R. P. Gogate College of Arts & Science and

R. V. Jogalekar College of Commerce (Autonomous),

Ratnagiri

Bachelor of Science (Computer Science) (B.Sc. CS)
Programme Three Year Integrated Programme-
Six Semesters

Course Structure

Under Choice Based Credit System (CBCS)

To be implemented from Academic Year 2023-2024
Progressively

Preamble

The restructured curriculum for the First year computer science is systematically designed considering the National Education Policy (NEP) along with the proposed Autonomy constraints.

The rise of Information and Communication Technology (ICT) has profoundly affected modern society. Increasing applications of computers in almost all areas of human endeavor has led to vibrant industries with concurrent rapid change in technology.

As the computing field advances at a rapid pace, the students must possess a solid foundation that allows and encourages them to maintain relevant skills as the field evolves. Specific languages and technology platforms change over time. Thus students must continue to learn and adapt their skills throughout their careers. To develop this ability, students will be exposed to multiple programming languages, tools, paradigms and technologies as well as the fundamental underlying principles throughout this programmed.

The core philosophy of this programme is to –

- Form strong foundations of Computer Science
- Nurture programming, analytical & design skills for the real-world problems.
- Introduce emerging trends to the students in a gradual way.
- Groom the students for the challenges of ICT industry

The students these days not only aspire for a career in the industry but also look for research opportunities. The main aim of this programme is to deliver a modern curriculum that will equip graduates with strong theoretical and practical backgrounds to enable them to excel in the workplace and to be lifelong learners. Not only does it prepare the students for a career in the Software industry, it also motivates them towards further studies and research opportunities.

In the first year i.e. for semester I & II, the basic foundation of important skills required for software development is laid. The syllabus proposes to have two major courses of Computer Science and one minor course of Applied Mathematics. All major subjects are proposed to have theory as well as practical tracks. While the Computer Science courses will form fundamental skills for solving computational problems, the Applied Mathematics allows students to develop the use of quantitative methods in thinking about and solving problems.

Courses on Value Education plays an important role in helping individuals carry out social, moral, and democratic obligations. The main purpose of Skill Enhancement courses is to provide students with life-skills in the hands-on mode to increase their employability. The purpose of Ability Enhancement Course is to introduce students to the theory, fundamentals and tools of communication and to develop vital communication skills in them which should be integral to personal, social and professional interactions. Vocational and Skill Enhancement Courses like web designing and Linux focuses on developing technical skills in Computer Science. Indian Knowledge System courses focuses on the evolution of computer and the computing environment in technology. Open Elective courses are offered to construct basic diluted structure of computer science.

We sincerely believe that any student enrolling in this programmed will get very strong foundation and exposure to basics, advanced and emerging trends of the subject.

We wholeheartedly thank all experts who shared their valuable feedbacks and suggestions in order to improvise the contents; we have sincerely attempted to incorporate each of them. We further thank Chairperson and members of Board of Studies for their confidence in us.

Special thanks to Department of Computer Science and colleagues from various colleges, who volunteered or have indirectly, helped designing certain specialized courses and the syllabus as a whole.

The performance of the learners shall be evaluated into two parts. The learner's performance shall be assessed by Internal Assessment with 40% marks in the first part and by conducting the Semester End Examinations with 60% marks in the second part.

Name of Programme	**BACHELOR OF SCIENCE (B.Sc.)(Computer Science)**
Level	Under Graduate
No of Semesters	06
Year of Implementation	**2023-24**
Programme Specific Outcomes (PSO)	1. To formulate, model, design solutions, procedure and to use software tools to solve real world problems. 2. To design and develop computer programs/computer - based systems in the areas such as networking, web design, security, cloud computing, IoT, data science and other emerging technologies. 3. To familiarize with the modern-day trends in industry and research-based settings and thereby innovate novel solutions to existing problems. 4. To apply concepts, principles, and theories relating to computer science to new situations. 5. To use current techniques, skills, and tools necessary for computing practice 6. To apply standard Software Engineering practices and strategies in real-time software project development 7. To pursue higher studies of specialization and to take up technical employment. 8. To work independently or collaboratively as an effective team member on a substantial software project. 9. To communicate and present their work effectively and coherently. 10. To display an ethical code of conduct in usage of Internet and Cyber systems. 11. To engage in independent and life-long learning in the background of rapid changing IT industry

Relevance of PSOs to the local, regional, national, and global developmental needs	The revised and restructured curriculum for the Three-year integrated course is systematically designed considering the National Education Policy (NEP) along with the proposed Autonomy constraints. The focus is on current industry needs in terms of skills sets demanded under the new technological environment. It also endeavors to align the programmer structure and course curriculum with student aspirations and corporate expectations. The proposed curriculum is more contextual, industry friendly and suitable to cater the needs of society and nation in the present-day context. As the world is heading towards digitalization, the syllabus is framed by concentrating on such needs. With the help of basic programming skills and designing technique, a student is able to make small workable projects which can be beneficial for local vendors. To make these projects strong, various concepts of project management, database management are involved. Various industries need IT support at local or across the country. Students having the skills of computers are able to sustain in the IT industry at various locations. Development in thought processes can lead one in a research area to contribute to an upcoming trend. Students can become entrepreneurs since the techniques of management and entrepreneurship are taught to them. Due to the learning of cyber safety, ethical hacking, students can become ethical hackers where students can contribute to national security. Students can make community awareness due to learning about cyber safety. Continuous growth in trends requires students updated which will help them mercurial. This will help in sustaining the IT industry and become employable.

B.Sc.(Computer Science) Programme
Under Choice Based Credit System(CBCS) Course Structure (Autonomous)

F.Y.B.Sc.(Computer Science)

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

No. of Courses	Semester I	Credits	No. of Courses	Semester II	Credits
	Department Specific Courses (DSC)			***Department Specific Courses (DSC)***	
	Major			***Major***	
23_USCS101	Digital Systems & Architecture (T)	**02**	23_USCS201	Database Systems (T)	**02**
23_USCS102	Fundamentals of Programming Paradigm (T)	**02**	23_USCS202	Programming with C++ (T)	**02**
23_USCS103	Computer Science Practical 1 (P)	**02**	23_USCS203	Computer Science Practical 2 (P)	**02**
	Minor			Minor	
23_USCS104	Applied Mathematics – I (T)	**02**	23_USCS204	Applied Mathematics – II (T)	**02**
23_USCS105	Statistics using R practical (P)	**02**	23_USCS205	Data Structures (T)	**02**
24_USCS110	Computer Organization and Architecture (T) (For other students)	**02**	23_USCS206	Data Structures Practical (P)	**02**
24_USCS111	Programming with C(T) (For other students)	**02**	24_USCS210	Fundamentals of Database Systems(T) (For other students)	**02**
			24_USCS211	Logic Building with Python (T) (For other students)	**02**
			24_USCS212	Database and python practical's (P)(For other students)	**02**
	Generic / Open Elective			***Generic / Open Elective***	
	Any one Course from table 1	**02**		Any one Course from table 1	**02**
	Indian Knowledge System (IKS)				

23_USCS106	Evolution of Computing in India (T)	**02**			
	Vocational Skill Course (VSC)			***Skill Enhancement Course (SEC)***	
23_USCS107	Programming with Python Practical (P)	**02**	23_USCSS207	LINUX Practical (P)	**02**
	Ability Enhancement Course (AEC)			***Ability Enhancement Course (AEC)***	
23_USCS108	English (Soft Skill Developme nt -I) (T)	**02**	23_USCS208	Content Writing (T)	**02**
	Value Education Course (VEC)			***Value Education Course (VEC)***	
23_USCS109	Green Computing – I (T)	**02**	23_USCS209	Green Computing – II (T)	**02**
	Co-Curricular			***Co-Curricular***	
	Any one Course from table 2 (P)	**02**		Any one Course from table 2 (P)	**02**
Total Credits		**22**	**Total Credits**		**22**

Table 1: Open Electives for Science, IT, BT, CS Faculty

Open Elective (Any One)		***Open Elective (Any One)***	
24_USOEIT101	Google workspace and multimedia apps (P)	24_USOE201	Basics of Excel (P)
24_USOEBT102	Introduction to Food Science(T)	24_USOEBT202	Introduction to Bioinformatics (T)
23_USOEZO103	Health and Hygiene I (T)	23_USOE203	Basic computer system (T)
24_USOEPH104	Introduction to basic Astronomy (P)	23_USOE204	Health and Hygiene II (T)
23_USOE106	Social media marketing (T)	24_USOEPH204	Observational Astronomy (T)
		24_USOEIT206	Social Media Awareness (P)

Table 2: Co-Curricular Courses

11. Co-Curricular (Any One)		*11. Co-Curricular (Any One)*	
25_GJCC101	National Social Service (NSS)	25_GJCC201	National Social Service (NSS)
24_GJCC102	National Cadet Corps (NCC)	24_GJCC202	National Cadet Corps (NCC)
24_GJCC103	Sports and Yoga	24_GJCC203	Sports and Yoga
24_GJCC104	Cultural	24_GJCC204	Cultural
24_GJCC105	Career Katta	24_GJCC205	Career Katta
24_GJCC106	Life Long Learning	24_GJCC206	Life Long Learning
24_GJCC107	Research club	24_GJCC207	Research club
24_GJCC108	Science Association	24_GJCC208	Science Association
24_GJCC109	Film Club	24_GJCC209	Film Club
24_GJCC110	Infosys courses	24_GJCC210	Infosys courses

Table 3. Skill enhancement courses for Science, IT, BT, CS Faculty

Skill Enhancement Courses	
23_USCH204	Skills in Chemical Analysis I (P)
23_USZOSEC204	Aquarium Fish Keeping (P)
24_USBOTSEC1	Propagation practices for garden plants I (P)
23_USPH204	Basic Measurement skills and data analysis (P)
25_USMT205	Basic Maths for competitive Exams (P)
23_USCSS207	LINUX Practical (P)
23_USBTS207	Bio-analytical Techniques (P)
23_USITS208	Computer Skills-2 Practical (P)

Semester I

Revised Syllabus of Courses of B.Sc. Computer Science

Programme at Semester I

with Effect from the Academic Year 2023-2024

Department Specific Courses (DSC)

Major Course

Name of the Course	**Digital Systems & Architecture**
Course Code	23_USCS101
Class	FYBSC
Semester	I
No of Credits	2
Nature	Theory
Type	Major
Employability/ entrepreneurship/ skill development	This course introduces basic components used in computer system. Various number systems such as binary, hexadecimal and octal are introduced which will help learners how data gets stored in computer system and transmitted through computer system. Logic gates, combinational and sequential circuits are introduced in practical. Hardware concepts are kept so that learners will able to understand mechanism of hardware. This course will help those students who are interested in hardware and networking.

Digital Systems & Architecture
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Fundamentals of Digital Logic, Computer System	10
2	Memory System Organization, Processor Organization	10
3	Control Unit, Fundamentals of Advanced Computer Architecture	10
Total		**30**

Course Outcomes
The learner will be able to **CO1:** Demonstrate a comprehensive understanding of digital logic design principles, including boolean algebra, logic gates, and simplification techniques. **CO2:** Get comprehensive knowledge of computer memory systems, including hierarchy and design, as well as proficiency in processor organization. **CO3:**To understand the working principles of multiprocessor and parallel organization's as advanced computer architectures

Sr. No.	Modules / Units
1	**Fundamentals of Digital Logic ,Computer System (10 Lectures)**
	Boolean algebra, Logic Gates, Simplification of Logic Circuits: Algebraic Simplification, Combinational Circuits: Adders, Mux, De-Mux, Sequential Circuits: Flip- Flops (SR, JK & D), Counters: synchronous and asynchronous Counter. Comparison of Computer Organization & Architecture, Computer Components and Functions, Interconnection Structures. Bus Interconnections, Input / Output: I/O Module, Interrupt Driven I/O, Direct Memory Access, Number System: Binary, Decimal, Octal, Hexadecimal.
2	**Memory System Organization, Processor Organization (10 Lectures)**
	Classification and design parameters, Memory Hierarchy, Internal Memory: RAM, SRAM and DRAM, Cache Memory: Design Principles, Virtual Memory, External Memory: Magnetic Discs, Optical Memory, Flash Memories. Instruction Formats, Instruction Sets, Addressing Modes, Addressing Modes Examples with Assembly Language [8085/8086 CPU], Register Organization, Basic Microprocessor operations: Data Transfer (Register / Memory) Operations, Arithmetic & Logical Operations, Introduction to RISC and CISC Architecture, Micro-Operations,
3	**Control Unit, Fundamentals of Advanced Computer Architecture (10 Lectures)**
	Processor Control, Hardwired Implementation, Micro-programmed Control. Parallel Architecture: Classification of Parallel Systems, Flynn's Taxonomy, Array Processors, Clusters, and NUMA Computers. Multiprocessor Systems: Structure & Interconnection Networks, Multi-Core Computers: Introduction, Organization and Performance.

Learning Resources recommended

Textbook(s):

1. M. Mano, Computer System Architecture 3rd edition, Pearson
2. Carl Hamacher et al., Computer Organization and Embedded Systems, 6 ed., McGraw-Hill 2012
3. R P Jain, Modern Digital Electronics, Tata McGraw Hill Education Pvt. Ltd. , 4th Edition, 2010

Additional Reference(s):

1. William Stallings (2010), Computer Organization and Architecture- designing for performance,8th edition, Prentice Hall, New Jersy.
2. Anrew S. Tanenbaum (2006), Structured Computer Organization, 5th edition, PearsonEducation Inc,
3. John P. Hayes (1998), Computer Architecture and Organization, 3rd edition, Tata McGrawHill

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester I
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Major Course

Name of the Course	**Fundamentals of Programming Paradigm**
Course Code	23_USCS102
Class	FYBSC
Semester	I
No of Credits	2
Nature	Theory
Type	Major
Employability/ entrepreneurship/ skill development	This course is new and concentrates on various programming concepts. General overview of programming concepts is introduced in this course. Also the course works on logic building of learners. Main thought behind this course is that 'any learner will get structure of various types/ formats of languages.' This will help learners to learn any language in future.

Fundamentals of Programming Paradigm
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Functional Programming, Logical Programming	10
2	Object Oriented Programming	10
3	Scripting Language	10
Total		**30**

Course Outcomes
The learner will be able to **CO1:** To understand logical programming principles, allowing them to use logical rules to solve problems efficiently. **CO2:** Understands proficiency in functional programming concepts, as well as logical programming fundamentals **CO3:** Possess a comprehensive understanding of Object-Oriented Programming principles and scripting languages, including their basic concepts, applications, and usage in various problem domains and web scripting environments.

Sr. No.	Modules / Units
1	**Fundamental of Programming**
	Logic Development - Overview of Computers and Logic, Introduction to logic building, Algorithm development, Understanding Program Structure and Design. General Terminology - literals , variables, scope of variable, operators, Special Symbols and its use, Wildcards and its use, Constants variables, static variables, Interface, User Interface, Data, Data Manipulation-Add, Delete, View, Update. Introduction to Control Statement- Various control statement, Loop, Breaking of Loop, Interface, User Interface Types of Language and language hierarchy
2	**Functional Programming and Logical Programming**
	Functional Programming: Language design, why to study programming language, compilation and interpretation, programming environments. Definition of a function: domain and range, total and partial functions, strict functions. Recursion, Referential transparency, Side effects of functions Logical Programming: Basic constructs, Facts: queries, existential queries, conjunctive queries and rules. Definition and semantics of a logic program, Recursive programming: Computational model of logic programming, Goal reduction, Negation in logic programming
3	**Object Oriented Programming and Scripting Language**
	Object Oriented Programming: Basic concepts: objects, classes, methods, overloading methods, messages inheritance: overriding methods, single inheritance, multiple inheritance Interfaces, encapsulation, polymorphism. Scripting Language: What is scripting language, Problem domain(Shell languages, Text processing and report generation, Mathematics and statistics, General purpose scripting, Extension languages), Scripting the world wide web(CGI scripts, Embedded server side script, client side script, Java Applets, XSLT,Python)

Learning Resources recommended

Textbook(s):

- Programming Language Pragmatics 4th edition, Michael Scott, Morgan Kaufmann

Additional Reference(s):

- "Foundations of Programming Languages Design & Implementation 3rd Editon . Roosta Seyed Cenage Learning

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester I
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Major Course

Name of the Course	**Computer Science Practical 1**
Course Code	23_USCS103
Class	FYBSC
Semester	I
No of Credits	2
Nature	Practical
Type	Major
Employability/ entrepreneurship/ skill development	Practical approach is given to understand circuit design and basic programming concepts. For circuit design any simulator can be used. This course also introduces algorithmic development of various problems which will get constructed using concepts taught in theory.

Course Outcomes
The learner will be able to **CO1 :** Develop proficiency in fundamental algorithms and problem-solving techniques using python. **CO2 :** Students will master essential JavaScript programming skills, including basic mathematical operations, number comparisons etc. **CO3 :** Demonstrate proficiency in logic gate analysis, Boolean expression simplification, design, and verification of adders, subtractors, comparators, and flip-flops in combinational and sequential circuits. **CO4 :** Demonstrate competence in verifying counters and shift registers, designing expressions with multiplexers/demultiplexers, implementation of flip-flops, and writing microprocessor programs

Sr. No.	**Practical Title**
The following practical's will be implemented using Logisim and python. Minimum 20 practical's to be completed as a journal work.	
1	Write pseudo code to detect whether a number is even or odd.
2	Write pseudo code for calculating whether no is prime or not.
3	Write a pseudocode for calculating the sum of 10 numbers.
4	Write a pseudo code for calculating Fibonacci series.

5	Demonstrate and Implement Recursive function for calculating square of a number
6	Demonstrate and Implement Recursive function for calculating factorial of a number
7	Write a JavaScript for basic mathematical functions (add, subtract, multiply and division)
8	Write a JavaScript to compare two numbers.
9	Write a JavaScript to print the factorial of a number.
10	Write a JavaScript to change background colour property.
11	Study and verify the truth table of various logic gates (NOT, AND, OR, NAND, NOR, EX-OR, and EX-NOR).
12	Simplify given Boolean expression and realize it.
13	Design and verify a half/full adder
14	Design and verify half/full subtractor
15	Design a 4 bit magnitude comparator using combinational circuits.
16	Design and verify the operation of flip-flops using logic gates.
17	Verify the operation of a counter.
18	Verify the operation of a 4 bit shift register
19	Design and implement expression using multiplexers / demultiplexers.
20	Design and implement 3-bit binary ripple counter using JK flip flops.
21	Simple microprocessor programs for data transfer operations
22	Simple microprocessor programs for arithmetic & logical transfer operations

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester I
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Minor Course

Name of the Course	**Applied Mathematics I**
Course Code	23_USCS104
Class	FYBSC
Semester	I
No of Credits	2
Nature	Theory
Type	Minor
Employability/ entrepreneurship/ skill development	This theoretical course will be more calculative and interesting for students. This course emphasizes on basic mathematics concepts which are required for understanding concepts of computer science. Many concepts are required and due course of time it is impossible to cover all concepts. Hence mathematical concepts are divided into three parts which are completed in a cascading semester.

Applied Mathematics I
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Set Theory, Functions	10
2	Relations, Recurrence Relations	10
3	Permutations and Combinations, Counting Principles	10
Total		**30**

Course Outcomes
The learner will be able to **CO1:** Demonstrate proficiency in analyzing sets, relations, functions, and their applications in problem-solving and mathematical reasoning. **CO2:** Understands the concepts of relations, as well as develop skills in solving and analyzing recurrence relations. **CO3:** Demonstrate proficiency in applying fundamental counting techniques to solve problems involving permutations, combinations, and advanced counting scenarios.

Sr. No.	Modules / Units
1	**Set Theory, Functions (10 Lectures)**
	Language of sets, Language of relation and functions, Definitions Properties of Sets, Algebraic & Boolean Definitions. Definition of function; Domain, co-domain, range of a function; Examples of standard functions such as identity and constant functions, absolute value function, logarithmic and exponential functions, flooring and ceiling functions; Injective, surjective and bijective functions; Composite and inverse functions.
2	**Relations, Recurrence Relations (10 Lectures)**
	Definition and examples of relation; Properties of relations, Representation of relations using digraphs and matrices; Equivalence relation; Partial Order relation, Hasse Diagrams, maximal, minimal, greatest, least element, Lattices. Definition and Formulation of recurrence relations; Solution of a recurrence relation; Solving recurrence relations- Backtracking method, Linear homogeneous recurrence relations with constant coefficients; Homogeneous solution of linear homogeneous recurrence relation with constant coefficients; Applications- Formulate and solve recurrence relation for Fibonacci numbers, Tower of Hanoi, Intersection of lines in a plane, Sorting Algorithms.
3	**Permutations and Combinations, Counting Principles (10 Lectures)**
	Permutation without and with repetition; Combination without and with repetition; Binomial numbers and identities: Pascal Identity, Vandermonde‟s Identity, Pascal triangle, Binomial theorem (without proof) and applications; Multinomial numbers, Multinomial theorem (without proof) and applications Basic Counting Principles (Sum and Product Rule); Pigeonhole Principle (without proof) - Simple examples; Inclusion Exclusion Principle (Sieve formula) (without proof); Counting using Tree diagrams.

Learning Resources recommended

1. Discrete Mathematics and Its Applications, Seventh Edition by Kenneth H. Rosen, McGraw Hill Education (India) Private Limited. (2011)
2. Discrete Mathematics: SemyourLipschutz, Marc Lipson, Schaum‟s out lines, McGraw-Hill Inc, 3rd edition

Additional References:

1. Elements of Discrete Mathematics: C.L. Liu, Tata McGraw- Hill Edition.
2. Concrete Mathematics (Foundation for Computer Science): Graham, Knuth, Patashnik Second Edition, Pearson Education.
3. Discrete Mathematics: SemyourLipschutz, Marc Lipson, Schaum‟s out lines, McGraw-Hill Inc.
4. Foundations in Discrete Mathematics: K.D. Joshi, New Age Publication, New Delhi.

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester I
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Minor Course Practical

Name of the Course	**Statistics using R Practical**
Course Code	23_USCS105
Class	FYBSC
Semester	I
No of Credits	2
Nature	Practical
Type	Minor
Employability/ entrepreneurship/ skill development	R is a programming language that is becoming very popular in the data analytics and data science field. It provides the foundation to the students on elementary topics of Statistics and prepares them to describe the given data. The students try to know and measure the chance of different events happening and their occurrence numerically. Students understand how probability has been distributed to the different events and standard notions of probability distributions. Students will be able to understand statistical concepts through the implementation in the programming language.

Statistics using R Practical
Modules at a Glance

Course Outcomes
The learner will be able to **CO1 :** Understand how to describe the data with available measures **CO2 :** Learn how to implement statistics concept using R language **CO3 :** Apply probability in real time situations and identify randomness in experiments **CO4 :** Differentiate between types of random variables and its distributions **CO5 :** Study the standard distributions and its properties

Statistics using R Practical

Sr. No.	**Practical Title**
The following practical's will be implemented using R studio. Minimum 20 practical's to be completed as a journal work.	
1	Frequency distribution and data presentation using R Programming
2	Create a Measures of central tendency using R Programming

3	Data entry using, functions, c(), scan (), Creating vectors, Mathematical Operations: ** +/-/*/ / ^ , exp, log, log10, etc, matrix operations, seq(), split()
4	Creating vector of text type, useful functions: data, frame R Programming
5	Implement Frequency distribution using cut(), table() and Data presentation R program.
6	Understanding Arithmetic mean, Median, mode: grouping and ungrouping data, effect of shift of origin.
7	Using R can read and write into various file formats like csv, excel, xml etc.
8	Create a grouped bar chart using R programming language.
9	Perform Skewness and Kurtosis in R Programming
10	Create histogram in R programming language.
11	Create Boxplots in R Programming Language.
12	Implement various functions to generate binomial distribution.
13	Implement Kruskal-Wallis test in R programming.
14	Implement simple linear regression in R programming.
15	Implement multiple linear regression in R programming
16	Calculate Sampling Distributions in R programming.
17	Perform one way ANOVA test in r programming.
18	Perform Exception Error handling in R programming.
19	Perform Exception Condition handling in R programming.
20	Represent data using frequency distribution, Histogram, Ogive curve.
21	Implementation of the concept of measures of central tendency.
22	Recognizing Partition values such as Quartiles, Deciles and Percentiles.
23	Importing/Reading Files in R, Exporting/Writing Files in R.

Learning Resources recommended

Textbooks:

1. R Programming : Statistical Analysis with R For Dummies Paperback – 16 May 2017
2. Trivedi, K.S.(2001) : Probability, Statistics, Design of Experiments and Queuing theory, with applications of Computer Science, Prentice Hall of India, New Delhi

Additional References:

1. Kulkarni, M.B., Ghatpande, S.B. and Gore, S.D. (1999): common statistical tests. Satyajeet Prakashan, Pune
2. Gupta, S.C. and Kapoor, V.K. (1987): Fundamentals of Mathematical Statistics, S. Chand and Sons, New Delhi
3. Gupta, S.C. and Kapoor, V.K. (1999): Applied Statistics, S. Chand and Son‟s, New Delhi

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester I
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Generic / Open Elective (OE) Course

Name of the Course	**Social Media Marketing**
Course Code	23_USOE106
Class	FYBSC
Semester	I
No of Credits	2
Nature	Theory
Type	OE
Employability/ entrepreneurship/ skill development	Enhance Your Employability and Entrepreneurial Skills! 1. Unlock your potential with our comprehensive social media marketing program! 2. Boost your employability: Gain in-demand skills that employers crave in today's digital world. 3. Unleash your entrepreneurial spirit: Learn how to leverage social media platforms to create and grow your own successful business. 4. Master the art of engagement: Discover effective strategies to captivate your audience, increase brand visibility, and drive conversions. 5. Stay ahead of the competition: Stay up-to-date with the latest trends and best practices in social media marketing.

Introduction to Social Media
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Introduction to Social Media	10
2	Understanding Social Media and tools	10
3	Introduction to Social Media Marketing	10
Total		**30**

Course Outcomes
The learner will be able to CO1: To know the importance of Social media Platforms and importance of Social media in Digital Marketing. CO2: Demonstrate an understanding of social media, the various channels through which it operates, and its role in marketing strategy. CO3: Identify globally acceptable best practices for digital and social media marketing.

Sr. No.	Modules / Units
1	**Introduction to Social Media (10 Lectures)**
	What is Social Media & its significance? , Necessity of Social media, The changing face of social media, Social Media Past and Present, Who owns the social Media, Classification of Social Media, Identification of social media, Profile Creation, expanding the network, Engagement.
2	**Understanding Social Media and tools (10 Lectures)**
	Monitoring and Analysis, Needs of Social Media in business, Benefits of social media for individuals, Benefits of social media for Business, Social Networking, Photo and Video Sharing, Blogs, Micro Blog, Social Curation, Reviews and Ratings, Location, Business Networking, Social Gaming.
3	**Introduction to Social Media Marketing (10 Lectures)**
	History of Social Media Marketing, Importance of Social Media, Facebook Marketing, LinkedIn Marketing, Pinterest Marketing, Video Marketing, Social Bookmarking, Image Optimization.

Learning Resources recommended
Textbook(s): 1. Understanding Social Media by Varinder Taprial, Priya Kanwar **Additional Reference(s):** 2. Social Media Marketing by tutorialspoint.com

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester I
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)

Indian Knowledge System (IKS) Course

Name of the Course	**Evolution of Computing in India**
Course Code	23_USCS106
Class	FY
Semester	I
No of Credits	02
Nature	Theory
Type	IKS
Employability/ entrepreneurship/ skill development	During India's computer industry infancy, a blend of technical expertise, visionary leadership, and strategic planning was vital. Advocacy for computers' societal impact was key, driving support and resource allocation. Efficient management of resources was crucial for establishing initial infrastructure. Policies emphasizing education in computing bolstered workforce skill. Proficiency in policy formulation and economic analysis guided sustainable growth. Anticipating tech trends and fostering innovation ensured industry relevance. Adapting to international regulations facilitated global operations. Contributions to an innovation ecosystem were made through research and technology solutions.

Evolution of Computing in India
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Computing Science in Ancient India	10
2	Contribution in Computing in 19th Century	10
3	Government Policies and Status of IT industry	10
Total		**30**

Course Outcomes
The learner will be able to **CO1** : Gain insights into the historical contributions and advancements in computing science from ancient Indian civilizations. **CO2 :** Explore and understand key contributions to computing during the 19th century, shaping the foundation of modern computational theory and technology. **CO3:** Analyze the impact of government policies on the IT industry and assess the current status and trends within the sector.

Sr. No.	Modules / Units
1	**Computing Science in Ancient India (10 Lectures)**
	An overview of Ancient Indian Science - Vedic Cognitive science, mathematical and physical science, Scientist of ancient India, Binary numbers in Indian Antiquity -Pingala's classification of Meters, Encode decode system in Ancient India- Katayapadi Scheme, Panini's contribution in computing -The Panini-Backus Form, Fowlers' automaton .
2	**Contribution in Computing in 19th Century (10 Lectures)**
	Laying the foundation, First computer in India, TIFRAC, Self-reliant growth of computer industry, ECIL's computer division, PARAM super computer, NASSCOM, Contribution of CDAC, Development of multilingual software. Indian computer scientist, Indian IT industrialist.
3	**Government Policies and Status of IT industry (10 Lectures)**
	Economy of India, Government initiative, Role of government institution & Ministry of Information Technology, Government Policies at early stage and revised policies, Consequences of Government policies., Entry of private sector in computer industry, Liberalization of economy and software expert growth, The period of rapid growth of IT industry, Indian IT industry in global perspective, Indian ICT industry in global perspective, Future of IT industry

Learning Resources recommended
Book(s): **Learning Resources Recommended:** 1. "Computing science in Ancient India", T.R.N. Rao / Subhash Kak 2. "History of Computing in India 1955-2010", V. Rajaraman, IEEE Computer Society 3. "Homi Bhabha and the Computer Revolution", R.K. Shyamsunder and M. A. Pai, Oxford University, 2011 **Additional references:** 1. https://www.scribd.com/document/57187049/History-of-Computer-Technology-in-India# 2. https://en.m.wikipedia.org/wiki/TCIFRAC 3. https://en.m.wikipedia.org/wiki/Information_technology_in_India

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester I
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Vocational Skill Course (VSC)

Name of the Course	**Programming with Python Practical**
Course Code	23_USCS107
Class	F.Y.B.Sc.
Semester	I
No of Credits	2
Nature	Practical
Type	VSC
Employability/ entrepreneurship/ skill development	In the past curriculum, Programming with Python was a course based on theory and practical approach. Python is a language which will be learned with hands-on training. Hence to develop practical approach the more trace is given on practical in this course. This will help learners to become more practically sound in the programming language.

Course Outcomes
The learner will be able to **CO1 :** Gain proficiency in Python fundamentals, including using Python IDLE and Shell, working with literals, variables, I/O functions, operators, and implementing simple programs. **CO2 :** Implement Python control flow statements, explore modules, List, Dictionary, and Tuple data structures. **CO3 :** Proficiently implement functions, exceptions, manage files and databases, and develop graphical user interfaces (GUIs) in Python. **CO4 :** Grasp networking fundamentals and object-oriented programming concepts.

Sr. No.	**Aim of the Practical**	**Literature**
The following practical's will be implemented using python, Minimum 20 practical's to be completed as a journal work.		
1	Introduction to Python IDLE, Python Shell	Information about IDLE, Diagram of IDLE, Menus and three commands delivering information about python
2	Understanding of Literal and variables	Information about literal (integer, float, character, string, etc…) and variable
3	Illustration of I/O functions such as print (), int(), float() functions	Syntax and examples of all functions
4	Use of Operators in Python	Information about all Operators
5	Implementation of simple programs	All the programs with statements containing basic I/O functions, operators, etc.
6	Understanding if statement	Syntax of if statement, if else statement, elif ladder, programs examining all types of if statement

7	Understanding while statement	Syntax of while statement, programs examining while statement
8	Understanding for statement	Syntax of for statement, range() function, programs examining for statement.
9	Understanding break, continue statement	Syntax for break, continue statement in if, while and for statements. All types of programs examining break, continue statement
10	Understanding modules in python	Concept of modules, examining various modules such as math module, random module, time module
11	Implementation of List data structure in python	Information of list and all operations performed on list, programs on list and list operation
12	Implementation of Dictionary data structure in python	Information of dictionary and all operations performed on dictionary, programs on dictionary and dictionary operation
13	Implementation of Tuple data structure in python	Information about tuple and demonstrative program on Tuple
14	Implementation of Functions	Syntax of function definition, function call, concept of local and global parameters and programs on functions
15	Illustration of Exception handling	Demonstration of programs on exception handling
16	File handling	File handling concepts and programs to examine file handling concepts
17	OOP concept	Types of Methods and Polymorphism Types of Methods: Instance Methods, Class Methods, and Static Methods Polymorphism: Method Overriding and Operator Overloading
18	OOP concept	Inheritance and Abstract Methods/Classes Concepts Covered: Single, Multiple, and Multilevel Inheritance Abstract Classes and Methods (Using ABC module)
19	OOP concept	Interfaces and Advanced OOP: Interfaces (Python does not have built-in interfaces, but they can be simulated using abstract classes) Demonstrate interface-based programming
20	GUI programming	GUI Concepts and Widgets in Python. Introduction to GUI programming in Python using Tkinter. Implementation of different widgets such as Labels, Buttons, Entry fields, Checkboxes, and Radio Buttons. Creating a simple application that incorporates multiple widgets.
21	GUI programming	Layout Managers in GUI Programming.

		Understanding layout managers in Python GUI frameworks. Demonstrating different layout managers like Pack, Grid, and Place in Tkinter. Designing an application that effectively organizes widgets using layout managers.
22	GUI programming	Event Handling, Fonts, and colours in GUI Introduction to event-driven programming in Python GUI. Demonstration of changing fonts and colours dynamically using GUI controls. Handling different events like button clicks, keyboard inputs, and mouse events.
23	Database handling	Database concepts, CRUD operations Write a Python Program to work with databases in Python to perform operations such as a. Connecting to database b. Creating and dropping tables c. Inserting and updating into tables.

Learning Resources recommended

Textbooks:

1. Practical Programming: An Introduction to Computer Science Using Python 3, Paul Gries , Jennifer Campbell, Jason Montojo, Pragmatic Bookshelf, 2nd Edition, 2014
2. Programming through Python, M. T Savaliya, R. K. Maurya& G M Magar, Sybgen Learning India, 2020

Additional References:

1. Python: The Complete Reference, Martin C. Brown, McGraw Hill, 2018
2. Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress, 2017
3. Programming in Python 3, Mark Summerfield, Pearson Education, 2nd Ed, 2018
4. Python Programming: Using Problem Solving Approach, ReemaThareja, Oxford Univeristy Press, 2017
5. Let Us Python, Yashwant. B. Kanetkar, BPB Publication, 2019

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester I
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Ability Enhancement Course (AEC)

Name of the Course	**English (Soft Skill Development -I)**
Course Code	23_USCS108
Class	F.Y.BSc
Semester	I
No of Credits	2
Nature	Theory
Type	AEC
Employability/ entrepreneurship/ skill development	The main aim of the paper is to learn basic communication skills required in the IT industry. To help learners develop their soft skills and develop their personality along with technical skills. Focus on various communication enhancement along with academic and professional ethics.

English Soft Skill Development - I
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Soft Skills: An Introduction, Personality Development, Communication Skills, Etiquettes and Mannerism	10
2	Employment Communication, Job Interviews, Group Discussion	10
3	Professional Presentation, Leadership and Team Building, Stress and Time Management	10
Total		**30**

Course Outcomes
The learner will be able to **CO1:** Cultivate a robust skill set encompassing effective communication, refined etiquette, and personality enhancement, facilitating professional and personal growth. **CO2:** Demonstrate proficiency in crafting compelling resumes, mastering job interview techniques, and excelling in group discussions, enhancing their employability and career advancement prospects. **CO3:** Develop expertise in delivering impactful presentations, fostering leadership skills for effective team building, and mastering stress and time management techniques, empowering them for success in professional endeavors.

Sr. No.	Modules / Units
1	**Soft Skills: An Introduction, Personality Development, Communication Skills, Etiquettes and Mannerism (10 Lectures)**
	Definition and Significance of Soft Skills; Process, Importance and Measurement of Soft Skill Development, Identifying your soft skills, SWOT analysis Knowing Yourself, Positive Thinking, Johari's Window, Effective communication Spoken English, Phonetics, Accent, Intonation Introduction, Professional Etiquette, Technology Etiquette.
2	**Employment Communication, Job Interviews, Group Discussion (10 Lectures)**
	Introduction, Resume, Curriculum Vitae, Scannable Resume, Developing an Impressive Resume, Formats of Resume, Job Application or Cover Letter Introduction, Importance of Resume, Definition of Interview, Background Information, Types of Interviews, Preparatory Steps for Job Interviews, Interview Skill Tips, Introduction, Ambience/Seating Arrangement for Group Discussion, Importance of Group Discussions, Types of Group Discussions
3	**Professional Presentation, Leadership and Team Building, Stress and Time Management (10 Lectures)**
	Nature of Oral Presentation, planning a Presentation, Preparing the Presentation, Delivering the Presentation Leader and Leadership, Leadership Traits, Culture and Leadership, Leadership Styles and Trends, Team Building, Types of Teams. Stress, Sources of Stress, Ways to Cope with Stress.

Learning Resources recommended

Textbook(s):

1. Managing Soft Skills for Personality Development – edited by B.N.Ghosh, McGraw Hill India, 2017.
2. Soft Skills: An Integrated Approach to Maximize Personality, Gajendra S. Chauhan, Sangeeta Sharma, Wiley India

Additional References:

1. Personality Development and Soft Skills, Barun K. Mitra, Oxford Press
2. Business Communication, ShaliniKalia, Shailja Agrawal, Wiley India
3. Cornerstone: Developing Soft Skills, Sherfield, Pearson India

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester I
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Value Education Course (VEC)

Name of the Course	**Green Computing I**
Course Code	23_USCS109
Class	F.Y.BSc
Semester	I
No of Credits	2
Nature	Theory
Type	VEC
Employability/ entrepreneurship/ skill development	Being a green in the IT field is a need in today's era. Harmful components present in the system may cause damage to the environment if not properly disposed of. Hence to give a more concentration on this concern, syllabus tried to cover all issues related with green technology. This vast topic is covered in two parts. Second semester will be a continuation of the first semester.

Green Computing I
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Green IT Overview, Green Devices and Hardware	10
2	Green Software, Sustainable Software Development	10
3	Green Data Centers, Green Data Storage	10
Total		**30**

Course Outcomes
The learner will be able to **CO1:**Understand sustainable computing practices, including the design, implementation, and utilization of energy-efficient devices and hardware, contributing to environmental responsibility in IT. **CO2:**Implement eco-friendly software solutions and apply sustainable development practices to minimize environmental impact.

CO3: Gain expertise in designing and managing environmentally friendly data centers and storage solutions, minimizing energy consumption and reducing carbon footprint in data management infrastructure.

Sr. No.	Modules / Units
1	**Green IT Overview, Green Devices and Hardware (10 Lectures)**
	Introduction, Environmental Concerns and Sustainable Development, Environmental Impacts of IT, Green I , Holistic Approach to Greening IT, Greening IT, Applying IT for Enhancing Environmental Sustainability, Green IT Standards and Eco-Labelling of IT , Enterprise Green IT Strategy, Green Washing, Green IT: Burden or Opportunity? Introduction , Life Cycle of a Device or Hardware, Reuse, Recycle and Dispose
2	**Green Software, Sustainable Software Development (10 Lectures)**
	Introduction, Processor Power States , Energy-Saving Software Techniques, Evaluating and Measuring Software Impact to Platform Power Introduction, Current Practices, Sustainable Software, Software Sustainability Attributes, Software Sustainability Metrics, Sustainable Software Methodology, Defining Actions
3	**Green Data Centers, Green Data Storage (10 Lectures)**
	Data Centers and Associated Energy Challenges, Data Centre IT Infrastructure, Data Centre Facility Infrastructure: Implications for Energy Efficiency, IT Infrastructure Management, Green Data Centre Metrics Introduction , Storage Media Power Characteristics, Energy Management Techniques for Hard Disks, System-Level Energy Management

Learning Resources recommended

Book(s):

1. Harnessing Green IT: Principles and Practices, San Murugesan, G. R. Ganadharan, Wiley & IEEE.

Additional References:

1. Green IT, Deepak Shikarpur, Vishwkarma Publications, 2014
2. Green Communications: Principles, Concepts and Practice- Samdanis et al, J. Wiley
3. Green IT for Sustainable Business Practice: An ISEB Foundation Guide, Mark G. O'Neill, The Chartered Institute for IT, 2010

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester I
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Minor Course

Name of the Course	**Computer Organization and Architecture**
Course Code	24_USCS110
Class	FYBSC
Semester	I
No of Credits	2
Nature	Theory
Type	Minor
Employability/ entrepreneurship/ skill development	This theoretical course will be help for a strong understanding of computer organization and architecture can pursue roles as hardware engineers, designing and developing computer systems, processors, and other electronic components. Computer architects focus on designing and evaluating computer systems and components. They analyze performance metrics and develop strategies to improve system performance and efficiency.

Computer Organization and Architecture
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	**Structure of Computers , Computer Arithmetic**	10
2	**Basic Computer Organization and Design, Register Transfer and Micro-Operations, Micro-Programmed Control**	10
3	**Memory System , Input Output, Multiprocessors**	10
Total		**30**

Course Outcomes
The learner will be able to **CO1:-** To Understand fundamental concepts such as data representation, instruction execution cycle, and memory hierarchy. **CO2:-**To Understand the functions and roles of registers in instruction execution and data manipulation. **CO3:-**To Analyze the performance impact of memory hierarchy, cache size, cache mapping, and cache replacement policies on computer systems.

Sr. No.	Modules / Units
1	**Structure of Computers , Computer Arithmetic (10 Lectures)**
	Computer types, Functional units, Basic operational concepts, VonNeumann Architecture, Bus Structures, Software, Performance, Multiprocessors and Multicomputer, Data representation, Fixed and Floating point, Error detection and correction codes. Addition and Subtraction, Multiplication and Division algorithms, Floating-point Arithmetic Operations, Decimal arithmetic operations. Data representation, Fixed and Floating points, Addition, subtraction, Multiplication, division binary operation. Conversion of number system (Decimal to binary, Decimal to octal, Decimal to hexadecimal, Binary to octal, Binary to hexadecimal and vice versa)
2	**Basic Computer Organization and Design, Register Transfer and Micro-Operations, Micro-Programmed Control (10 Lectures)**
	Instruction codes, Computer Registers, Computer Instructions and Instruction cycle. Timing and Control, Memory-Reference Instructions, Input-Output and interrupt. Central processing unit: Stack organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Complex Instruction Set Computer (CISC) Reduced Instruction Set Computer (RISC), CISC vs RISC Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro-Operations, Logic Micro-Operations, Shift Micro-Operations, Arithmetic logic shift unit. Control Memory, Address Sequencing, Micro-Program example, Design of Control Unit.
3	**Memory System , Input Output, Multiprocessors (10 Lectures)**
	Memory Hierarchy, Semiconductor Memories, RAM(Random Access Memory),Read Only Memory (ROM), Types of ROM, Cache Memory, Performance considerations, Virtual memory, Paging, Secondary Storage, RAID. I/O interface, Programmed IO, Memory Mapped IO, Interrupt Driven IO, DMA. Characteristics of multiprocessors, Interconnection structures, Inter Processor Arbitration, Inter processor Communication and Synchronization, Cache Coherence

Learning Resources recommended

1. M. Moris Mano (2006), Computer System Architecture, 3rd edition, Pearson/PHI, India **.**

Additional References:

1. Carl Hamacher, Zvonks Vranesic, SafeaZaky (2002), Computer Organization, 5th edition, McGraw Hill, New Delhi, India.
2. William Stallings (2010), Computer Organization and Architecture- designing for performance, 8th edition, Prentice Hall, New Jersy.
3. Anrew S. Tanenbaum (2006), Structured Computer Organization, 5th edition, Pearson Education Inc, 4. John P. Hayes (1998), Computer Architecture and Organization, 3rd edition, Tata McGrawHill

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester I
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Minor Course

Name of the Course	**Programming with C**
Course Code	24_USCS111
Class	FYBSC
Semester	I
No of Credits	2
Nature	Theory
Type	Minor
Employability/ entrepreneurship/ skill development	Course takes a main aim on logic development using C language. For logic development algorithm formation will be taught which helps learners to develop programs in C language easily. Basic concepts of C language are covered in this course. Since the main aim of the course is to make student aware with programming language hence advance topics like memory handling, file handling, etc… are excluded from syllabus.

Programming with C
***Modules at a* Glance**

Sr. No.	**Modules**	**No. of Lectures**
1	**Fundamentals of algorithms:,Algorithmic problems, Fundamentals of C Programming**	10
2	**Control Structures (Branching and looping structures), Functions**	10
3	**Arrays and Strings,Structure and Union**	10
	Total	**30**

Course Outcomes
The learner will be able to **CO1:**To Formulate simple algorithms for arithmetic, logical problems and translate them to programs in C language **CO2:** Implement, test and execute programs comprising of control structures and Decompose a **CO3:** problem into functions and synthesize a complete program. Demonstrate the use of arrays, strings and structures in C language.

Sr. No.	Modules / Units
1	**Fundamentals of algorithms:,Algorithmic problems, Fundamentals of C Programming (10 Lectures)**
	Notion of an algorithm. Pseudo-code conventions like assignment statements and basic control structures. Develop fundamental algorithms for (i) Exchange the values of two variables with and without temporary variable, (ii) Counting positive numbers from a set of integers, (iii) Summation of set of numbers, (iv) Reversing the digits of an integer, Keywords, Identifiers, Constants and Variables , Data types in C , Operators in C ,Basic Input and Output Operations ,Expressions and Precedence of Operators, In-built Functions
2	**Control Structures (Branching and looping structures), Functions (10 Lectures)**
	Introduction to Control Structures, If statement, If-else statement, Nested if-else, else-if Ladder , Switch statement , For loop, While loop, Do while loop , break and continue Introduction to functions, Function prototype, Function definition, Accessing a function and parameter passing. , Recursion.
3	**Arrays and Strings,Structure and Union (10 Lectures)**
	Introduction to Arrays , Declaration and initialization of one dimensional and two-dimensional arrays. Definition and initialization of String , String functions Concept of Structure and Union, Declaration and Initialization of structure and union, Nested structures ,Array of Structures, Passing structure to functions

Learning Resources recommended
Text Books: • E. Balaguruswamy, Programming in ANSI C, McGraw-Hill • Kernighan , Ritchie, "The C programming Language", Prentice Hall of India • Sumitabha Das, Computer Fundamentals and C Programming, McGraw-Hill • Pradeep Day and ManasGosh ,"Programming in C", Oxford University Press. **References:** • Byron Gottfried, "Programing with C", McGraw Hill (Schaum"s outline series) • Venugopal K.R, Prasad Sudeep, "Mastering C", McGraw-Hill • KanetkarYashwant," "Let Us C", BPB Publication.

Semester II

Revised Syllabus of Courses of B.Sc. Computer Science

Programme at Semester II

with Effect from the Academic Year 2023-2024

Department Specific Courses (DSC)

Major Course

Name of the Course	**Database system**
Course Code	23_USCS201
Class	FYBSC
Semester	II
No of Credits	2
Nature	Theory
Type	Major
Employability/ entrepreneurship/ skill development	Database is a core concept that every IT professional needs to know. All the basic concepts of database are included in this course. This course will make learners aware about the database concepts and they can learn advanced concepts. Due to the course schedule only necessary concepts are mentioned in this course. Indexes and views are taken as a part of theory. Data models which are included in previous course are omitted as per time constraints. Concepts which are used frequently and need more attention are included in course.

Database system
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Introduction to DBMS,Entity Relationship Model,DDL Statements DML Statements	10
2	Relational data model, Relational Algebra, Functions ,Joining Tables, Sub-queries	10
3	Schema refinement and Normal forms, Database Protection , Transaction control commands	10
Total		**30**

Course Outcomes
The learner will be able to **CO1 :** Gain proficiency in DBMS fundamentals, including the Entity Relationship Model, DDL statements, and DML statements. **CO2 :** Understand relational data model, relational algebra, functions, table joining techniques, and sub-query usage in database management.

CO3 : Demonstrate proficiency in schema refinement, normalization up to higher normal forms, database protection mechanisms, and transaction control commands for ensuring data integrity and security in database management.

Sr. No.	Modules / Units
1	**Introduction to DBMS,Entity Relationship Model,DDL Statements DML Statements (10 Lectures)**
	Introduction to DBMS Database, DBMS – Definition, Overview of DBMS, Advantages of DBMS, Levels of abstraction, Data independence, DBMS Architecture Entity Relationship Model Entities, attributes, entity sets, relations, relationship sets, Additional constraints (key constraints, participation constraints, weak entities, aggregation / generalization, Conceptual Design using ER (entities VS attributes, Entity Vs relationship, binary Vs ternary, constraints beyond ER) DDL Statements - Creating Databases, Using Databases, datatypes, Creating Tables (with integrity constraints – primary key, default, check, not null), Altering Tables, Renaming Tables, Dropping Tables, Truncating Tables DML Statements Viewing the structure of a table insert, update, delete, Select all columns, specific columns, unique records, conditional select, in clause, between clause, limit, aggregate functions (count, min, max, avg, sum), group by clause, having clause
2	**Relational data model, Relational Algebra, Functions ,Joining Tables, Sub-queries (10 Lectures)**
	Relational data model : Domains, attributes, Tuples and Relations, Relational Model Notation, Characteristics of Relations, Relational Constraints - primary key, referential integrity, unique constraint, Null constraint, Check constraint Relational Algebra : Operations (selection, projection, set operations union, intersection, difference, cross product, Joins –conditional, equi join and natural joins, division) Functions : String Functions (concat, instr, left, right, mid, length, lcase/lower, ucase/upper, replace, strcmp, trim, ltrim, rtrim), Math Functions (abs, ceil, floor, mod, pow, sqrt, round, truncate) Date Functions (adddate, datediff, day, month, year, hour, min, sec, now, reverse) Joining Tables Inner join, outer join (left outer, right outer, full outer) Subqueries : Subqueries with IN, EXISTS, subqueries restrictions, Nested subqueries, ANY/ALL clause, correlated subqueries
3	**Schema refinement and Normal forms, Database Protection , Transaction control commands Architecture (10 Lectures)**

	Schema refinement and Normal forms : Functional dependencies, first, second, third, and BCNF normal forms based on primary keys, lossless join decomposition. Database Protection : Security Issues, Threats to Databases, Security Mechanisms, Role of DBA, Discretionary Access Control, Backing Up and Restoring databases Transaction control commands : Commit, Rollback

Learning Resources recommended
Textbooks: 1. "Fundamentals of Database System", ElmasriRamez, NavatheShamkant, Pearson Education, Seventh edition, 2017 2. "Database Management Systems", Raghu Ramakrishnan and Johannes Gehrke, 3rd Edition, 2014 3. "Murach's MySQL", Joel Murach, 3rd Edition, 3rd Edition, 2019 **Additional References:** 1. "Database System Concepts", Abraham Silberschatz,HenryF.Korth,S.Sudarshan, McGraw Hill, 2017 2. "MySQL: The Complete Reference", VikramVaswani , McGraw Hill,

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester II
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Major Course

Name of the Course	**Programming with C++**
Course Code	23_USCS202
Class	FYBSC
Semester	II
No of Credits	2
Nature	Theory
Type	Major
Employability/ entrepreneurship/ skill development	As being a new structure and abject oriented language; course delivers constructs of C++. Introduction of UML and concepts of UML such as class diagram will get included in software engineering which will get introduced in the upcoming year. This course illustrated implementation of object-oriented programming through C++. The highly configured objects are formed by taking the help of the C++. Hence the course 'Programming with C++' is included in the program.

Programming with C++
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Introduction to C++,Data Types, Data Input Output and Operators, Decision Making, Loops, Arrays and Strings, Classes, Abstraction & Encapsulation	10
2	Constructors and Destructors, Working with objects, Polymorphism	10
3	Inheritance, Pointers, File Handling	10
Total		**30**

Course Outcomes
The learner will be able to **CO1:** After successful completion of this course, students would be able to Work with numeric, character and textual data and arrays. **CO2:** Understand the importance of OOP approach over procedural language. **CO3:**Apply the concepts of OOPS like encapsulation, inheritance and polymorphism.

Handle basic file operations.

Sr. No.	Modules / Units
1	**Introduction to C++, Data Types, Data Input Output and Operators, Decision Making, Loops, Arrays and Strings, Classes, Abstraction & Encapsulation (10 Lectures)**
	Introduction to C++: Features of C++ and its basic structure, simple C++ program without class, compiling and running C++ program. Data Types, Data Input Output and Operators: Basic data types, variables, rules for naming variables, programming constants, the type cast operator, implicit and explicit type casting, cout and cin statements, operators, precedence of operators. Decision Making, Loops, Arrays and Strings: Conditional statements-if, if...else, switch loops- while, do...while, for, types of arrays and strings Classes, Abstraction & Encapsulation: Classes and objects, Dot Operator, data members, member functions, passing data to functions, scope and visibility of variables in function.
2	**Constructors and Destructors, Working with objects, Polymorphism (10 Lectures)**
	Constructors and Destructors: Default constructor, parameterized constructor, copy constructor, private constructor, destructors. Working with objects Accessor - mutator methods, static data and static function, access specifiers, array of objects. Polymorphism: Binding-static binding & overloading, constructor overloading function overloading, operator overloading, overloading unary and binary operators. Run time Polymorphism
3	**Inheritance,Pointers,File Handling (10 Lectures)**
	Inheritance: Defining base class and its derived class, access specifiers, types of inheritance-single, multiple, hierarchical, multilevel, hybrid inheritance, friend function and friend class, constructors in derived classes. Pointers: Introduction to pointers, * and & operators, assigning addresses to pointer variables, accessing values using pointers, pointers to objects & this pointer, pointers to derived classes File Handling: File Stream classes, opening and closing file-file opening modes, text file handling, binary file handling.

Learning Resources recommended
Textbooks: 1. Object Oriented Programming with C++, Balagurusamy E., 8th Edition, McGraw Hill Education India. 2. UML & C++: A Practical Guide to Object Oriented Development, Lee/Tepfenhart, Pearson Education, 2nd Edition2015 **Additional References:** 1. Mastering C++ by Venugopal, Publisher: McGraw-Hill Education, 2017 2. Let Us C++ by KanetkarYashwant, Publisher: BPB Publications, 2020 3. Object Oriented Analysis and Design by Timothy Budd TMH, 2001

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester II
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Major Course

Name of the Course	**Computer Science Practical 2**
Course Code	23_USCS203
Class	FYBSC
Semester	II
No of Credits	2
Nature	Practical
Type	Major
Employability/ entrepreneurship/ skill development	These course focuses on practical implementation of databases and C++. Giving hands on theory topics helps to understand theoretical concepts very easily.

Course Outcomes
The learner will be able to **CO1 :** Design ER diagrams, implement different database management operations with simple queries and functions. **CO2 :** proficiently apply subqueries with different clauses,, normalize databases, create and manipulate views, and demonstrate object-oriented programming principles. **CO3 :** Demonstrate proficiency in using scope resolution operators, constructors, destructors, and scope specifiers in object-oriented programming. **CO4 :** Implement proficiency in inheritance, constructors in derived classes, friend functions, inline functions, this pointer, function overloading/overriding, pointer usage, and file handling in C++.

Sr. No.	**Practical Title**
The following practical's will be implemented using MySQL and turbo C++/DevCpp. Minimum 20 practical's to be completed as a journal work.	
1	Conceptual Designing using ER Diagrams (Identifying entities, attributes, keys and relationships between entities, cardinalities, generalization, specialization etc.)
2	Perform the following: • Viewing all databases • Creating a Database • Viewing all Tables in a Database • Creating Tables (With and Without Constraints) • Inserting/Updating/Deleting Records in a Table
3	Perform the following: • Altering a Table • Dropping/Truncating/Renaming Tables • Backing up / Restoring a Database

4	Perform the following: • Simple Queries • Simple Queries with Aggregate functions
5	Queries involving • Date Functions • String Functions • Math Functions
6	Sub queries • With IN clause • With EXISTS clause
7	Converting ER Model to Relational Model and apply Normalization on database. (Represent entities and relationships in Tabular form, Represent attributes as columns, identifying keys and normalization up to 3rd Normal Form).
8	Views Creating Views (with and without check option) Dropping views Selecting from a view
9	Program to demonstrate use of data members & member functions.
10	Programs based on branching and looping statements using classes.
11	Program to demonstrate one- and two-dimensional arrays using classes
12	Program to use scope resolution operator. Display the various values of the same variables declared at different scope levels.
13	Programs to demonstrate various types of constructors and destructors.
14	Programs to demonstrate use of public, protected & private scope specifiers.
15	Programs to demonstrate single and multilevel inheritance
16	Programs to demonstrate multiple inheritance and hierarchical inheritance
17	Programs to demonstrate inheritance and derived class constructors
18	Programs to demonstrate friend function.
19	Programs to demonstrate inline function, this pointer
20	Programs to demonstrate function overloading and overriding.
21	Programs to demonstrate use of pointers
22	Programs to demonstrate text and binary file handling

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester II
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Minor Course

Name of the Course	**Applied Mathematics II**
Course Code	23_USCS204
Class	FYBSC
Semester	II
No of Credits	2
Nature	Theory
Type	Minor
Employability/ entrepreneurship/ skill development	This course tries to cover a few more concepts of mathematics which are lagged in semester I due to course schedule. Concentration of this course is on derivatives and integration. These concepts helps learners to understand machine learning, optimization concepts.

Applied Mathematics II
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Derivative, Partial Derivatives	10
2	Integration, Applications of Integration	10
3	Random Variables, Standard Probability distributions	10
Total		**30**

Course Outcomes
The learner will be able to **CO1 :** Introduce the basic tools of Calculus which are helpful in demonstrating proficiency in computing derivatives and partial derivatives of functions. **CO2 :** Master the techniques of integration and apply them to solve various real-world problems. **CO3 :** Understand and apply random variables and standard probability distributions in statistical analysis.

Sr. No.	Modules / Units
1	**Derivative, Partial Derivatives (10 Lectures)**
	Derivative In Graphing And Applications: Increase, Decrease, Concavity, Relative Extreme; Graphing Polynomials, Rational Functions, Cusps and Vertical Tangents. Absolute Maxima and Minima, Applied Maximum and Minimum Problems, Newton‟s Method. Partial Derivatives, Differentiability, Differentials, and Local Linearity, Chain Rule, Implicit Differentiation, Directional Derivatives and Gradients
2	**Integration, Applications of Integration (10 Lectures)**
	An Overview of the Area Problem, Indefinite Integral, Definition of Area as a Limit; Sigma Notation, Definite Integral, Evaluating Definite Integrals by Substitution, Numerical Integration: Simpson‟s Rule. Area between two curves, Length of a plane curve.
3	**Random Variables, Standard Probability distributions (10 Lectures)**
	Concept and definition of a discrete random variable and continuous random variable. Probability mass function, Probability density function and cumulative distribution function of discrete and continuous random variable, Properties of cumulative distribution function. Introduction, properties, examples and applications of each of the following distributions: Binomial distribution, Normal distribution, Chi-square distribution, t distribution, F distribution

Learning Resources recommended

Learning Resources recommended:

1. Calculus: Early transcendental (10th Edition): Howard Anton, IrlBivens, Stephen Davis, John Wiley & sons, 2012.
2. Gupta, S.C. and Kapoor, V.K. (1987): Fundamentals of Mathematical Statistics, S. Chand and Sons, New Delhi
3. Goon, A. M., Gupta, M. K. and Dasgupta, B. (1983). Fundamentals of Statistics, Vol. 1, Sixth Revised Edition, The World Press Pvt. Ltd., Calcutta

Additional References:

1. Mood, A. M. and Graybill, F. A. and Boes D.C. (1974). Introduction to the Theory of Statistics, Ed. 3, McGraw Hill Book Company.
2. Hoel P. G. (1971). Introduction to Mathematical Statistics, John Wiley and Sons, New York.
3. Hogg, R.V. and Craig R.G. (1989). Introduction to Mathematical Statistics, Ed. MacMillan Publishing Co., New York.
4. Walpole R. E., Myers R. H. and Myers S. L. (1985), Probability and Statistics for Engineers and Scientists

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester II
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Minor Course

Name of the Course	**Data Structures**
Course Code	23_USCS205
Class	FYBSC
Semester	II
No of Credits	2
Nature	Theory
Type	Minor
Employability/ entrepreneurship/ skill development	The course focuses to give an understanding of different types of data structures that can be used to store data in memory, how to create-manipulate them and to use them in the best possible manner as per the requirements of the application. Understanding the data structures used in computer system will help them how data gets organized in memory. This will help them in the development of their software related to memory management.

Data Structures
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Abstract Data Types, Linked Structures, Stacks,Queues	10
2	Doubly Linked list, Trees, Priority Queues & Heaps	10
3	Graph, Hashing	10
Total		**30**

Course Outcomes
The learner will be able to **CO1 :** Understand abstract data types and demonstrate proficiency in implementing linked structures, stack, and queue data structures for efficient data management. **CO2 :** Implement and manipulate doubly linked lists, trees, priority queues, and heaps for efficient data storage and retrieval. **CO3 :** Demonstrate proficiency in graph theory concepts and hashing techniques for efficient data representation and retrieval.

Sr. No.	Modules / Units
1	**Abstract data Type, Linked Structures, Stacks, Queues (10 Lectures)**
	Different Data Types, different types of data structures & their classifications, Introduction to ADT ADT for linked list, Advantages & Disadvantages, Singly Linked List-Traversing, Searching, Prepending and Removing Nodes Stack ADT for Stack, Advantages & Disadvantages, Applications of stack Queue ADT, Advantages & Disadvantages, linked representations. Circular Queue operations, Dequeues
2	**Doubly Linked list, Trees, Priority Queues & Heaps (10 Lectures)**
	ADT of doubly linked list, Advantages & Disadvantages, Insertion and deletion of nodes at various positions ADT for Tree Structure. Advantages & disadvantages, Binary Tree-Properties, Implementation and Traversals, Binary Search Tree, Balanced BST,Threaded Binary Trees, AVL Trees, Applications of Tree like Huffman Coding Priority Queue, Priority Queue ADT, Advantages and Disadvantages, Applications, Heaps, types of heaps, Heapifying the element
3	**Graph, Hashing (10 Lectures)**
	Introduction, Graph ADT, Advantages and Disadvantages, Graph Representation using adjacency matrix and adjacency list, Graph operations like insertion and deletion of nodes, Graph Traversals using BFS & DFS, Applications of Graphs like shortest path algorithms, Hash Table ADT, Advantages & Disadvantages, Concept of hashing, hash table, hash functions, collision, collision avoidance techniques, Applications of hashing

Learning Resources recommended

Textbooks:

1. Introduction to Algorithm, Thomas H Cormen, PHI
2. Data Structures And Algorithms Made Easy, NarasimhaKarumanchi, 2021.

Additional References:

1. Fundamentals of Computer Algorithms, SartajSahni and SanguthevarRajasekaran Ellis Horowitz, Universities Press, 2018
2. Data Structures and Algorithms in Python, Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, Wiley, 2016

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester II
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Data Structures Practical

Name of the Course	**Data Structures Practical**
Course Code	23_USCS206
Class	FYBSC
Semester	II
No of Credits	2
Nature	Practical
Type	Minor
Employability/ entrepreneurship/ skill development	Mathematics serves as a base for Computer Science. This paper helps in logic building and basic computational techniques.

Data Structures Practical
Modules at a Glance

Course Outcomes
The learner will be able to **CO1:** The course enables students to apply concepts like continuity, derivatives, extreme values, Newton's method, first-order differential equations, Euler's method, and partial derivatives to solve mathematical problems effectively. **CO2**: Understanding and applying various probability distributions, such as normal, binomial, Poisson, and exponential distributions, to model and analyze real-world phenomena, as well as interpreting their properties and characteristics. **CO3:** Implement and utilize data structures and algorithms, including binary trees, Huffman coding, graphs, and the traveling salesman problem, to efficiently solve computational problems in various domains.

Sr. No.	**Practical Title**
The following practical's will be implemented using python and Sagemath. Minimum 20 practical's to be completed as a journal work.	
1	Implement Continuity of functions and Derivative of functions with suitable example
2	Calculate Relative maxima, relative minima, absolute maxima, absolute minima with suitable example.
3	Implement Newton's method to find approximate solution of an equation
4	Implement Solution of a first order first degree differential equation, Euler's method
5	Calculate partial derivatives of given suitable functions
6	Implement any Problems based on normal distribution & Property Plotting
7	Calculate and Plot pdf, cdf, pmf, for discrete and continuous distribution

8	Implement t test, normal test, F test with suitable example
9	Write a program to implement Abstract Data Types (ADT)
10	Write a program to implement Singly Linked list with create and traversal operations
11	Write a program to implement Singly Linked list with insertion and deletion operations.
12	Write a program to implement Doubly Linked list with create and traversal operations
13	Write a program to implement Doubly Linked list with insertion and deletion operations.
14	Write a program to implement Stack with insertion, deletion, traversal operations.
15	Write a program to implement Queue with insertion, deletion, traversal operations.
16	Write a program to implement Priority Queue with insertion, deletion, traversal operations.
17	Write a program to implement Binary Tree with insertion and traversal operations.
18	Write a program to implement Binary Tree with deletion and traversal operations.
19	Write a program to implement Graph with insertion and traversal operations.
20	Write a program to implement Graph with deletion and traversal operations.
21	Write a program to implement Huffman Coding
22	Write a program to implement Travelling Salesman Problem

Learning Resources recommended

Textbooks:

1. Data Structure and algorithm Using Python, Rance D. Necaise, 2016 Wiley India Edition
2. Data Structure and Algorithm in Python, Michael T. Goodrich, Robertom Tamassia, M. H. Goldwasser, 2016 Wiley India Edition

Additional References:

1. Data Structure and Algorithmic Thinking with Python- Narasimha Karumanchi, 2015, Careermonk Publications
2. Fundamentals of Python: Data Structures, Kenneth Lambert, Delmar Cengage Learning

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester II
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Open Elective

Name of the Course	**Basic of Excel**
Course Code	24_USOE201
Class	FYBSC
Semester	II
No of Credits	2
Nature	Theory
Type	Open Elective
Employability/ entrepreneurship/ skill development	This paper focuses on advanced statistical techniques and its computation programmatically. It gives the best opportunity in the field of data science.

Basic of Excel
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Introduction to Excel, Understanding and Using Basic Functions	10
2	Proofing and Formatting, Printing Workbooks , Advance Paste Special Techniques, Time and Date Functions	10
3	Filtering and Sorting, Pivot Tables, Charts, Introduction to Excel	10
Total		**30**

Course Outcomes
The learner will be able to **CO1:** understand basic of MS Excel with simple function **CO2:** Organize data for analysis. **CO3:** Draw graphical representation of data.

Sr. No.	Aim of the Practical	Literature
The following practical's will be implemented using Microsoft Excel Minimum 20 practical's to be completed as a journal work.		
1	Implement the creation of workbooks and sheets in Microsoft excel	Introduction to Excel : A description of the interface, the menu system, and the fundamentals of spreadsheets
2	Introduction to MS Excel files.	Introduction to Excel: Workbooks, Worksheets, Columns and Rows.
3	Explain how we can maintain excel for personal use.	Personal use of Microsoft Excel like to making do list, daily routine etc.
4	Implement the use of AutoCorrect and Customizing the Ribbon	Changing Excel's Default Options Using AutoCorrect and Customizing It Customizing the Ribbon
5	Explain points to be considered for printing workbooks in Excel	Printing Workbooks in Microsoft Excel
6	Implement the printing of Workbooks in Microsoft excel	Setting Up Print Area. Customizing Headers & Footers. Designing the structure of a template. Print Titles –Repeat Rows / Columns
7	Create a worksheet and implement basic functions in Excel	Understanding and Using Basic Functions :Using Functions – Sum, Average, Max, Min, Count, Counta.
8	Illustrate the use of Spreadsheets.	Fundamentals of spreadsheets in Microsoft Excel.
9	Explain paste special techniques in Microsoft Excel	Advance Paste Special Techniques in Microsoft Excel
10	Explain filtering techniques in Excel	Filtering and Sorting using Microsoft Excel.
11	Create a worksheet and implement Text functions in Excel	Text Functions:Upper, Lower, Proper,Left, Mid, Right,Trim, Len, Exact, Concatenate, Find, Substitute.
12	Illustrate arithmetic functions in excel.	Understanding and Using Basic Functions in Microsoft Excel
13	Create a worksheet and implement arithmetic functions in Excel	Arithmetic Functions SumIf,,SumIfs CountIf, CountIfs AverageIf, AverageIfs
14	Implement advanced paste special techniques.	Advance Paste Special Techniques : Paste Formulas, Paste Formats, Transpose Tables, Paste Validations
15	Implement filtering and sorting techniques using Microsoft Excel	Filtering on Text, Numbers & Colors, Sorting Options, Advanced Filters
16	Elaborate time and date functions in Excel.	Use of Time and Date Functions in Microsoft Excel.
17	Create a worksheet and implement types of charts in Microsoft Excel	Charts : Various Charts i.e. Bar Charts / Pie Charts / Line Charts
18	Illustrate how we can secure any Excel File.	Maintaining security in Microsoft Excel.
19	Implement various formats of time and date in Microsoft Excel	Time and Date Functions: Today, Now, Date, Date if, DateAdd, Day, Month, Year, Month, Weekday
20	Elaborate the use of Pivot tables in Microsoft excel	Pivot Tables in Excel :Creating Simple Pivot Tables, Classic Pivot table

21	Create and implement pivot tables in Excel	Pivot Tables :Creating Simple Pivot Tables, Classic Pivot table
22	Implement how to protect workbook and worksheet in Microsoft Excel.	File Level Protection: Workbook, Worksheet Protection

Learning Resources recommended
Additional References: 1. https://www.w3schools.com/EXCEL/index.php 2. https://edu.gcfglobal.org/en/excel/# 3. https://www.javatpoint.com/excel-tutorial 4. https://www.excel-easy.com/ 5. https://support.microsoft.com/en-au/office/excel-video-training-9bc05390-e94c-46af-a5b3-d7c22f6990bb

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester II with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Open Elective

Name of the Course	**Basic Computers System**
Course Code	23_USOE203
Class	FYBSC
Semester	II
No of Credits	2
Nature	Practical
Type	Open Elective
Employability/ entrepreneurship/ skill development	It gives basic computer knowledge and techniques to the students which create a base for employability in various fields.

Basic Computers System
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Introduction to computers	10
2	Computer networking	10
3	The Internet and Internet Services	10
Total		**30**

Course Outcomes
The learner will be able to **CO1 :** Gain a comprehensive understanding of computer basics, including operating systems and word processing and spreadsheet concepts. **CO2 :** Understand computer networking principles and the ability to identify and mitigate the risks associated with computer viruses. **CO3 :** Acquire a comprehensive understanding of the Internet and its various services, enabling effective utilization of electronic mail.

Sr. No.	Modules / Units
1	**Introduction to computers (10 Lectures)**
	Introduction to computers: Overview and functions of a computer system, Input and output devices, Storage devices.Modern computers: The workstation, The Minicomputer, Mainframe Computers, Parallel processing Computer & The Super Computer **Introduction to operating systems:**Operating System concept, Windows, Unix/Linux & servers **Word Processing:**Basic Operations, Creating and Editing documents, Formatting documents. **Spreadsheet:**Creating and editing workbook, Organizing and formatting worksheets; Data analysis and management; Using formulas and functions.
2	**Computer networking (10 Lectures)**
	Introduction to networking: Various terminologies Associated hardware devices, gadgets (Router, Switch) tools, services, and resources Network Topologies and Protocols, LAN, WAN and MAN World Wide Web (WWW) Network security: fire walls **Computer viruses:** An overview of Computer viruses: What is a virus? Virus signs, how do they get transmitted? What are the dangers? General Precautions
3	**The Internet and Internet Services (10 Lectures)**
	The Internet and Internet Services: Introduction, History of Internet, Internetworking Protocol, The Internet Architecture, Managing the Internet, connecting to Internet, Internet Connections: Dial-up Access, Leased Line, Integrated Services Digital Network (ISDN), Digital Subscriber Line (DSL), Cable, Modem Internet Address **Internet Services:** World Wide Web (WWW), Web Browser, Uniform Resource Locator (URL), Internet Search Engines, WWW Development Languages, Uses of Internet **Electronic Mail:** E-mail Address, E-mail Message Format, E-mail Services, How E-mail Works File Transfer Protocol (FTP), How FTP Works, Terminal Network (Telnet), News, Internet Relay Chat (IRC), MS Outlook.

Learning Resources recommended
References: 1. Sinha, P. K., Sinha, P. (2004). Computer Fundamentals. India: BPB Publications. 2. Goel, A. (2010). Computer Fundamentals. India: Pearson Education. 3. Wempen, F. (2014). Computing Fundamentals: Introduction to Computers. Germany: Wiley. 4. Tanenbaum, A. S., Wetherall, D. (2014). Computer Networks. United Kingdom: Pearson Education.

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester II
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Skill Enhancement Course

Name of the Course	**Linux Practical**
Course Code	23_USCSS207
Class	FYBSC
Semester	II
No of Credits	2
Nature	Practical
Type	SEC
Employability/ entrepreneurship/ skill development	This syllabus will help to train students in fundamental skills and build-up sustainable interest in Linux Operating System. It will improve the necessary knowledge base to understand Linux Operating System and its practical implementation, it will also help to develop Linux based solutions for real life problems. The said course requires theory as well as practical sessions. Theoretical session will be covered during the practical session. Visual appearance of theory topic will be given during practical work. Commands and scripts are taken in the form of practical sessions.

Course Outcomes
The learner will be able to **CO1** : Gain proficiency in Linux fundamentals, including installation, desktop management, file system operations, and becoming power users, enabling efficient use of Ubuntu. **CO2** : Mastering Linux command-line operations, including file handling, zipping, advanced file manipulation, general-purpose utility commands, simple filters, I/O redirection, and networking commands. **CO3** : Demonstrate proficiency in shell scripting, editors, process management, function usage, and mathematical scripting for efficient automation and task execution in Linux environments.

Sr. No.	**Concept**	**Practical Title**
The following practical's will be implemented using VMware , Ubuntu. Minimum 20 practical's to be completed as a journal work.		
1	Introduction	Introduction to Linux and its distributions.
2	Installation	Booting and Installing from (USB/DVD)
3	Managing Desktop	Customize the desktop environment by changing different default options like changing default background, themes, and

		screensavers.
4	Operations performed on a Linux	Administering system and User setting-Screen Resolution, Time Settings,etc
5	Becoming Ubuntu power user	Learning Unity keyboard Using the Terminal
6	File System Commands	touch, help, man, more, less, pwd, cd, mkdir, rmdir, ls, find, ls
7	File handling Commands	cat, cp, rm, mv, more, file, wc, od, cmp, diff, comm,
8	Zipping Co mmands	gzip and gunzip, zip and unzip, tar
9	Advanced File handling Commands	chmod, chown, chgrp, , ln, umask,, chmod, chgrp, chown, etc
10	General purpose utility Comma nds	cal, date, echo, man, printf, passwd, script, who, uname, tty, stty
11	Simple Filters and I/O redirection	head, tail, cut paste, sort, grep family, tee, uniq, tr, etc.
12	Networking Commands	who, whoami, ping, telnet, ftp, ssh, etc
13	Shell scripting I	Conditions (for loop, until loop and while loop) arithmetic operations, examples
14	Shell scripting II	Redirecting Input / Output in scripts, creating your own Redirection
15	Editors	vi, sed, awk
16	Working and Managing with processes	sh, ps, kill, nice, at and batch etc.
17	Shell scripting III	Defining variables, reading user input, exit and exit status commands

18	Shell scripting IV	expr, test, [], if conditional, logical operators
19	Use of Functions	Create a function, pass value to a function and return a value from a function.
20	Maths scripting	Pipes performing maths

Learning Resources recommended

Textbooks:

1. "Linux Command line and Shell Scripting Bible", Richard Blum, Wiley India.
2. "Unix: Concepts and Applications", Sumitabha Das, 4th Edition, McGraw Hill.
3. "Official Ubuntu Book", Matthew Helmke& Elizabeth K. Joseph with Jose Antonio Rey and Philips Ballew, 8th Ed.

Additional References:

1. "Linux Administration: A Beginner's Guide", Fifth Edition, Wale Soyinka, Tata McGraw-Hill, 2008.
2. "Linux: Complete Reference", Richard Petersen, 6th Edition, Tata McGraw-Hill

"Beginning Linux Programming", Neil Mathew, 4th Edition, Wiley Publishing, 2008.

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester II
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
AEC

Name of the Course	**Content Writing**
Course Code	23_USCS208
Class	FYBSC
Semester	II
No of Credits	2
Nature	Theory
Type	AEC
Employability/ entrepreneurship/ skill development	With the advent of the internet, content writing has become a very lucrative and promising career. The course is designed to equip students to comprehend, refine, and enhance their writing abilities so that they may become proficient web content developers. The course aims to prepare students to enter the industry with enhanced skill and substantial competence. Tools for content writing, developing contents on web sites are curtain from unit I as these points will be covered in Unit II and Unit III. Also grammatical errors and legal English vocabulary are eliminated to cope up with 30 hours since the course originally designed for 45 hours.

Content Writing
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Basics of Content writing , Best Practices for Writing for the Web	10
2	Social Media Writing, Infographics	10
3	Content Tools, Ethical and Legal aspects of content writing	10
Total		**30**

Course Outcomes
The learner will be able to **CO1:** Understand concepts of content writing. **CO2:** Connect them with various writing and editing styles and techniques and develop their creative abilities. **CO3:** Proficiently utilize content tools while demonstrating a comprehensive understanding of ethical and legal considerations in content writing.

Sr. No.	Modules / Units
1	**Basics of Content writing , Best Practices for Writing for the Web (10 Lectures)**
	Basics of Content writing: Introduction to Content Writing, Learning Tone in Writing and Its Types, Comprehending style in writing and its Types. Best Practices for Writing for the Web: Making our story Elegant, Professional, Write with an Attitude, Keep Verbs Active, List Items, Title and Subtitle, Organize for Your Audience.
2	**Social Media Writing, Info graphics (10 Lectures)**
	Social Media Writing: Writing for Twitter, writing with Hashtags, Writing Social Media with Humor, writing for Facebook, writing for LinkedIn, Writing Your LinkedIn Profile, writing for Email, Writing Landing Pages, Writing Headlines, writing a Home Page, Writing the About Us Page, Writing Better Blog Posts, Writing Annual Reports. Infographics : Visual Communication-What Are Infographics?, The Science of Visualization, Creating Infographics-Purpose, The Art of Observation, Processing Your Ideas, Designing Your Infographics, Publishing Your Infographics.
3	**Content Tools, Ethical and Legal aspects of content writing (10 Lectures)**
	Content Tools.: Research and Knowledge Management Tools, Writing Tools, Productivity Tools, Editing Tools, Non-Text Writing Tools, Image Sources, Tools for Content Writing. Ethical and Legal aspects of content writing: Learn Legal English, Learn Legal Vocabulary In Legal Writing, IPR Laws, and Copywriting, Plagiarism laws in Content Writing.

Learning Resources recommended
Textbooks: 1. Content Writing Handbook, Author:Kounal Gupta, 2020, Henry Harvin. 2. Feldar, Lynda. Writing for the Web: Creating Compelling Web Content Using Words, Pictures, and Sound. New Riders, CA, USA, 2011 **Additional References:** 1. Everybody Writes: Your Go-To Guide to Creating Ridiculously Good Content Paperback Ann Handley Pan Macmillan India 2016 2. The Power of Infographics: Using Pictures to Communicate and Connect With Your Audiences Paperback – 15 June 2012 Mark Smiciklas 3. Law Relating to Intellectual Property Rights Book by V. K. Ahuja, 2017 **Web Resources:** 1. https://www.locationrebel.com/b2b-writing/ 2. https://www.mindler.com/blog/how-to-become-a-content-writer-in-india/

3. https://study.com/articles/What_is_a_Content_Writer.htmlhttps://www.mondaq.com/india/contracts-and-commercial-law/445620/legal-contractsagreements-drafting-and-legal-vetting
4. https://www.crazyegg.com/blog/copywriting/

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester II
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
VEC

Name of the Course	***Green Computing II***
Course Code	23_USCS209
Class	FYBSC
Semester	II
No of Credits	2
Nature	Theory
Type	VEC
Employability/ entrepreneurship/ skill development	This course is a continuation of the Semester I paper. Topics like the role of IT and managing green IT are introduced. These topics will make awareness about green IT among students and develop an era of Computer Systems which are less harmful to the environment.

Green Computing II
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	Green Networks and Communications, Enterprise Green IT Strategy	10
2	Sustainable Information Systems and Green Metrics, Enterprise Green IT Readiness	10
3	Sustainable IT Services, Green Enterprises and the Role of IT, Managing Green IT	10
Total		**30**

Course Outcomes
The learner will be able to **CO1:** Learning about green IT can be achieved in and by hardware, software, network communication and data center operations. **CO2:**Understand the strategies, frameworks, processes and management of green IT **CO3:**Demonstrate critical thinking and problem-solving skills in addressing green IT challenges and opportunities.

Sr. No.	Modules / Units
1	**Green Networks and Communications, Enterprise Green IT Strategy (10 Lectures)**
	Green Networks and Communications: Introduction, Objectives of Green Network Protocols, Green Network Protocols and Standards Enterprise Green IT Strategy: Introduction, Approaching Green IT Strategies, Business Drivers of Green IT Strategy, Business Dimensions for Green IT Transformation, Organizational Considerations in a Green IT Strategy, Steps in Developing a Green IT Strategy, Metrics and Measurements in Green Strategies.
2	**Sustainable Information Systems and Green Metrics, Enterprise Green IT Readiness (10 Lectures)**
	Sustainable Information Systems and Green Metrics: Introduction, Multilevel Sustainable Information, Sustainability Hierarchy Models, Product Level Information, Individual Level Information, Functional Level Information, Organizational Level Information, Measuring the Maturity of Sustainable ICT Enterprise Green IT Readiness: Introduction, Readiness and Capability, Development of the G-Readiness Framework, Measuring an Organization's G-Readiness
3	**Sustainable IT Services, Green Enterprises and the Role of IT, Managing Green IT (10 Lectures)**
	Sustainable IT Services: Creating a Framework for Service Innovation: Introduction, Factors Driving the Development of Sustainable IT, Sustainable IT Services (SITS), SITS Strategic Framework Green Enterprises and the Role of IT : Introduction, Organizational and Enterprise Greening, Information Systems in Greening Enterprises, Greening the Enterprise: IT Usage and Hardware, Inter-organizational Enterprise Activities and Green Issues Managing Green IT : Introduction, Strategizing Green Initiatives, Implementation of Green IT, Information Assurance, Communication and social media, case study

Learning Resources recommended

Text book:

1. Harnessing Green IT: Principles and Practices, San Murugesan, G. R. Ganadharan, Wiley & IEEE.

Additional References:

1. Green IT, Deepak Shikarpur, Vishwkarma Publications, 2014
2. Green Communications: Principles, Concepts and Practice- Samdanis et al, J. Wiley Green IT for Sustainable Business Practice: An ISEB Foundation Guide, Mark G. O'Neill, The Chartered Institute for IT, 2010

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester I
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Minor Course

Name of the Course	**Fundamentals of Database Systems**
Course Code	24_USCS210
Class	FYBSC
Semester	II
No of Credits	2
Nature	Theory
Type	Minor
Employability/ entrepreneurship/ skill development	Database is a core concept that every IT professional needs to know. All the basic concepts of database are included in this course. This course will make learners aware about the database concepts and they can learn advanced concepts. Due to the course schedule only necessary concepts are mentioned in this course.

Fundamentals of Database Systems
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	**Introduction to DBMS ,Data models. Entity Relationship Model**	10
2	**DDL Statements, DML Statements, Relational data model**	10
3	**Functions, Database Protection, Transaction control commands**	10
Total		**30**

Course Outcomes
The learner will be able to **CO1:** Aware of the fundamentals of the database system and idea how ERD components are helpful in database design and implementation. **CO2:** Familiarize the student with normalization, database protection and different DCL Statements. **CO3:** Aware about the importance of protecting data from unauthorized users and skills in optimizing functions and procedures to enhance database performance and scalability.

Sr. No.	Modules / Units
1	**Introduction to DBMS ,Data models. Entity Relationship Model (10 Lectures)**
	Database, DBMS – Definition, Overview of DBMS, Advantages of DBMS, Levels of abstraction, Data independence, DBMS Architecture Client/Server Architecture, Object Based Logical Model, Record Based Logical Model (relational, hierarchical, network) Entities, attributes, entity sets, relations, relationship sets, Additional constraints (key constraints, participation constraints, weak entities, aggregation / generalization, Conceptual Design using ER (entities VS attributes, Entity Vs relationship, binary Vs ternary, constraints beyond ER)
2	**DDL Statements, DML Statements, Relational data model (10 Lectures)**
	Creating Databases, Using Databases, datatypes, Creating Tables (with integrity constraints – primary key, default, check, not null), Altering Tables, Renaming Tables, Dropping Tables, Truncating Tables Viewing the structure of a table insert, update, delete, Select all columns, specific columns, unique records, conditional select, in clause, between clause, limit, aggregate functions (count, min, max, avg, sum), group by clause, having clause Domains, attributes, Tuples and Relations, Relational Model Notation, Characteristics of Relations, Relational Constraints - primary key, referential integrity, unique constraint, Null constraint, Check constraint
3	**Functions, Database Protection, Transaction control commands (10 Lectures)**
	String Functions (concat, instr, left, right, mid, length, lcase/lower, ucase/upper, replace, strcmp, trim, ltrim, rtrim), Math Functions (abs, ceil, floor, mod, pow, sqrt, round, truncate) Date Functions (adddate, datediff, day, month, year, hour, min, sec, now, reverse) Security Issues, Threats to Databases, Security Mechanisms, Role of DBA, Discretionary Access Control, Backing Up and Restoring databases Commit, Rollback, Savepoints

Learning Resources recommended

Textbooks:

1. "Fundamentals of Database System", ElmasriRamez, NavatheShamkant, Pearson Education, Seventh edition, 2017
2. "Database Management Systems", Raghu Ramakrishnan and Johannes Gehrke, 3rd Edition, 2014
3. "Murach's MySQL", Joel Murach, 3rd Edition, 3rd Edition, 2019

Additional References:

1. "Database System Concepts", Abraham Silberschatz,HenryF.Korth,S.Sudarshan, McGraw Hill, 2017
2. "MySQL: The Complete Reference", VikramVaswani , McGraw Hill,

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester II
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Minor Course

Name of the Course	**Logic Building with Python**
Course Code	24_USCS211
Class	FYBSC
Semester	II
No of Credits	2
Nature	Theory
Type	Minor
Employability/ entrepreneurship/ skill development	Python is serving as a major component in programming in the case of programming environments. Many industries use python as the base for the development. The base of the language will help developers to bloom with different ideas and grab advanced concepts of python. Having an ease to the development concepts of python, any learner from any stream can able to learn this language easily and can seek for job in IT industry.

Logic Building with Python
Modules at a Glance

Sr. No.	**Modules**	**No. of Lectures**
1	**Python Basics, Flow control**	10
2	**Functions**	10
3	**Lists, Dictionaries and Structuring Data**	10
Total		**30**

Course Outcomes
The learner will be able to **CO1** : Understand python basics **CO2** : Understand flow control **CO3** : Understand function and list

Sr. No.	Modules / Units
1	**Python Basics, Flow control (10 Lectures)**
	Entering Expressions into the Interactive Shell, The Integer, Floating-Point, and String Data Types, String Concatenation and Replication, Storing Values in Variables, Your First Program, Dissecting Your Program Boolean Values, Comparison Operators, Boolean Operators,Mixing Boolean and Comparison Operators, Elements of Flow Control, Program Execution, Flow Control Statements, Importing Modules,Ending a Program Early with sys.exit()
2	**Functions (10 Lectures)**
	def Statements with Parameters, Return Values and return Statements,The None Value, Keyword Arguments and print(), Local and Global Scope, The global Statement, Exception Handling, A Short Program: Guess the Number
3	**Lists, Dictionaries and Structuring Data (10 Lectures)**
	The List Data Type, Working with Lists, Augmented Assignment Operators, Methods, Example Program: Magic 8 Ball with a List, List-like Types: Strings and Tuple The Dictionary Data Type, Pretty Printing, Using Data Structures to Model Real-World Things,

Learning Resources recommended

Textbooks:

1. Introduction to Algorithm, Thomas H Cormen, PHI
2. Data Structures And Algorithms Made Easy, NarasimhaKarumanchi, 2021.

Additional References:

1. Fundamentals of Computer Algorithms, SartajSahni and SanguthevarRajasekaran Ellis Horowitz, Universities Press, 2018
2. Data Structures and Algorithms in Python, Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, Wiley, 2016

Revised Syllabus of Courses of B.Sc. Computer Science
Programme at Semester II
with Effect from the Academic Year 2023-2024
Department Specific Courses (DSC)
Minor Course Practical

Name of the Course	**Database and python practicals**
Course Code	24_USCS212
Class	FYBSC
Semester	II
No of Credits	2
Nature	Practical
Type	Minor
Employability/ entrepreneurship/ skill development	Giving hands on theory topics helps to understand theoretical concepts very easily.

Database and python practicals
Modules at a Glance

Course Outcomes
Course Outcomes: The learner will be able to **CO1** : Implement basic Python programs to solve simple computational problems. **CO2** : Implement python programs by using different types of loops, list and dictionaries. **CO2** : Develop proficiency in creating, querying, and modifying databases using basic SQL commands, understand database management principles. **CO4** : Use different Mysql functions along with transactional language.

Database and python practicals

Sr. No.	**Practical Title**
The following practical's will be implemented using Python and MySQL. Minimum 20 practical's to be completed as a journal work.	
1	Working with input and output statements in python.
2	Develop a program to read the student details like Name, USN, and Marks in three subjects. Display the student details, total marks and percentage with suitable messages.
3	Develop a program to read the name and year of birth of a person. Display whether the person is a senior citizen or not.
4	Develop a program to generate Fibonacci sequences of length (N). Read N from the console.
5	Write a function to calculate the factorial of a number. Develop a program to compute binomial coefficient (Given N and R)

6	Read N numbers from the console and create a list. Develop a program to print mean, variance and standard deviation with suitable messages.
7	Read a multi-digit number (as chars) from the console. Develop a program to print the frequency of each digit with suitable message
8	Develop a program to print 10 most frequently appearing words in a text file. [Hint: Use a dictionary with distinct words and their frequency of occurrences. Sort the dictionary in the reverse order of frequency and display dictionary slice of first 10 item
9	Develop a program to sort the contents of a text file and write the sorted contents into a separate text file
10	Write a Python program to implement a dictionary in Python for suitable problems. Demonstrate various operations on it.
11	Write a suitable Python program to implement recursion for problems such as Fibonacci series, Factorial etc.
12	Write a suitable Python Programs to study various loops (while loop, for loop, Nested loop)
13	Write a suitable Python Programs to study various loops (switch statement, break statement, continue statement)
14	Conceptual Designing using ER Diagrams (Identifying entities, attributes, keys and relationships between entities, cardinalities, generalization, specialization etc.)
15	Perform the following: 1.Viewing all databases 2.Creating a Database 3.Viewing all Tables in a Database 4. Dropping Databases
16	Perform the following: 1.Creating Tables without constraints 2. Creating Tables with constraints
17	Inserting/Updating/Deleting Records in a Table.
18	Perform the following: Altering a Table Dropping/Truncating/Renaming Tables
19	Queries involving Date functions.
20	Queries involving String functions.
21	Queries involving Math functions.
22	DCL statements like Commit, Rollback and savepoints.

Learning Resources recommended
Textbooks: 1. Practical Programming: An Introduction to Computer Science Using Python 3, Paul Gries , Jennifer Campbell, Jason Montojo, Pragmatic Bookshelf, 2nd Edition, 2014 2. Programming through Python, M. T Savaliya, R. K. Maurya& G M Magar, Sybgen Learning India, 2020 3. MySQL: The Complete Reference", VikramVaswani , McGraw Hill, **Additional References**: 1. Python: The Complete Reference, Martin C. Brown, McGraw Hill, 2018 2. Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress, 2017 3. Programming in Python 3, Mark Summerfield, Pearson Education, 2nd Ed, 2018

Evaluation Scheme

For 2 Credit Theory Course:

The allocation of marks for the Internal Assessment and Semester End Examinations are as follows

Evaluation for the course will be of 50 marks conducted in a 30:20 pattern. 30 marks will be for semester end examination and 20 marks will be for internal evaluation.

Internal Evaluation: 20 Marks (40%)

Method	Marks
Mid-Term Class Test ● It should be conducted using any **learning management system** such as **Moodle** (Modular object-oriented dynamic learning environment) ● The test should have **20 MCQ's** which should be solved in a time duration of **40 minutes.**	10
Assignment/ Case study/ Presentations ● Assignment / Case Study Report / Presentation can be uploaded on any **learning management system**.	05
Attendance and behaviour	05

Semester End Evaluation: 30 Marks (60%)

Duration: 1 Hour

	All questions are compulsory.			
Question	**Based on**	**Sub-Question**	**Options**	**Marks**
Q.1	Unit I	A	*Any 2 out of 4*	6
		B	*Any 1 out of 2*	4
Q.2	Unit II	A	*Any 2 out of 4*	6
		B	*Any 1 out of 2*	4
Q.3	Unit III	A	*Any 2 out of 4*	6
		B	*Any 1 out of 2*	4

Evaluation Scheme

For Practical Course:

The allocation of marks for the Internal Assessment and Semester End Examinations are as follows

Evaluation for the course will be of 50 marks conducted in a 60: 40 pattern. 30 marks will be for semester end examination and 20 marks will be for internal evaluation.

Evaluation Pattern

Internal Examination: 20 Marks (40%)

Sr No	Criterion	Marks
1	Journals containing minimum 10 practical's which are timely completed with desired output	10
2	Attendance & Practical Performance	10

Semester End Examination: 30 Marks (60%)

Sr No	Criterion	Marks
1	One Practical Question **OR** Combination of Practical Questions **OR** Combination of Practical Question and Theory Question **Duration : 2 Hours**	25
2	Viva	05

Standard of Passing

The learner to pass a course shall have to obtain a minimum of 40% marks in aggregate for each course where the course consists of Internal Assessment & Semester End Examination. The learnershall obtain minimum of 40% marks (i.e. 16 out of 40) in the Internal Assessmentand 40% marks inSemester End Examination (i.e. 24 out of 60) separately, to pass the courseand minimum of Letter Grade "P" in the project component, wherever applicable to pass a particular semester. A learner will be said to have passed the course if the learner passes the Internal Assessment & Semester EndExamination together. **Performance Grading:**

Letter Grades and Grade Points

Semester GPA/ Program CGPA Semester/Program	**% of Marks**	**Alpha-Sign / Letter Grade Result**
9.00-10.00	90.0 -100	0 (Outstanding)
8.00 ≤ 9.00	80.0 ≤ 90.0	A+ (Excellent)
7.00 ≤ 8.00	70.0 ≤ 80.0	A (Very Good)
6.00 ≤ 7.00	60.0 ≤ 70.0	B+ (Good)
5.50 ≤ 6.00	55.0 ≤ 60.0	B (Above Average)
5.00 ≤ 5.50	50.0 ≤ 55.0	C (Average)
4.00 ≤ 5.00	40.0 ≤ 50.0	P (Pass)
Below 4.00	Below 40	F (Fail)
Ab (Absent)	-	Absent

Date: The Chairperson

Place: Ratnagiri. BoS of Computer Science