This is sample Practical list. Course instructor may change the practical as per syllabus.

#### Learning Resources recommended:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1.	Android Programming with Kotlin for Beginners	John Horton	packt publisher	-	2019
2.	Head First Android Development: A Brain-Friendly Guide.	David Griffiths	O'Reilly Media, Inc.	-	2015
3	Android App Development for Dummies.	Michael Burton	Wiley	-	2015

## Evaluation Pattern

### A. Continuous Internal Evaluation (20 Marks):

Method	Marks
Certified Journal	10
Attendance and active participation in Laboratory	10

### B. Semester End Evaluation (Paper Pattern) (30 Marks – 2 hours):

Question No	Unit	Marks
1	Program 1	10
2	Program 2	15
3	Viva	05

---

Chairperson

BoS, Information Technology



# APPENDIX I

## PAPER PATTERN

### 1. Internal Evaluation

Test: 1 Class test of 20 marks. (Can be taken online/offline)

<b>Q</b>	<b>Attempt <i>any four</i> of the following:</b>	<b>20</b>
<b>a.</b>		
<b>b.</b>		
<b>c.</b>		
<b>d.</b>		
<b>e.</b>		
<b>f.</b>		

### 2. External Examination: (60 marks) (Currently for SY and TY)

	<b>All questions are compulsory</b>	
<b>Q1</b>	<b>(Based on Unit 1) Attempt <i>any three</i> of the following:</b>	<b>12</b>
<b>a.</b>		
<b>b.</b>		
<b>c.</b>		
<b>d.</b>		
<b>e.</b>		
<b>f.</b>		
<b>Q2</b>	<b>(Based on Unit 2) Attempt <i>any three</i> of the following:</b>	<b>12</b>
<b>Q3</b>	<b>(Based on Unit 3) Attempt <i>any three</i> of the following:</b>	<b>12</b>
<b>Q4</b>	<b>(Based on Unit 4) Attempt <i>any three</i> of the following:</b>	<b>12</b>
<b>Q5</b>	<b>(Based on Unit 5) Attempt <i>any three</i> of the following:</b>	<b>12</b>

**3. External Examination: (60 marks) (For FY)**

	<b>All questions are compulsory</b>	
<b>Q1</b>	<b>(Based on Unit 1) Attempt <i>any three</i> of the following:</b>	<b>15</b>
<b>a.</b>		
<b>b.</b>		
<b>c.</b>		
<b>d.</b>		
<b>e.</b>		
<b>f.</b>		
<b>Q2</b>	<b>(Based on Unit 2) Attempt <i>any three</i> of the following:</b>	<b>15</b>
<b>Q3</b>	<b>(Based on Unit 3) Attempt <i>any three</i> of the following:</b>	<b>15</b>
<b>Q4</b>	<b>(Based on Units 1,2 &amp; 3) Attempt <i>any three</i> of the following:</b>	<b>15</b>
	<b>(Set 2 Sub questions on each unit)</b>	

# APPENDIX – II

## Project Dissertation Semester V and Project Implementation Semester VI

Chapter 1 to 4 should be submitted in Semester V in spiral binding. These chapter have also to be included in the Semester VI report. Semester VI report has to be hard bound with golden embossing. Students will be evaluated based on the dissertation in semester V and dissertation and viva voce in Semester VI.

### I. OBJECTIVES

- Describe the Systems Development Life Cycle (SDLC).
- Evaluate systems requirements.
- Complete a problem definition.
- Evaluate a problem definition.
- Determine how to collect information to determine requirements.
- Perform and evaluate feasibility studies like cost-benefit analysis, technical feasibility, time feasibility and Operational feasibility for the project.
- Work on data collection methods for fact finding.
- Construct and evaluate data flow diagrams.
- Construct and evaluate data dictionaries.
- Evaluate methods of process description to include structured English, decision tables and decision trees.
- Evaluate alternative tools for the analysis process.
- Create and evaluate such alternative graphical tools as systems flow charts and state transition diagrams.
- Decide the S/W requirement specifications and H/W requirement specifications.
- Plan the systems design phase of the SDLC.
- Distinguish between logical and physical design requirements.
- Design and evaluate system outputs.

- Design and evaluate systems inputs.
- Design and evaluate validity checks for input data.
- Design and evaluate user interfaces for input.
- Design and evaluate file structures to include the use of indexes.
- Estimate storage requirements.
- Explain the various file update processes based on the standard file organizations.
  - Decide various data structures.
- Construct and evaluate entity-relationship (ER) diagrams for RDBMS related projects.
  - Perform normalization for the unnormalized tables for RDBMS related projects
- Decide the various processing systems to include distributed, client/server, online and others.
- Perform project cost estimates using various techniques.
- Schedule projects using both GANTT and PERT charts.
- Perform coding for the project.
- Documentation requirements and prepare and evaluate systems documentation.
  - Perform various systems testing techniques/strategies to include the phases of testing.
  - Systems implementation and its key problems.
- Generate various reports.
- Be able to prepare and evaluate a final report.
- Brief the maintenance procedures and the role of configuration management in operations.
  - To decide the future scope and further enhancement of the system.
- Plan for several appendices to be placed in support with the project report documentation.
  - Decide the various processing systems to include distributed, client/server, online and others.
- Perform project cost estimates using various techniques.
- Schedule projects using both GANTT and PERT charts.
- Perform coding for the project.
- Documentation requirements and prepare and evaluate systems documentation.
  - Perform various systems testing techniques/strategies to include the

phases of testing.

- Systems implementation and its key problems.
- Generate various reports.
- Be able to prepare and evaluate a final report.
- Brief the maintenance procedures and the role of configuration management in operations.
- To decide the future scope and further enhancement of the system.
- Plan for several appendices to be placed in support with the project report documentation.
- Work effectively as an individual or as a team member to produce correct, efficient, well organized and documented programs in a reasonable time.
- Recognize problems that are amenable to computer solutions, and knowledge of the tool necessary for solving such problems.
- Develop of the ability to assess the implications of work performed.
- Get good exposure and command in one or more application areas and on the software
- Develop quality software using the software engineering principles
- Develop of the ability to communicate effectively.

## II. Type of the Project

The majority of the students are expected to work on a real-life project preferably in some industry/ Research and Development Laboratories/Educational Institution/Software Company. Students are encouraged to work in the areas listed below. However, it is **not mandatory** for a student to work on a real-life project. The student can formulate a project problem with the help of her/his Guide and submit the project proposal of the same. **Approval of the project proposal is mandatory.** If approved, the student can commence working on it, and complete it. Use the latest versions of the software packages for the development of the project.

### III. SOFTWARE AND BROAD AREAS OF APPLICATION

<b>FRONT END / GUI Tools</b>	.Net Technologies, Java
<b>DBMS/BACK END</b>	Oracle, SQL Plus, MY SQL, SQL Server,
<b>LANGUAGES</b>	C, C++, Java, VC++, C#, R,Python
<b>SCRIPTING LANGUAGES</b>	PHP,JSP, SHELL Scripts (Unix), Tcl/TK,
<b>.NET Platform</b>	F#,C#. Net, Visual C#. Net, ASP.Net
<b>MIDDLE WARE (COMPONENT) TECHNOLOGIES</b>	COM/DCOM, Active-X, EJB
<b>UNIX INTERNALS</b>	Device Drivers, RPC, Threads, Socket ming
<b>NETWORK/WIRELESS TECHNOLOGIES</b>	-
<b>REALTIME OPERATING SYSTEM/ EMBEDDED SKILLS</b>	LINUX, Raspberry Pi, Arduino, 8051
<b>APPLICATION AREAS</b>	Financial / Insurance / Manufacturing / Multimedia / Computer Graphics / Instructional Design/ Database Management System/ Internet / Intranet / Computer Networking-Communication Software development/ E Commerce/ ERP/ MRP/ TCP-IP programming / Routing protocols programming/ Socket programming.

### IV.Introduction

The project report should be documented with scientific approach to the solution of the problem that the students have sought to address. The project report should be prepared in order to solve the problem in a methodical and professional manner, making due references to appropriate techniques, technologies and professional standards. The student should start the documentation process from the first phase of software development so that one can easily identify the issues to be focused upon in the ultimate project report. The student should also include

the details from the project diary, in which they will record the progress of their project throughout the course. The project report should contain enough details to enable examiners to evaluate the work. The important points should be highlighted in the body of the report, with details often referred to appendixes.

## **1.1 PROJECT REPORT:**

Title Page

Original Copy of the Approved Proforma of the Project Proposal

Certificate of Authenticated work

Role and Responsibility Form

Abstract

Acknowledgement

Table of Contents

Table of Figures

### **CHAPTER 1: INTRODUCTION**

1.1 Background

1.2 Objectives

1.3 Purpose, Scope, and Applicability

1.3.1 Purpose

1.3.2 Scope

1.3.3 Applicability

1.4 Achievements

1.5 Organization of Report

### **CHAPTER 2: SURVEY OF TECHNOLOGIES**

### **CHAPTER 3: REQUIREMENTS AND ANALYSIS**

3.1 Problem Definition

3.2 Requirements Specification

3.3 Planning and Scheduling

3.4 Software and Hardware Requirements

3.5 Preliminary Product Description

3.6 Conceptual Models

### **CHAPTER 4: SYSTEM DESIGN**

4.1 Basic Modules

4.2 Data Design

- 4.2.1 Schema Design
- 4.2.2 Data Integrity and Constraints
- 4.3 Procedural Design
  - 4.3.1 Logic Diagrams
  - 4.3.2 Data Structures
  - 4.3.3 Algorithms Design
- 4.4 User interface design
- 4.5 Security Issues
- 4.6 Test Cases Design

The documentation should use tools like star UML, Visuo for windows, Rational Rose for design as part of Software Project Management Practical Course. The documentation should be spiral bound for semester V and the entire documentation should be hard bound during semester VI.

## CHAPTER 5: IMPLEMENTATION AND TESTING

- 5.1 Implementation Approaches
- 5.2 Coding Details and Code Efficiency
  - 5.2.1 Code Efficiency
- 5.3 Testing Approach
  - 5.3.1 Unit Testing
  - 5.3.2 Integrated Testing
  - 5.3.3 Beta Testing
- 5.4 Modifications and Improvements
- 5.5 Test Cases

## CHAPTER 6: RESULTS AND DISCUSSION

- 6.1 Test Reports
- 6.2 User Documentation

## CHAPTER 7: CONCLUSIONS

- 7.1 Conclusion
  - 7.1.1 Significance of the System
- 7.2 Limitations of the System
- 7.3 Future Scope of the Project



REFERENCES

GLOSSARY

APPENDIX A

APPENDIX B

## **V. EXPLANATION OF CONTENTS**

### **Title Page**

Sample format of Title page is given in Appendix 1 of this block. Students should follow the given format.

### **Original Copy of the Approved Proforma of the Project Proposal**

Sample Proforma of Project Proposal is given in Appendix 2 of this block. Students should follow the given format.

### **Certificate of Authenticated work**

Sample format of Certificate of Authenticated work is given in Appendix 3 of this block. Students should follow the given format.

### **Role and Responsibility Form**

Sample format for Role and Responsibility Form is given in Appendix 4 of this block. Students should follow the given format.

### **Abstract**

This should be one/two short paragraphs (100-150 words total), summarizing the project work. It is important that this is not just a re-statement of the original project outline. A suggested flow is background, project aims and main achievements. From the abstract, a reader should be able to ascertain if the project is of interest to them and, it should present results of which they may wish to know more details.

### **Acknowledgements**

This should express student's gratitude to those who have helped in the preparation of project.

### **Table of Contents:**

The table of contents gives the readers a view of the detailed structure of the report. The students would need to provide section and subsection headings with associated pages. The formatting details of these sections and subsections are given below.

### **Table of Figures:**

List of all Figures, Tables, Graphs, and Charts etc. along with their page numbers in a table of figures.

### **Chapter 1: Introduction**

The introduction has several parts as given below:

**Background:** A description of the background and context of the project and its relation to work already done in the area. Summarize existing work in the area concerned with the project work.

**Objectives:** Concise statement of the aims and objectives of the project. Define exactly what is going to be done in the project; the objectives should be about 30 /40 words. **Purpose, Scope and Applicability:** The description of Purpose, Scope, and Applicability are given below:

- **Purpose:** Description of the topic of the project that answers questions on why this project is being done. How the project could improve the system its significance and theoretical framework.
- **Scope:** A brief overview of the methodology, assumptions and limitations. The students should answer the question: What are the main issues being covered in the project? What are the main functions of the project?
- **Applicability:** The student should explain the direct and indirect applications of their work. Briefly discuss how this project will serve the computer world and people. **Achievements:** Explain what knowledge the student achieved after the completion of the work. What contributions has the project made to the chosen area? **Goals achieved** - describes the degree to which the findings support the original objectives laid out by the project. The goals may be partially or fully achieved, or exceeded.

Organization of Report: Summarizing the remaining chapters of the project report, in effect, giving the reader an overview of what is to come in the project report.

### **Chapter 2: Survey of Technologies**

In this chapter Survey of Technologies should demonstrate the student's awareness and understanding of Available Technologies related to the topic of the project. The student should give the detail of all the related technologies that are necessary to complete the project. They should describe the technologies available in the chosen area and present a comparative study of all those Available Technologies. Explain why the student selected the one technology for the completion of the objectives of the project.

### **Chapter 3: Requirements and Analysis**

**Problem Definition:** Define the problem on which the students are working in the project. Provide details of the overall problem and then divide the problem in to sub-problems. Define each sub-problem clearly.

**Requirements Specification:** In this phase the student should define the requirements of the system, independent of how these requirements will be accomplished. The Requirements Specification describes the things in the system and the actions that can be done on these things. Identify the operation and problems of the existing system.

**Planning and Scheduling:** Planning and scheduling is a complicated part of software development. Planning, for our purposes, can be thought of as determining all the small tasks that must be carried out in order to accomplish the goal. Planning also takes into account, rules known as constraints, which, control when certain tasks can or cannot happen. Scheduling can be thought of as determining whether adequate resources are available to carry out the plan. The student should show the Gantt chart and Program Evaluation Review Technique (PERT).

**Software and Hardware Requirements:** Define the details of all the software and hardware needed for the development and implementation of the project.

- **Hardware Requirement:** In this section, the equipment, graphics card, numeric co-processor, mouse, disk capacity, RAM capacity etc. necessary to run the

software must be noted. • **Software Requirements:** In this section, the operating system, the compiler, testing tools, linker, and the libraries etc. necessary to compile, link and install the software must be listed. **Preliminary Product Description:** Identify the requirements and objectives of the new system. Define the functions and operation of the application/system the students are developing as project.

**Conceptual Models:** The student should understand the problem domain and produce a model of the system, which describes operations that can be performed on the system, and the allowable sequences of those operations. Conceptual Models could consist of complete Data Flow Diagrams, ER diagrams, Object-oriented diagrams, System Flowcharts etc.

#### **Chapter 4: System Design**

Describes desired features and operations in detail, including screen layouts, business rules, process diagrams, pseudo code and other documentation.

**Basic Modules:** The students should follow the divide and conquer theory, so divide the overall problem into more manageable parts and develop each part or module separately. When all modules are ready, the student should integrate all the modules into one system. In this phase, the student should briefly describe all the modules and the functionality of these modules.

**Data Design:** Data design will consist of how data is organized, managed and manipulated.

- **Schema Design:** Define the structure and explanation of schemas used in the project.

- **Data Integrity and Constraints:** Define and explain all the validity checks and constraints provided to maintain data integrity.

**Procedural Design:** Procedural design is a systematic way for developing algorithms or procedurals.

- **Logic Diagrams:** Define the systematic flow of procedure that improves its comprehension and helps the programmer during implementation. e.g., Control Flow Chart, Process Diagrams etc.

- **Data Structures:** Create and define the data structure used in procedures. •

**Algorithms Design:** With proper explanations of input data, output data, logic of processes, design and explain the working of algorithms.

User Interface Design: Define user, task, environment analysis and how to map those requirements in order to develop a “User Interface”. Describe the external and internal components and the architecture of user interface. Show some rough pictorial views of the user interface and its components.

Security Issues: Discuss Real-time considerations and Security issues related to the project and explain how the student intends avoiding those security problems. What are the security policy plans and architecture?

Test Cases Design: Define test cases, which will provide easy detection of errors and mistakes within a minimum period of time and with the least effort. Explain the different conditions in which the students wish to ensure the correct working of the project.

## **Chapter 5: Implementation and Testing**

Implementation Approaches: Define the plan of implementation, and the standards the students have used in the implementation.

Coding Details and Code Efficiency: Students not need include full source code, instead, include only the important codes (algorithms, applets code, forms code etc). The program code should contain comments needed for explaining the work a piece of code does. Comments may be needed to explain why it does it, or, why it does a particular way.

The student can explain the function of the code with a shot of the output screen of that program code.

- Code Efficiency: The student should explain how the code is efficient and how the students have handled code optimization.

Testing Approach: Testing should be according to the scheme presented in the system design chapter and should follow some suitable model – e.g., category partition, state machine-based. Both functional testing and user-acceptance testing are appropriate. Explain the approach of testing.

- Unit Testing: Unit testing deals with testing a unit or module as a whole. This

would test the interaction of many functions but, do confine the test within one module.

- **Integrated Testing:** Brings all the modules together into a special testing environment, then checks for errors, bugs and interoperability. It deals with tests for the entire application. Application limits and features are tested here.

**Modifications and Improvements:** Once the students finish the testing they are bound to be faced with bugs, errors and they will need to modify your source code to improve the system. Define what modification are implemented in the system and how it improved the system.

## **Chapter 6: Results and Discussion**

**Test Reports:** Explain the test results and reports based on the test cases, which should show that the project is capable of facing any problematic situation and that it works fine in different conditions. Take the different sample inputs and show the outputs.

**User Documentation:** Define the working of the software; explain its different functions, components with screen shots. The user document should provide all the details of the product in such a way that any user reading the manual, is able to understand the working and functionality of the document.

## **Chapter 7: Conclusions**

**Conclusion:** The conclusions can be summarized in a fairly short chapter (2 or 3 pages). This chapter brings together many of the points that would have made in the other chapters. **Limitations of the System:** Explain the limitations encountered during the testing of the project that the students were not able to modify. List the criticisms accepted during the demonstrations of the project.

**Future Scope of the Project** describes two things: firstly, new areas of investigation prompted by developments in this project, and secondly, parts of the current work that was not completed due to time constraints and/or problems encountered.

## REFERENCES

It is very important that the students acknowledge the work of others that they have used or adapted in their own work, or that provides the essential background or context to the project. The use of references is the standard way to do this. Please follow the given standard for the references for books, journals, and online material. The citation is mandatory in both the reports.

E.g:

Linhares, A., & Brum, P. (2007). Understanding our understanding of strategic scenarios: What role do chunks play? *Cognitive Science*, 31(6), 989-1007.  
<https://doi.org/doi:10.1080/03640210701703725>

*Lipson, Charles (2011). Cite right: A quick guide to citation styles; MLA, APA, Chicago, the sciences, professions, and more (2nd ed.). Chicago [u.a.]: University of Chicago Press. p. 187. ISBN 9780226484648.*

Elaine Ritchie, J Knite. (2001). *Artificial Intelligence, Chapter 2 ,p.p 23 - 44.* Tata McGrawHill.

## GLOSSARY

If you the students any acronyms, abbreviations, symbols, or uncommon terms in the project report then their meaning should be explained where they first occur. If they go on to use any of them extensively then it is helpful to list them in this section and define the meaning.

## APPENDICES

These may be provided to include further details of results, mathematical derivations, certain illustrative parts of the program code (e.g., class interfaces), user documentation etc.

In particular, if there are technical details of the work done that might be useful to others who wish to build on this work, but that are not sufficiently important to the project as a whole to justify being discussed in the main body of the project, then they should be included as appendices.

## **VI. SUMMARY**

Project development usually involves an engineering approach to the design and development of a software system that fulfils a practical need. Projects also often form an important focus for discussion at interviews with future employers as they provide a detailed example of what the students are capable of achieving. In this course the students can choose your project topic from the lists given in Unit 4: Category-wise Problem Definition.

## **VII. FURTHER READINGS**

1. Modern Systems Analysis and Design; Jeffrey A. Hoffer, Joey F. George, Joseph,S. Valacich; Pearson Education; Third Edition; 2002.
2. ISO/IEC 12207: Software Life Cycle Process  
(<http://www.software.org/quagmire/descriptions/iso-iec12207.asp>).
3. IEEE 1063: Software User Documentation (<http://ieeexplore.ieee.org>).
4. ISO/IEC: 18019: Guidelines for the Design and Preparation of User Documentation for Application Software.
5. <http://www.sce.carleton.ca/squall>.
6. <http://en.tldp.org/HOWTO/Software-Release-Practice-HOWTO/documentation.html>.
7. <http://www.sei.cmu.edu/cmm/>



**PROFORMA FOR THE APPROVAL PROJECT PROPOSAL**

***(Note:All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.)***

PNR no.: ..... Roll no: \_\_\_\_\_

1. Name of the Student

\_\_\_\_\_

2. Title of the Project

\_\_\_\_\_

3. Name of the Guide

\_\_\_\_\_

4. Teaching experience of the Guide

\_\_\_\_\_

5. Is this your first submission? Yes No

Signature of the Student

Signature of the Guide

Date: .....

Date: .....

Signature of the Co-ordinator

Date: .....

(All the text in the report should be in times new roman)

# **TITLE OF THE PROJECT**

(NOT EXCEEDING 2 LINES, 24 BOLD, ALL CAPS)

**A Project Report (12 Bold)**

Submitted in partial fulfillment of the  
Requirements for the award of the Degree of (size-12)

**BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)(14 BOLD, CAPS)**

**By(12 Bold)**

Name of The Student (size-15, title case)

Seat Number (size-15)

**Under the esteemed guidance of (13 bold)**

**Mr./Mrs. Name of The Guide (15 bold, title case)**

**Designation (14 Bold, title case)**

**COLLEGE LOGO**

**DEPARTMENT OF INFORMATION TECHNOLOGY(12 BOLD, CAPS)**

**COLLEGE NAME (14 BOLD, CAPS)**

**CITY, PIN CODE(12 bold, CAPS)**

**MAHARASHTRA (12 bold, CAPS)**

**YEAR (12 bold)**

**COLLEGE NAME (14 BOLD, CAPS)**  
**CITY-MAHARASHTRA-PINCODE(13 bold, CAPS)**

**DEPARTMENT OF INFORMATION TECHNOLOGY (14 BOLD, CAPS)**

**College Logo**

**CERTIFICATE (14 BOLD, CAPS, underlined, centered)**

This is to certify that the project entitled, "**Title of The Project**", is bonafied work of **NAME OF THE STUDENT** bearing Seat.No: **(NUMBER)** submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY from College Name . (12, times new roman, justified)

**Internal Guide (12 bold)**

**Coordinator**

(Don't write names of lecturers or HOD)

**External Examiner**

**Date:**

**College Seal**

**COMPANY CERTIFICATE (if applicable)**

(Project Abstract page format)

**Abstract (20bold, caps, centered)**

**Content (12, justified)**

**Note: Entire document should be with 1.5  
line spacing and all paragraphs should start with 1 tab space.**

# **ACKNOWLEDGEMENT**

**(20, BOLD, ALL CAPS, CENTERED)**

**The acknowledgement should be in times new roman, 12 font with 1.5 line spacing, justified.**

(Declaration page format)

## **DECLARATION (20 bold, centered, allcaps)**

**Content (12, justified)**

I here by declare that the project entitled, “**Title of the Project**” done at **place where the project is done**, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of **BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)** to be submitted as final semester project as part of our curriculum.

**Name and Signature of the Student**

# **TABLE OF CONTENTS (20bold, caps, centered)**

Should be generated automatically using word processing software.

<b>Chapter 1: Introduction</b>	01(no bold)
<b>1.1 Background</b>	02(no bold)
<b>1.2 Objectives</b>	....
<b>1.3 Purpose and Scope</b>	....
1.2.1 Purpose	....
1.2.2 Scope	....

.....  
.....

<b>Chapter 2: System Analysis</b>	
<b>2.1 Existing System</b>	
<b>2.2 Proposed System</b>	
<b>2.3 Requirement Analysis</b>	
<b>2.4 Hardware Requirements</b>	
<b>2.5 Software Requirements</b>	
<b>2.6 Justification of selection of Technology</b>	

<b>Chapter 3: System Design</b>	
<b>3.1 Module Division</b>	
<b>3.2 Data Dictionary</b>	
<b>3.3 ER Diagrams</b>	
<b>3.4 DFD/UML Diagrams</b>	

## **Chapter 4: Implementation and Testing**

<b>4.1 Code (Place Core segments)</b>	
<b>4.2 Testing Approach</b>	
<b>4.2.1 Unit Testing (Test cases and Test Results)</b>	
<b>4.2.2 Integration System (Test cases and Test Results)</b>	

<b>Chapter 5: Results and Discussions (Output Screens)</b>	
<b>Chapter 6: Conclusion and Future Work</b>	
<b>Chapter 7: References</b>	



## **List of Tables (20 bold, centered, Title Case)**

Should be generated automatically using word processing software.

## **List of Figures (20 bold, centered, Title Case)**

Should be generated automatically using word processing software.

**(Project Introduction page format)**

# **Chapter 1**

## **Introduction (20 Bold, centered)**

**Content or text (12, justified)**

**Note: Introduction has to cover brief description of the project with minimum 4 pages.**

# Chapter 2

## System Analysis (20 bold, Centered)

Subheadings are as shown below with following format (16 bold, CAPS)

### 2.1 Existing System (16 Bold)

#### 2.1.1 ----- (14 bold, title case)

##### 2.1.1.1 ----- (12 bold, title case)

### 2.2 Proposed System

### 2.3 Requirement Analysis

### 2.4 Hardware Requirements

### 2.5 Software Requirements

### 2.6 Justification of Platform – (how h/w & s/w satisfying the project)

Table 2.1: Caption


# Chapter 3

## System Design (20 bold, centered)

Subheadings are as shown below with following format (16 bold, CAPS)  
Specify figures as Fig 11.1 – caption

**3.1 Module Division**

**3.2 Data Dictionary**

**3.3 E-R Diagrams**

**3.4 Data Flow Diagrams / UML**

Note: write brief description at the bottom of all diagrams

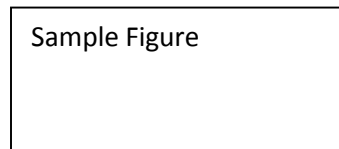


Fig. 3.1: Caption

# Chapter 4

## Implementation and Testing (20 bold, centered)

### 4.1 Code (Place Core segments)

Content includes description about coding phase in your project (Font-12)

(\* don't include complete code ---- just description)

### 4.2 Testing Approach

Subheadings are as shown below with following format (16 bold, CAPS)

#### 4.2.1 Unit Testing

#### 4.2.2 Integration Testing

Note:

- Explain about above testing methods
- Explain how the above techniques are applied in your project  
Provide Test plans, test cases, etc relevant to your project

## **Chapter 5**

### **Results and Discussions(20 bold, centered)**

Note: Place Screen Shots and write the functionality of each screen at the bottom

## **Chapter 6**

### **Conclusion and Future Work (20 bold, centered)**

The conclusions can be summarized in a fairly short chapter around 300 words. Also include limitations of your system and future scope (12, justified)



# Chapter 7

## References (20 bold, centered)

Content (12, LEFT)

[1] Title of the book, Author

[2] Full URL of online references

[3] .....

### **\*NOTE ABOUT PROJECT VIVA VOCE:**

Student may be asked to write code for problem during VIVA to demonstrate his coding capabilities and he/she may be asked to write any segment of coding used in the in the project. The project can be done in group of at most four students. However, the length and depth of the project should be justified for the projects done in group. A big project can be modularised and different modules can be assigned as separate project to different students.

*The plagiarism should be maintained as per the UGC guidelines.*

**R. E. Society's  
R. P. Gogate College of Arts & Science And  
R. V. Jogalekar College of Commerce (Autonomous), Ratnagiri**

**Department of Information Technology**

**Add-On Course**

<b>ADD-ON COURSE</b>			
<b>Course Code</b>	<b>Course Name</b>	<b>Course Nature</b>	<b>No. of Hrs.</b>
S/IT/23-24/1	Hardware and Networking	Tutorials + Practical	30 hrs

Name of the Course	Hardware and Networking
Course Code (refer to student handbook)	
Class	FY/SY/TYBSc.IT
Semester	
No of Credits	
Nature	Tutorials + Practical

**Nomenclature: Hardware and Networking**

**Course Objectives:**

1. To provide students with practical skills in computer hardware and networking.
2. To enhance employability by preparing students for entry-level IT support and network technician roles.
3. To help students obtain industry-recognized certifications in hardware and networking.

**Course Outcomes:**

On the successful completion of this course, the learner will be able to

1. Identify and describe the major hardware components of a computer system, including CPU, RAM, motherboard, storage devices, and peripheral devices.
2. Demonstrate knowledge of various operating systems and their installation.
3. Understand software installation and configuration.
4. Configure and troubleshoot basic network settings, such as IP addresses, subnet masks, and DNS settings.
5. Identify and resolve common network issues.
6. Understand the principles of network security, including firewalls, encryption, and access control.

**Curriculum:**

<b>Unit</b>	<b>Title</b>	<b>Learning Points</b>	<b>No of Hours</b>
I	Computer System	Introduction of Computer, Operating Systems, Introduction Of BIOS ( Basic Input Out-put System),Computer Ports and Interfaces	10
II	Hardware & Software	Introduction Of Motherboard , Processors and RAM, Internal And External Storage Devices (HDD, DVD etc.),Peripheral Devices (I/O Devices),Utility Software	10
III	Networking n Troubleshooting	Assembling & Disassembling Desktop, Installation of operating system and software,Networking,Troubleshooting Of Computer	10

**Evaluation Pattern :**

<b>Method</b>	<b>Marks</b>
quizzes and assignments.	50
Practical hands-on labs and projects.	50