RANI CHANNAMMA UNIVERSITY, BELAGAVI

PROGRAM /COURSE STRUCTURE AND SYLLABUS
as per the Choice Based Credit System (CBCS)
designed in accordance with
Learning Outcomes-Based Curriculum Framework (LOCF)
of National Education Policy (NEP) 2020
for

Bachelor of Science
(Computer Science)

w.e.f.

Academic Year 2021-22 and onwards

Board of Studies (UG) Committee

Bachelor of Science (Computer Science)Programme 2021-22

1	**Prof. Dayanand G Savakar, RCU, Belagavi**	**Chairman**	
2	**Smt. Asmitha S Deshpande, ,Gogte Commerce College, Belagavi.**	**Member**	
3	**Shri. AnandPatil, DMSM Society, BCA College, Belagavi**	**Co-Opt Member**	
4	**Shri. Prasad Khode, SKE Society, GSS, BCA College, Belagavi**	**Co-Opt Member**	

Dr. Vijayalaxmi S Shigehalli
Dean of Science Faculty
Rani Channamma University, Belagavi

Dr. Dayanad G Savakar
Chairman BoS(UG)
Department of Computer Science
RCU, Belagavi

PREAMBLE

Computer Science (CS) has been evolving as an important branch of science and technology in last two decade and it has carved out a space for itself like engineering. Computer Science spans theory and more application and it requires thinking both in abstract terms and in concrete terms.

The ever -evolving discipline of computer science has strong connections to other disciplines. Many problems in science, engineering, health care, business, and other areas can be solved effectively with computers, but finding a solution requires both computer science expertise and knowledge of the particular domain. Computer science has a wide range of specialties. These include Computer Architecture, Software Systems, Graphics, Artificial Intelligence, Mathematical and Statistical Analysis, Data Science, Computational Science, and Software Engineering.

Universities and other HEIs introduced programmes of computer science. Information Technology is growing rapidly. Increasing applications of computers in almost all areas of human endeavour has led to vibrant industries with concurrent rapid change in technology. Unlike other basic disciplines, developing core competency in this discipline that can be reasonably stable becomes a challenge.InIndia, it was initially introduced at the Master (postgraduate) level as MCA and M.Tech. Later on, engineering programmes such as B.Tech and B.E in Computer Science & Engineering and in Information Technology were introduced in various engineering College/Institutions to cater to the growing demand for trained engineering manpower in IT industries.Parallelly, BCA, BSc and MSc programmes with specialisation in Computer Science were introduced to train manpower in this highly demanding area.

BSc and BSc (Hons) are aimed at undergraduate level training facilitating multiple career paths. Students so graduated, can take up postgraduate programmes in CS(M.Sc)or MCA leading to research as well as R&D, can be employable at IT industries, or can pursue a teaching profession or can adopt a business management career. BSc and BSc (Hons) aims at laying a strong foundation of computer science at

an early stage of the career. There are several employment opportunities and after successful completion of BSc, graduating students can fetch employment directly in companies as programmer, Web Developer, Software Engineer, Network Administrator, Data Scientist, or AI/ML personnel.

The Program outcomes in BSc are aimed at allowing flexibility and innovation in design and development of course content, in method of imparting training, in teaching learning process and in assessment procedures of the learning outcomes. The emphasis in BSc courses, in outcome-based curriculum framework, help students learn solving problems, accomplishing IT tasks, and expressing creativity, both individually and collaboratively. The proposed framework will help Students learn programming techniques and the syntax of one or more programming languages.

All students must, therefore, have access to a computer with a modern programming language installed. The computer science framework does not prescribe a specific language. The teacher and students will decide which modern programming languages students will learn. More importantly, students will learn to adapt to changes in programming languages and learn new languages as they are developed.

The present Curriculum Framework for BSc degrees is intended to facilitate the students to achieve the following.

- To develop an understanding and knowledge of the basic theory of Computer Science and Information Technology with good foundation on theory, systems and applications such as algorithms, data structures, data handling, data communication and computation
- To develop the ability to use this knowledge to analyse new situations in the application domain
- To acquire necessary and state-of-the-art skills to take up industry challenges. The objectives and outcomes are carefully designed to suit to the above-mentioned purpose.
- The ability to synthesize the acquired knowledge, understanding and experience for a better and improved comprehension of the real-life problems

- To learn skills and tools like mathematics, statistics and electronics to find the solution, interpret the results and make predictions for the future developments
- To formulate, to model, to design solutions, procedure and to use software tools to solve real world problems and evaluate

PROGRAM OUTCOMES:

1. **Discipline knowledge:** Acquiring knowledge on basics of Computer Science and ability to apply to design principles in the development of solutions for problems of varying complexity
2. **Problem Solving**: Improved reasoning with strong mathematical ability to Identify, formulate and analyze problems related to computer science and exhibiting a sound knowledge on data structures and algorithms.
3. **Programming a computer**: Exhibiting strong skills required to program a computer for various issues and problems of day-to-day applications with thorough knowledge on programming languages of various levels.
4. **Application Systems Knowledge:** Possessing a sound knowledge on computer application software and ability to design and develop app for applicative problems.
5. **Communication:** Must have a reasonably good communication knowledge both in oral and writing.
6. **Ethics on Profession, Environment and Society:** Exhibiting professional ethics to maintain the integrality in a working environment and also have concern on societal impacts due to computer-based solutions for problems.
7. **Lifelong Learning:** Should become an independent learner. So, learn to learn ability.
8. **Motivation to take up Higher Studies:** Inspiration to continue educations towards advanced studies on Computer Science.

By the end of the program the students will be able to:

The Bachelor of Computer Science (BSc (Hons)) program enables students to attain following additional attributes besides the afore-mentioned attributes, by the time of graduation:

1. Apply standard Software Engineering practices and strategies in real -time software project development
2. Acquaint with the contemporary trends in industrial/research settings and thereby innovate novel solutions to existing problems
3. The ability to apply the knowledge and understanding noted above to the analysis of a given information handling problem.
4. The ability to work independently on a substantial software project and as an effective team member.

RANI CHANNAMMA UNIVERSITY

VIdyaSangam, NH-4, Belagavi. -591156

Curricular and Credits Structure under Choice Based Credit System [CBCS] of Computer Science Major& One Minor Discipline Scheme for the Four Years Computer Science B.Sc. Undergraduate Honors Programmewith effect from 2021-22

SEMESTER-I											
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
L1	21BSC1L1LK1	Kannada	40	60	100	4	-	-	3	2
	21BSC1L1LFK1	Functional Kannada								
L2	21BSC1L2LEN2	English	40	60	100	4	-	-	3	2
	21BSC1L2LHI2	Hindi								
	21BSC1L2LSN2	Sanskrit								
	21BSC1L2LTE2	Telugu								
	21BSC1L2LUR2	Urdu								
DSC1	21BSC1C1CS1L	Computer Fundamentals and Programming in C	40	60	100	4	-	-	4	2
	21BSC1C1CS1P	C Programming Lab	25	25	50	-	-	4	2	3
DSC1	Another Department Code	Another Department Course Title	40	60	100	4	-	-	4	2
			25	25	50	-	-	4	2	3
SEC1	21BSC1SE1CS1	Digital Fluency	25	25	50	1	-	2	2	2
VBC1	21BSC1V1PE1	Physical Education- Yoga	25	-	25	-	-	2	1	-
VBC2	21BSC1V2HW1	Health & Wellness	25	-	25	-	-	2	1	-
OEC1	21BSC1O1CS1	C Programming Concepts	40	60	100	3	-	-	3	2
			Total Marks		**700**	**Semester Credits**			**25**	

SEMESTER-II										
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
L3	21BSC2L3LK2	Kannada	40	60	100	4	-	-	3	2
	21BSC2L3FKL2	Functional Kannada								
L4	21BSC2L4EN2	English	40	60	100	4	-	-	3	2
	21BSC2L4HI2	Hindi								
	21BSC2L4SN2	Sanskrit								
	21BSC2L4TE2	Telugu								
	21BSC2L4UR2	Urdu								
DSC2	21BSC2C2CS2L	Data Structures using C	40	60	100	4	-	-	4	2
	21BSC2C2CS2P	Data structures Lab	25	25	50	-	-	4	2	3
DSC2	Another Department Code	Another Department Course Title	40	60	100	4	-	-	4	2
			25	25	50	-	-	4	2	3
AECC1	21BSC2AE1ES2	Environmental Studies	20	30	50	3	-	-	2	2
VBC3	21BSC2V3PE2	Physical Education- Sports	25	-	25	-	-	2	1	-
VBC4	21BSC2V4NC1	NCC/NSS/R&R(S&G) / Cultural	25	-	25	-	-	2	1	-
OEC2	21BSC2O2CS2	Web Designing	40	60	100	3	-	-	3	2
Total Marks					**700**	**Semester Credits**			**25**	
Exit option with Certificate (with the completion of courses equal to a minimum of 48 credits)					**1400**				**50**	

SEMESTER-III										
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
L5	21BSC3L5LK3	Kannada	40	60	100	4	-	-	3	2
	21BSC3L5LFK3	Functional Kannada								
L6	21BSC3L6EN3	English	40	60	100	4	-	-	3	2
	21BSC3L6HI3	Hindi								
	21BSC3L6SN3	Sanskrit								
	21BSC3L6TE3	Telugu								
	21BSC3L6UR3	Urdu								
DSC3	21BSC3C3CS1L	Object Oriented Programming in JAVA	40	60	100	4	-	-	4	2
	21BSC3C3CS1P	JAVA Lab	25	25	50	-	-	4	2	3
DSC3	Another Department Code	Another Department Course Title	40	60	100	4	-	-	4	2
			25	25	50	-	-	4	2	3
SEC2	21BSC3SE2ES2	Artificial Intelligence	25	25	50	1	-	2	2	2
VBC5	21BSC3V5PE3	Physical Education- Sports	25	-	25	-	-	2	1	-
VBC6	21BSC3V6NC2	NCC/NSS/R&R(S&G) / Cultural	25	-	25	-	-	2	1	-
OEC3	21BSC3O3CS5	R Programming	40	60	100	3	-	-	3	2
			Total Marks		**700**	**Semester Credits**			**25**	

SEMESTER-IV										
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
L7	21BSC4L7LK4	Kannada	40	60	100	4	-	-	3	2
	21BSC4L7LFK4	Functional Kannada								
L8	21BSC4L8EN4	English	40	60	100	4	-	-	3	2
	21BSC4L8HI4	Hindi								
	21BSC4L8SN4	Sanskrit								
	21BSC4L8TE4	Telugu								
	21BSC4L8UR4	Urdu								
DSC4	21BSC4C2CS2L	Database Management Systems	40	60	100	4	-	-	4	2
	21BSC4C2CS2P	DBMS Lab	25	25	50	-	-	4	2	3
DSC4	Another Department Code	Another Department Course Title	40	60	100	4	-	-	4	2
			25	25	50	-	-	4	2	3
AECC2	21BSC4AE1CI2	Constitution of India	20	30	50	3	-	-	2	2
VBC7	21BSC4V5PE4	Physical Education- Sports	25	-	25	-	-	2	1	-
VBC8	21BSC4V6NC3	NCC/NSS/R&R(S&G) / Cultural	25	-	25	-	-	2	1	-
OEC4	21BSC4O3CS7	Multimedia & Animation	40	60	100	3	-	-	3	2
Total Marks					**700**	**Semester Credits**			**25**	
Exit option with Diploma in Science (with the completion of courses equal to a minimum of 96 credits)OR continue studies with Major and Minor					**2800**				**100**	

SEMESTER-V										
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
Computer Science as Major Discipline										
DSC5	21BSC5C5CSMJ1L	Programming in PYTHON	40	60	100	3	-	-	3	2
	21BSC5C5CSMJ1P	PYTHON Programming lab	25	25	50	-	-	4	2	3
DSC6	21BSC5C5CSMJ2L	Computer Networks	40	60	100	3	-	-	3	2
	21BSC5C5CSMJ2P	Computer Networks Lab	25	25	50	-	-	4	2	3
DSC5	Another Department Code as a Minor Subject	Another Department Course Title	40	60	100	3	-	-	3	2
			25	25	50	-	-	4	2	3
VC1	21BSC5VC1US	Unix & Shell Programming	40	60	100	3	-	-	3	2
	21BSC5VC1FD	Fundamentals of Data Science								
VBC9	21BSC5V5PE5	Physical Education- Sports	25	-	25	-	-	2	1	-
VBC10	21BSC5V6NC4	NCC/NSS/R&R(S&G) / Cultural	25	-	25	-	-	2	1	-
SEC3	21BSC5SE3CS3	Cyber Security	25	25	50	1	-	2	2	2
		Total Marks			**650**	**Semester Credits**			**22**	

SEMESTER-VI										
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
Computer Science as Major Discipline										
DSC7	21BSC6C6CSMJ1L	Internet Technologies	40	60	100	3	-	-	3	2
	21BSC6C6CSMJ1P	Internet Technology Lab	25	25	50	-	-	3	2	3
DSC8	21BSC6C6CSMJ2L	Operating System Concepts	40	60	100	3	-	-	3	2
	21BSC6C6CSMJ2P	C# Programming Lab	25	25	50	-	-	3	2	3
DSC6	Another Department Code as a Minor Subject	Another Department Course Title	40	60	100	3	-	-	3	2
			25	25	50	-	-	3	2	3
VC2	21BSC6VC2HT	Health Care Technologies	40	60	100	3	-	-	3	2
	21BSC6VC2DM	Digital Marketing								
INT1	21BSC6 INT1L	Internship*	25	50	75	-	-	-	2	2
VBC1	21BSC6V5PE5	Physical Education- Sports	25	-	25	-	-	2	1	-
VBC2	21BSC6V6NC4	NCC/NSS/R&R(S&G) / Cultural	25	-	25	-	-	2	1	-
SEC4	21BSC6SE4CS4	Professional Communication	25	25	50	1	-	2	2	2
Total Marks					**700**	**Semester Credits**			**24**	
Exit with Bachelor of Science Degree, B. Sc. (with the completion of courses equal to a minimum of 140 credits)or continue studies with the Major					**4175**	**Total Credits for BSC Program**			**146**	

*Internship between 5 th& 6th Semester with 3 to 4 weeks

Computer Science Subject as a Minor Discipline

SEMESTER-V									
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
DSC5 As a Minor Subject	21BSC5C5CSMN1L	Programming in PYTHON	40	60	100	3	-	-	3	2
	21BSC5C5CSMN1P	PYTHON Programming lab	25	25	50	-	-	3	2	3

SEMESTER-VI									
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
DSC6 As a Minor Subject	21BSC6C6CSMN1L	Internet Technologies	40	60	100	3	-	-	3	2
	21BSC6C6CSMN1P	Internet Technology Lab	25	25	50	-	-	3	2	3

Concept Note, Abbreviation Explanation and Coding:

Concept Note:

1. **CBCS** is a mode of learning in higher education which facilitates a student to have some freedom in selecting his/her own choices, across various disciplines for completing a UG/PG program.
2. A credit is a unit of study of a fixed duration. For the purpose of computation of workload as per UGC norms the following is mechanism be adopted in the University:
   One credit (01) = One Theory Lecture (L) period of one (1) hour.
   One credit (01) = One Tutorial (T) period of one (1) hour.
   One credit (01) = One practical (P) period of two (2) hours.
3. Course: paper/subject associated with AECC, DSC, DSEC, SEC, VBC, OEC, VC, IC and MIL
4. In case of **B.Sc**. **Once a candidate chose two courses/subjects of a particular two department in the beginning, he/she shall continue the same till the end of the degree, then there is no provision to change the course(s) and Department(s).**
5. A candidate shall choose **one of the Department's courses as major and other Department course as minor in fifth and sixth semester and major course will get continued in higher semester.**
6. Wherever there is a practical there will be no tutorial and vice-versa
7. A major subject is the subject that's the main focus of Core degree/concerned.
8. A minor is a secondary choice of subject that complements core major/ concerned.
9. Vocational course is a course that enables individual to acquire skills set that are required for a particular job.
10. Internship is a designated activity that carries some credits involving more than **25 days** of working in an organization (either in same organization or outside) under the guidance of an identified mentor. Internship shall be an integral part of the curriculum.
11. **OEC: For non- computer science students. Computer Science students have to opt for OEC from departments other than major and minor disciplines.**

Abbreviation Explanations:

1. AECC: Ability Enhancement Compulsory Course.
2. DSC: Discipline Specific Core Course.
3. DSEC: Discipline Specific Elective Course.
4. SEC: Skill Enhancement Course.
5. VBC: Value Based Course.
6. OEC: Open/Generic Elective Course
7. VC: Vocational Course.
8. IC: Internship Course
9. L1: Language One
10. L2: MIL
11. L= Lecture; T= Tutorial; P=Practical.
12. MIL= Modern Indian Language; English or Hindi or Telugu or Sanskrit or Urdu

Program Coding:

1. Code 21: Year of Implementation
2. Code BSC: BSC Program under the faculty of Applied Science of the University
3. Code 1: First Semester of the Program, (2 to 6 represent higher semesters)
4. Code AE: AECC, (C for DSC, S for SEC, V for VBC and O for OEC)
5. Code 1: First "AECC" Course in semester, similarly in remaining semester for such other courses
6. Code LK: Language Kannada, similarly Language English, Language Hindi, Language Telugu, Language Sanskrit, &Language Urdu
7. Code 1: Course in that semester.
8. CS: Computer Science

ASSESSMENT METHODS

Evaluation Scheme for Internal Assessment:

Theory:

Assessment Criteria	40 marks
1st Internal Assessment Test for30 marks 1 hr after 8 weeks and 2nd Internal Assessment Test for 30 marks 1 hr after 15 weeks . Average of two tests should be considered.	30
Assignment	10
Total	**40**

Assessment Criteria	25 marks
1st Internal Assessment Test for20 marks 1hr after 8 weeks and 2nd Internal Assessment Test for 20 marks 1hr after 15 weeks. Average of two tests should be considered.	20
Assignment	05
Total	**25**

Practical:

Assessment Criteria	25 marks
Semester End Internal Assessment Test for 20 marks 2 hrs	20
Journal (Practical Record)	05
Total	**25**

Question Paper Pattern:

RANI CHANNAMMA UNIVERSITY
Department of Computer Science

BSc(Computer Science)

Sub: Code: Maximum Marks: 60

a. Answer any Six Questions from Question 1 b. Answer any Three each Questions from Question 2,3,4 and 5

Q.No.1.	Answer any Six Questions (Atlest Two question from Each Unit) a. b. c. d, e. f. g. h.	2X6=12
Q.No.2.	(Should cover Entire Unit-I) a. b. c. d.	4X3=12
Q.No.3.	(Should cover Entire Unit-II) a. b. c. d.	4X3=12
Q.No.4.	(Should cover Entire Unit-III) a. b. c. d.	4X3=12
Q.No.5.	(Should cover Entire Unit-IV) a. b. c. d.	4X3=12

COURSE-WISE SYLLABUS

Semester I

Year	I	**Course Code:**21BSC1C1CS1L	**Credits**	04
Sem.	1	**Course Title:** Computer Fundamentals and Programming in C	**Hours**	52

Course Pre-requisites, if any	NA	
Formative Assessment Marks: 40	Summative Assessment Marks: 60	Duration of ESA:.02 hrs.

Course Outcomes	After completing this course satisfactorily, a student will be able to: • Confidently operate Desktop Computers to carry out computational tasks • Understand working of Hardware and Software and the importance of operating systems • Understand programming languages, number systems, peripheral devices, networking, multimedia and internet concepts • Read, understand and trace the execution of programs written in C language • Write the C code for a given problem • Perform input and output operations using programs in C • Write programs that perform operations on arrays	
Unit No.	**Course Content**	**Hours**
Unit I	**Fundamentals of Computers**: Introduction to Computers - Computer Definition, Characteristics of Computers, Evolution and History of Computers, Types of Computers, Basic Organisation of a Digital Computer; Number Systems – different types, conversion from one number system to another; Computer Codes – BCD, Gray Code, ASCII and Unicode; Boolean Algebra – Boolean Operators with Truth Tables; Types of Software – System Software and Utility Software; Computer Languages - Machine Level, Assembly Level & High Level Languages, Translator Programs – Assembler, Interpreter and Compiler; Planning a Computer Program - Algorithm, Flowchart andPseudo code with Examples. **Introduction to C Programming**: Over View of C; History and Features of C; Structure of a C Program with Examples; Creating and Executing a C Program; Compilation process in C.	13
Unit II	**C Programming Basic Concepts**: C Character Set; C tokens - keywords, identifiers, constants, and variables; Data types; Declaration & initialization of variables; Symbolic constants. **Input and output with C**: Formatted I/O functions - printf and scanf, control stings and escapesequences, output specifications with printf functions; Unformatted I/O functions to read and display single character and a string - getchar, putchar, gets and	13

	puts functions. **C Operators & Expressions**: Arithmetic operators; Relational operators; Logical operators; Assignment operators; Increment & Decrement operators; Bitwise operators; Conditional operator; Special operators; Operator Precedence and Associatively; Evaluation of arithmeticexpressions; Type conversion.	
Unit III	**Control Structures**: Decision making Statements - Simple if, if_else, nested if_else, else_ifladder,Switch-case, goto, break & continue statements; Looping Statements - Entry controlled and Exit controlled statements, while, do-while, for loops, Nested loops. **Arrays**: One Dimensional arrays - Declaration, Initialization and Memory representation; Two Dimensional arrays - Declaration, Initialization and Memory representation. **Strings**: Declaring & Initializing string variables; String handling functions - *strlen, strcmp, strcpy and strcat;* Character handling functions - *toascii, toupper, tolower, isalpha, isnumeric*etc.	13
Unit IV	**Pointers in C:** Understanding pointers - Declaring and initializing pointers, accessing address and value of variables using pointers; Pointers and Arrays; Pointer Arithmetic; Advantages and disadvantages of using pointers; **User Defined Functions:** Need for user defined functions; Format of C user defined functions; Components of user defined functions - return type, name, parameter list, function body, return statement and function call; Categories of user defined functions - With and without parameters and return type. **User defined data types:** Structures - Structure Definition, Advantages of Structure, declaring structure variables, accessing structure members, Structure members initialization, comparing structure variables, Array of Structures; Unions - Union definition; difference between Structures and Unions.	13
Recommended Leaning Resources		
Print Resources	**Text Books** 1. Pradeep K. Sinha and PritiSinha: Computer Fundamentals (Sixth Edition), BPBPublication 2. E. Balgurusamy: Programming in ANSI C(TMH) **References** 1. Kamthane: Programming with ANSI and TURBO C (PearsonEducation) 2. V. Rajaraman: Programming in C (PHI –EEE) 3. S. ByronGottfried: Programming with C(TMH) 4. Kernighan &Ritche: The C Programming Language(PHI) 5. YashwantKanitkar: Let usC	

Year	I	**Course Code:** 21BSC1C1CS1P	**Credits**	02
Sem.	I	**Course Title:**C Programming Lab	**Hours**	45
Course Pre-requisites, if any:		Knowledge of Programming		
Formative Assessment Marks: 25		Summative Assessment Marks: 25	Duration of ESA: 03 hrs.	

Practice Labs

1. The following activities be carried out/ discussed in the lab during the initial period of the semester.
    1. Basic Computer Proficiency
        a. Familiarization of Computer Hardware Parts
        b. Basic Computer Operations and Maintenance.
        c. Do's and Don'ts, Safety Guidelines in Computer Lab
    2. Familiarization of Basic Software – Operating System, Word Processors, Internet Browsers, Integrated Development Environment (IDE) with Examples.
    3. Type Program Code, Debug and Compile basic programs covering C Programming fundamentals discussed during theory classes.

Part A:

1. Write a C Program to read radius of a circle and to find area and circumference
2. Write a C Program to read three numbers and find the biggest of three
3. Write a C Program to demonstrate library functions in *math.h*
4. Write a C Program to check for prime
5. Write a C Program to generate n primes
6. Write a C Program to read a number, find the sum of the digits, reverse the number and check it for palindrome
7. Write a C Program to read numbers from keyboard continuously till the user presses 999 and to find the sum of only positive numbers
8. Write a C Program to read percentage of marks and to display appropriate message (Demonstration of else-ifladder)
9. Write a C Program to find the roots of quadratic equation (demonstration of switch-case statement)
10. Write a C program to read marks scored by n students and find the average of marks (Demonstration of single dimensional array
11. Write a C Program to remove Duplicate Element in a single dimensionalArray
12. Program to perform addition and subtraction ofMatrices

PART B:

1. Write a C Program to find the length of a string without using built in function
2. Write a C Program to demonstrate string functions.
3. Write a C Program to demonstrate pointers in C
4. Write a C Program to check a number for prime by defining *isprime( )*function
5. Write a C Program to read, display and to find the trace of a square matrix
6. Write a C Program to read, display and add two m x n matrices using functions

	7. Write a C Program to read, display and multiply two m x n matrices using functions 8. Write a C Program to read a string and to find the number of alphabets, digits, vowels, consonants, spaces and special characters. 9. Write a C Program to Reverse a String using Pointer 10. Write a C Program to Swap Two Numbers using Pointers 11. Write a C Program to demonstrate student structure to read & display records of n students. 12. Write a C Program to demonstrate the difference between structure &union.

Note: Student has to execute a minimum of 10 programs in each part to complete the Lab course

Evaluation Scheme for Lab Examination

Assessment Criteria		**Marks**
Activity – 1 from Part A	Write up on the activity/ task	3
	Demonstration of the activity/ task	07
Activity-2 from Part B	Write up on the activity/ task	3
	Demonstration of the activity/ task	07
Viva based on Lab Activities		05
Total		**25**

OPEN-ELECTIVE SYLLABUS :

Year	I	**Course Code:** 21BSC1O1CS1	**Credits**	03
Sem.	1	**Course Title:** C Programming Concepts	**Hours**	40

Course Pre-requisites, if any	NA	
Formative Assessment Marks: 40	Summative Assessment Marks: 60	Duration of ESA:.02 hrs.

Course Outcomes	At the end of the course the student should be able to: 1. Read, understand and trace the execution of programs written in C language 2. Write the C code for a given problem 3. Perform input and output operations using programs in C 4. Write programs that perform operations on arrays 5. Write user defined functions to perform a task

Unit No.	**Course Content**	**Hours**
Unit I	**Introduction to C Programming:** Overview of C; History and Features of C; Structure of a C Program with Examples; Creating and Executing a C Program; Compilation process in C. **C Programming Basic Concepts:** C Character Set; C tokens - keywords, identifiers, constants, and variables; Data types; Declaration & initialization of variables; Symbolic constants.	10
Unit II	**Input and output with C:** Formatted I/O functions – *printf*and*scanf,* control stings and escape sequences, output specifications with *printf*functions; Unformatted I/O functions to read and display single character and a string - *getchar*, *putchar*, *gets* and *puts* functions **C Operators & Expressions:** Arithmetic operators; Relational operators; Logical operators; Assignment operators; Increment & Decrement operators; Bitwise operators; Conditional operator; Special operators; Operator Precedence and Associatively; Evaluation of arithmetic expressions; Type conversion. **Control Structures:** Decision making Statements - S*imple if, if_else*, *nested if_else, else_if ladder, Switch Case*, *goto, break* &*continue* statements	10
Unit III	Looping Statements - Entry controlled and exit controlled statements, *while, do-while, for* loops, Nested loops. **Derived data types in C:** Arrays: One Dimensional arrays - Declaration, Initialization and Memory representation; Two Dimensional arrays - Declaration, Initialization and Memory representation. **Strings:** Declaring & Initializing string variables; String handling functions - *strlen, strcmp, strcpy and strcat;* Character handling functions - *toascii, toupper, tolower, isalpha, isnumericetc*	10

Unit IV	**User Defined Functions:** Need for user defined functions; Format of C user defined functions; Components of user defined functions - return type, name, parameter list, function body, return statement and function call; Categories of user defined functions - With and without parameters and return type.	10
Recommended Leaning Resources		
Print Resources	**Text Books:** 1. C: The Complete Reference, By HerbertSchildt. 2. C Programming Language, By Brain W.Kernighan 3. Kernighan & Ritchie: The C Programming Language(PHI) **References** 1. E. Balaguruswamy: Programming in ANSI C(TMH) 2. Kamthane: Programming with ANSI and TURBO C (PearsonEducation) 3. V. Rajaraman: Programming in C (PHI –EEE) 4. S. Byron Gottfried: Programming with C(TMH) 5. YashwantKanitkar: Let usC	

Semester: II

Year	I	**Course Code:** 21BSC2C2CS2L	**Credits**	04
Sem.	2	**Course Title:** Data Structures using C	**Hours**	52

Course Pre-requisites, if any	NA	
Formative Assessment Marks: 40	Summative Assessment Marks: 60	Duration of ESA: 02 hrs.

Course Outcomes	After completing this course satisfactorily, a student will be able to: • Describe how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used byalgorithms • Describe common applications for arrays, records, linked structures, stacks, queues, trees, andgraphs • Write programs that use arrays, records, linked structures, stacks, queues, trees, andgraphs • Demonstrate different methods for traversingtrees • Compare alternative implementations of data structures with respect toperformance • Describe the concept of recursion, give examples of itsuse • Discuss the computational efficiency of the principal algorithms for sorting andsearching	
Unit No.	**Course Content**	**Hours**
Unit I	Introduction to data structures: Definition; Types of data structures - Primitive & Non-primitive, Linear and Non-linear; Operations on data structures.Algorithm Specification, Performance Analysis, Performance Measurement Recursion: Definition; Types of recursions; Recursion Technique Examples - Fibonacci numbers,GCD, Binomial coefficient ${}^{n}C_{r}$, Towers of Hanoi; Comparison between iterative and recursive functions.Arrays: Basic Concepts – Definition, Declaration, Initialization, Operations on arrays; Types of arrays; Arrays as abstract data types (ADT); Representation of Linear Arrays in memory;	13
Unit II	Traversinglinear arrays;Insertinganddeletingelements;Sorting– Selectionsort,Bubblesort,Quick sort, Selection sort, Insertion sort; Searching - Sequential Search, Binary search; Iterative and Recursive searching; Stacks:BasicConcepts– DefinitionandRepresentationofstacks;Operationsonstacks;Applications of stacks; Infix, postfix and prefix notations; Conversion from infix	13

	to postfix using stack; Evaluation of postfix expression using stack;	
Unit III	Queues: Basic Concepts – Definition and Representation of queues; Types of queues - Simple queues, Circular queues, Double ended queues, Priority queues; Operations on Simple queues;Dynamic memory allocation: Static & Dynamic memory allocation; Memory allocation and de- allocation functions - malloc, calloc, reallocandfree.Linked list: Basic Concepts – Definition and Representation of linked list, Types of linked lists - Singly linked list, Doubly liked list, Header liked list, Circular linked list; Representation of Linked list in Memory; Operations on Singly linked lists – Traversing, Searching, Insertion, Deletion; Memory allocation; Garbage collection	13
UnitIV	Trees: Definition; Tree terminologies –node, root node, parent node, ancestors of a node, siblings, terminal & non-terminal nodes, degree of a node, level, edge, path, depth; Binary tree: Type of binary trees - strict binary tree, complete binary tree, binary search tree and heap tree; Array representation of binary tree. Traversal of binary tree; *preorder, inorder*and*Postorder*traversal; Reconstruction of a binary tree when any two of the traversals are given.	13
Recommended Leaning Resources		
Print Resources	**Reference Books:** 1. Ellis Horowitz and SartajSahni: Fundamentals of Data Structures 2. Tanenbaum: Data structures using C (Pearson Education) 3. Kamathane: Introduction to Data structures (Pearson Education) 4. Y. Kanitkar: Data Structures Using C(BPB) 5. Kottur: Data Structure Using C 6. Padma Reddy: Data Structure Using C 7. Sudipa Mukherjee: Data Structures using C – 1000 Problems and Solutions (McGraw Hill Education,2007)	

Year	I	**Course Code:** 21BSC2C2CS2P	**Credits**	02
Sem.	I	**Course Title:**Data Structure Lab	**Hours**	45
Course Pre-requisites, if any:		Knowledge of Programming		
Formative Assessment Marks: 25		Summative Assessment Marks: 25	Duration of ESA: 03 hrs.	

Part A:

1. Write a C Program to find GCD using recursive function
2. Write a C Program to display Pascal Triangle using binomial function
3. Write a C Program to generate n Fibonacci numbers using recursive function.
4. Write a C Program to implement Towers of Hanoi.
5. Write a C Program to implement dynamic array, find smallest and largest element of the array.
6. Write a C Program to create two files to store even and odd numbers.
7. Write a C Program to create a file to store student records.
8. Write a C Program to read the names of cities and arrange them alphabetically.
9. Write a C Program to sort the given list using selection sort technique.
10. Write a C Program to sort the given list using bubble sort technique.

PART B:

1. Write a C Program to sort the given list using insertion sort technique.
2. Write a C Program to sort the given list using quick sort technique.
3. Write a C Program to sort the given list using merge sort technique.
4. Write a C Program to search an element using linear search technique.
5. Write a C Program to search an element using recursive binary search technique.
6. Write a C Program to implement Stack.
7. Write a C Program to convert an infix expression to postfix.
8. Write a C Program to implement simple queue.
9. Write a C Program to implement linear linked list.
10. Write a C Program to display traversal of a tree.

Evaluation Scheme for Lab Examination

Assessment Criteria		**Marks**
Activity – 1 from Part A	Write up on the activity/ task	3
	Demonstration of the activity/ task	07
Activity-2 from Part B	Write up on the activity/ task	3
	Demonstration of the activity/ task	07
Viva based on Lab Activities		05
Total		**25**

OPEN-ELECTIVE SYLLABUS:

Year	I	**Course Code:** 21BSC2O2CS2	**Credits**	03
Sem.	II	**Course Title:** Web Designing	**Hours**	40
Course Pre-requisites, if any		NA		
Formative Assessment Marks: 40		Summative Assessment Marks: 60	Duration of ESA:.02 hrs.	
Course Outcomes	At the end of the course the student should be able to: 1. Read, understand and trace the execution of programs 2. Write the code for a given problem 3. Perform input and output operations using programs 4. Write user defined functions to perform a task			

Unit No.	**Course Content**	**Hours**
Unit I	History of Internet, The World Wide Web, Web Browser, Web Server, URL, Working of Web, Web Page, Types of Web Pages, Web Content, Websites, Home Pages, Building Website, Website building tools; Web graphics design, basic tips for graphics design, Web Designing tools: Gimp-image resize, crop, edit background, save with different file types. Introduction to web programming: what is web programming? , web programming languages.	10
Unit II	Introduction to XHTML- Basic Syntax, Standard structure, Basic text markup, Images, Hypertext, Links, Lists, Tables, Forms-<form>,<input>,<label>,<select>,<textarea> tags and action buttons(submit and reset). CSS- Introduction, Levels of style sheets, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The box model, Background images, The and tags.	10
Unit III	JavaScript: Object orientation and JavaScript; General syntactic characteristics; Primitives, operations, and expressions; Screen output and keyboard input; Control statements; Object creation and modification; Arrays; Functions; Constructor; Pattern matching using regular expressions; Errorsin scripts; Examples.	10
Unit IV	Introduction to XML, Syntax of XML , XML document structure, Displaying raw XML documents, Displaying XML documents with CSS,XSLT Stylesheets and Displaying XML documents with XSLT. Web Design: Concepts of effective web design, Web design issues including Browser, Bandwidth and Cache, Display resolution, Look and Feel of the Website, Page Layout and linking, User centric design, Sitemap, Planning and publishing website, Designing effective navigation	10

Recommended Leaning Resources	
Print Resources	**Text Books:** **1.** Robert W. Sebestra, "Programming the World Wide Web", 7th Edition /4th edition Addison Wesley Publication,2013. **References:** 1. Developing Web Applications, Ralph Moseley and M. T. Savaliya, Wiley-India 2. Web Technologies, Black Book, dreamtech Press 3. HTML 5, Black Book, dreamtech Press 4. Web Design, Joel Sklar, Cengage Learning 5. Developing Web Applications in PHP and AJAX, Harwani, McGrawHill 6. Internet and World Wide Web How to program, P.J. Deitel& H.M. Deitel, Pearson

Rani Channamma University Belagavi

Vidyasangama, NH-04, Bhutaramanahatti, Belagavi – 591 156

Bachelor of Science (B.Sc.) in Computer Science

Syllabus for III and IV Semester

(as per National Education Policy – 2020)

2022-23 onwards

RANI CHANNAMMA UNIVERSITY

Vidyasangama, NH-04, Bhutaramanahatti, Belagavi – 591 156

SYLLABUS

Bachelor of Science (B.Sc.) in Computer Science

(as per National Education Policy – 2020)

Submitted by

Dr. Parashuram Bannigidad
Chairperson BoS (UG) – Rani Channamma University, Belagavi

Members of Board of Studies

1. Dr. Vijayalaxmi M Belagumpi — Member
   Assistant Professor,
   Dept. of Computer Science,
   GFGC Vijayapur
2. Dr. Bhagirathi Halalli — Member
   Assistant Professor,
   Dept. of Computer Science,
   GFGC Raibag

Curriculum Structure for B.Sc. (Computer Science) III SEM Program of RCUB as per NEP 2020 w.e.f. 2022-23

SEMESTER-III										
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
L5	21BSC3L5LK3	Kannada	40	60	100	4	-	-	3	2
	21BSC3L5LFK3	Functional Kannada								
L6	21BSC3L6EN3	English	40	60	100	4	-	-	3	2
	21BSC3L6HI3	Hindi								
	21BSC3L6SN3	Sanskrit								
	21BSC3L6TE3	Telugu								
	21BSC3L6UR3	Urdu								
DSC3	21BSC3C3CS1L	Object Oriented Programming in JAVA	40	60	100	4	-	-	4	2
	21BSC3C3CS1P	JAVA Lab	25	25	50	-	-	4	2	3
DSC3	Another Department Code	Another Department Course Title with lab	40	60	100	4	-	-	4	2
			25	25	50	-	-	4	2	3
AECC2	21BSC4AE1CI2	Constitution of India	20	30	50	1	-	2	2	1hr30min
VBC5	21BSC3V5PE3	Physical Education- Sports	25	-	25	-	-	2	1	-
VBC6	21BSC3V6NC2	NCC/NSS/R&R(S&G) / Cultural	25	-	25	-	-	2	1	-
OEC3	21BSC3O3CS5	Python Programming Concepts	40	60	100	3	-	-	3	2
			Total Marks		**700**	**Semester Credits**			**25**	

Curriculum Structure for B.Sc. (Computer Science) IV SEM Program of RCUB as per NEP 2020 w.e.f. 2022-23

SEMESTER-IV										
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
L7	21BSC4L7LK4	Kannada	40	60	100	4	-	-	3	2
	21BSC4L7LFK4	Functional Kannada								
L8	21BSC4L8EN4	English	40	60	100	4	-	-	3	2
	21BSC4L8HI4	Hindi								
	21BSC4L8SN4	Sanskrit								
	21BSC4L8TE4	Telugu								
	21BSC4L8UR4	Urdu								
DSC4	21BSC4C2CS2L	Database Management Systems	40	60	100	4	-	-	4	2
	21BSC4C2CS2P	DBMS Lab	25	25	50	-	-	4	2	3
DSC4	Another Department Code	Another Department Course Title	40	60	100	4	-	-	4	2
			25	25	50	-	-	4	2	3
SEC2	21BSC3SE2ES2	Artificial Intelligence	20	30	50	3	-	-	2	1hr30min
VBC7	21BSC4V5PE4	Physical Education- Sports	25	-	25	-	-	2	1	-
VBC8	21BSC4V6NC3	NCC/NSS/R&R(S&G) / Cultural	25	-	25	-	-	2	1	-
OEC4	21BSC4O3CS7	Financial Education and Investment Awareness	20	30	50	3	-	-	2	1hr30min
				Total Marks	**650**			**Semester Credits**	**24**	
Exit option with Diploma in Science (with the completion of courses equal to a minimum of 96 credits) OR continue studies with Major and Minor					**2750**				**99**	

Syllabus for BSc III and IV Semesters

Subject: Computer Science

Sem	Discipline Specific Core Courses(DSC)	Hour / Week	
		Theory	Lab
III	**DSC-3:** Object Oriented Programming Concepts and Programming in JAVA	**4**	-
	DSC-3 Lab: JAVA Lab	-	**4**
	OEC: Python Programming Concepts	**3**	**-**
IV	**DSC-4:** Database Management Systems	**4**	-
	DSC-4 Lab: DBMS Lab	-	**4**

Syllabus for BSc III Semester

Course Title: **Object Oriented Programming Concepts and Programming in Java**	Course code: 21BSC3C3CS1L
Total Contact Hours: 52	Course Credits: 04
Formative Assessment or IA Marks: 40	Duration of SEE/Exam: 02 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:

- Explain the object-oriented concepts and JAVA.
- Write JAVA programs using OOP concepts like Abstraction, Encapsulation, Inheritance and Polymorphism.
- Implement Classes and multithreading using JAVA.
- Demonstrate the basic principles of creating Java applications with GUI.

DSC3: Object Oriented Programming Concepts and Programming in Java

Unit	Description	Hours
1	**Introduction to Java:** Basics of Java programming, Data types, Variables, Operators, Control structures including selection, Looping, Java methods, Overloading, Math class, Arrays in java.	09
2	**Objects and Classes:** Basics of objects and classes in java, Constructors, Finalizer, Visibility modifiers, Methods and objects, Inbuilt classes like String, Character, String Buffer, File, this reference.	10
3	**Inheritance and Polymorphism:** Inheritance in java, Super and sub class, Overriding, Object class, Polymorphism, Dynamic binding, Generic programming, Casting objects, Instance of operator, Abstract class, Interface in java, Package in java, UTIL package.	09
4	**Event and GUI programming:** Event handling in java, Event types, Mouse and key events, GUI Basics, Panels, Frames, Layout Managers: Flow Layout, Border Layout, Grid Layout, GUI components like Buttons, Check Boxes, Radio Buttons, Labels, Text Fields, Text Areas, Combo Boxes, Lists, Scroll Bars, Sliders, Windows, Menus, Dialog Box, Applet and its life cycle, Introduction to swing, Exceptional handling mechanism.	12
5	**I/O programming:** Text and Binary I/O, Binary I/O classes, Object I/O, Random Access Files. Multithreading in java: Thread life cycle and methods, Runnable interface, Thread synchronization, Exception handling with try catch-finally, Collections in java, Introduction to JavaBeans and Network Programming	12

References:

1. Programming with Java, By E Balagurusamy – A Primer, 4th Edition, McGraw Hill Publication.
2. Core Java Volume I – Fundamentals, By Cay S. Horstmann, Prentice Hall.
3. Object Oriented Programming with Java: Somashekara M.T., Guru, D.S., Manjunatha K.S, 1st Edition, PHI Learning 2017.
4. Java 2 - The Complete Reference, Herbert Schildt, 5th Edition, McGraw Hill Publication, 2017.
5. Java - The Complete Reference, Herbert Schildt, 7th Edition, McGraw Hill Publication, 2017.

Year	II	**Course Code:** 21BSC3C3CS1P	**Credits**	02
Sem.	III	**Course Title: JAVA LAB**	**Hours**	40
Course Pre-requisites, ifany:		Knowledge of Programming		
Formative Assessment Marks: 25		Summative Assessment Marks: 25	Duration of ESA: 03 hrs.	

Practice Labs

1. Program to print the following triangle of numbers

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

2. Program to simple java application, to print the message,"Welcome to java"
3. Program to display the month of a year. Months of the yearshould be held in an array.
4. Program to find the area of rectangle.
5. program to demonstrate a division by zero exception
6. Program to create a user defined exception say Pay Out ofBounds.

Part A: Programming Lab – Java Fundamentals – OOPS inJAVA

1. Program to assign two integer values to X and Y. Using the "if"statement the output of the program should display a message whether X is greater than Y.
2. Program to list the factorial of the numbers 1 to 10. Tocalculate the factorial value, use while loop. (Hint: Fact of 4 = 4*3*2*1)
3. Program to find the area and circumference of the circle by accepting the radius from the user.
4. Program to add two integers and two float numbers. When no arguments are supplied, give a default value to calculate the sum. Use function overloading.
   Program to perform mathematical operations. Create a class called AddSub with methods to add and subtract. Create another class called MulDiv that extends from AddSub class to use the member data of the super class. MulDiv should have methods to multiply and divide A main function should access the methods and perform the mathematical operations.
6. Program with class variable that is available for all instancesof a class. Use static variable declaration. Observe the changes that occur in the object"s member variable values.
7. Program to create a student class with following attributes; Enrollment No: Name, Mark of sub1, Mark of sub2, mark ofsub3, Total Marks. Total of the three marks must becalculated only when the student passes in all three subjects.The passing mark for each subject is 50. If a candidate fails inany one of the subjects his total mark must be declared aszero. Using this condition write a constructor for this class.Write separate functions for accepting and displaying studentdetails. In the main

method create an array of three studentobjects and display the details.

8. Write a program to demonstrate multiple inheritance and useof Implementing Interfaces
9. Illustrate creation of thread by

   a) Extending Thread class. b) Implementing RunnableInterfaces
10. Write a program to demonstrate multiple inheritance and use of implementing Interfaces.

PART B: Exception Handling & GUI Programming

1. Program to catch Negative Array Size Exception. This exception is caused when the array size is initialized to negative values.
2. Program to demonstrate exception handling with try, catch and finally.
3. Program which create and displays a message on the window
4. Program to draw several shapes in the created window
5. Programto create a 4×4 grid and fills it in with 15 buttons, each labeled with its index.
6. Program which creates a frame with two buttons father and mother. When we click the father button the name of the father, his age and designation must appear. When we click mother button similar details of mother also appear.
7. Create a frame which displays your personal details with respect to a button click
8. Program to create a window with Text Fields and Buttons. The"ADD" button adds the two integers and display the result. The "CLEAR" button shall clear all the text fields.
9. Program to create a window, when we press M or m, the window displays “Good morning”, A or a, the window display’s Good Afternoon”, E or e, the window displays “Good Evening”, N or n, the window displays “Hello”.
10. Demonstrate the various mouse handling events using suitable example.

Evaluation Scheme for Java Lab Examination

Assessment Criteria		Marks
Program – 1 from Part A	Writing the Program	03
	Execution and Formatting	07
Program -2 from Part B	Writing the Program	03
	Execution and Formatting	07
Viva Voice		05

Open Elective for III Semester

Course Code: 21BSC3O3CS5 Course Title: **Python Programming Concepts**	Course Credits: 3 (3L+0T+0P)
Semester: III	Duration of SEE: 02 Hour
Total Contact Hours: 42	Summative Assessment Marks : 60 Marks Formative AssessmentMarks: 40 Marks

Course Outcomes (COs):

- Explain the fundamentals of Computers.
- Explain the basic concepts of Python Programming.
- Demonstrate proficiency in the handling of loops and the creation of functions.
- Identify the methods to create and store strings.

Unit I Fundamentals of Computers 10 Hrs

Introduction to Computers - Computer Definition, Characteristics of Computers, Evolution and History of Computers, Types of Computers, Basic Organization of a Digital Computer; Number Systems – different types, conversion from one number system to another; Computer Codes – BCD, Gray Code, ASCII and Unicode; Boolean Algebra – Boolean Operators with Truth Tables; Types of Software – System Software and Utility Software; Computer Languages - Machine Level, Assembly Level & High Level Languages, Translator Programs – Assembler, Interpreter and Compiler; Planning a Computer Program - Algorithm, Flowchart and Pseudo code with Examples.

Unit II Python Basics 10 Hrs

Introduction to Features and Applications of Python; Python Versions; Installation of Python; Python Command Line mode and Python IDEs; Simple Python Program. Identifiers; Keywords; Statements and Expressions; Variables; Operators; Precedence and Association; Data Types; Indentation; Comments; Built-in Functions- Console Input and Console Output, Type Conversions; Python Libraries; Importing Libraries with Examples; Illustrative programs.

Unit III 08 Hrs

Python Control Flow: Types of Control Flow; Control Flow Statements- if, else, elif, while loop, break, continue statements, for loop Statement; range () and exit () functions; Illustrativeprograms.

Unit IV 08 Hrs

Python Functions: Types of Functions; Function Definition- Syntax, Function Calling, Passing Parameters/arguments, the return statement; Default Parameters; Command line Arguments; Key Word Arguments; Illustrative programs.

Unit V **06 Hrs**

Strings: Creating and Storing Strings; Accessing Sting Characters; the str() function; Operations on Strings- Concatenation, Comparison, Slicing and Joining, Traversing; Format Specifiers; Escape Sequences; Raw and Unicode Strings; Python String Methods; Illustrative programs.

References

1. Computer Fundamentals (BPB), P. K. Sinha & Priti Sinha
2. Think Python How to Think Like a Computer Scientist, Allen Downey et al., 2nd Edition, Green Tea Press. Freely available online 2015. @https://www.greenteapress.com/thinkpython/thinkCSpy.pdf
3. Introduction to Python Programming, Gowrishankar S et al., CRC Press, 2019.
4. http://www.ibiblio.org/g2swap/byteofpython/read/
5. http://scipy-lectures.org/intro/language/python_language.html
6. https://docs.python.org/3/tutorial/index.html

Syllabus for BSc IV Semester

Course Title: **Database Management System**	Course code: 21BSC4C2CS2L
Total Contact Hours: 52	Course Credits: 04
Formative Assessment or IA Marks: 40	Duration of SEE/Exam: 02 Hours
Summative Assessment Marks: 60	

Course Outcomes (COs):

At the end of the course, students will be able to:

- Explain the various database concepts and the need for database systems.
- Identify and define database objects, enforce integrity constraints on a database using DBMS.
- Demonstrate a Data model and Schemas in RDBMS.
- Identify entities and relationships and draw ER diagram for a given real-world problem.
- Convert an ER diagram to a database schema and deduce it to the desired normal form.
- Formulate queries in Relational Algebra, Structured Query Language (SQL) for database manipulation.
- Explain the transaction processing and concurrency control techniques.

DSC7: Database Management System (DBMS)

Unit	Description	Hours
1	**Database Architecture:** Introduction to Database system applications. Characteristics and Purpose of database approach. People associated with Database system. Data models. Database schema. Database architecture. Data independence. Database languages, interfaces, and classification of DBMS.	10
2	**E-R Model:** Entity-Relationship modeling: E – R Model Concepts: Entity, Entity types, Entity sets, Attributes, Types of attributes, key attribute, and domain of an attribute. Relationships between the entities. Relationship types, roles and structural constraints, degree and cardinality ratio of a relationship. Weak entity types, E -R diagram.	10
3	**Relational Data Model:** Relational model concepts. Characteristics of relations. Relational model constraints: Domain constrains, key constraints, primary & foreign key constraints, integrity constraints and null values. Relational Algebra: Basic Relational Algebra operations. Set theoretical	12

	operations on relations. JOIN operations Aggregate Functions and Grouping. Nested Sub Queries-Views. Introduction to PL/SQL & programming of above operations in PL/SQL	
4	**Data Normalization:** Anomalies in relational database design. Decomposition. Functional dependencies. Normalization. First normal form, Second normal form, Third normal form. Boyce-Codd normal form.	09
5	**Query Processing Transaction Management:** Introduction Transaction Processing. Single user & multiuser systems. Transactions: read & write operations. Need of concurrency control: The lost update problem, Dirty read problem. Types of failures. Transaction states. Desirable properties (ACID properties) of Transactions. Concurrency Control Techniques: Locks and Time stamp Ordering. Deadlock & Starvation.	11

References:

1. Fundamentals of Database Systems, Ramez Elamassri, Shankant B. Navathe, 7th Edition, Pearson, 2015
2. An Introduction to Database Systems, Bipin Desai, Galgotia Publications, 2010.
3. Introduction to Database System, C J Date, Pearson, 1999.
4. Database Systems Concepts, Abraham Silberschatz, Henry Korth, S.Sudarshan, 6th Edition, McGraw Hill, 2010.
5. Database Management Systems, Raghu Rama Krishnan and Johannes Gehrke, 3rd Edition, McGraw Hill, 2002

Year	II	**Course Code:** 21BSC4C2CS2P	**Credits**	02
Sem.	III	**Course Title: DBMS LAB**	**Hours**	40
Course Pre-requisites, ifany:		Knowledge of Programming		
Formative Assessment Marks: 25		Summative Assessment Marks: 25	Duration of ESA: 03 hrs.	
		Practical: CO: Student would be able to create tables, execute queries and PL/SQL programs. 1. Execute a single line query and group functions. 2. Execute DDL Commands. 3. Execute DML Commands 4. Execute DCL and TCL Commands. 5. Implement the Nested Queries. 6. Implement Join operations in SQL 7. Create views for a particular table 8. Implement Locks for a particular table 9. Write PL/SQL procedure for an application using exception handling. 10. Write PL/SQL procedure for an application using cursors. 11. Write a PL/SQL procedure for an application using functions 12. Write a PL/SQL procedure for an application using package		

Evaluation Scheme for DBMS Lab Examination

Assessment Criteria		Marks
Program – 1 from Part A	Writing the Program	03
	Execution and Formatting	07
Program -2 from Part B	Writing the Program	03
	Execution and Formatting	07
Viva Voice		05

Semester End Exam Question Paper Pattern

Duration of the examination: 2hour **Max. Marks:60**

Section A

Answer any TEN from the following, each carries 2 marks: [10X2=20]

1. ----------
2. ----------
3. ----------
4. ----------
5. ----------
6. ----------
7. ----------
8. ----------
9. ----------
10. ----------
11. ----------
12. ----------

Section B

Answer any FOUR from the following questions each carries 5 marks.

[4X5=20]

13. ----------
14. ----------
15. ----------
16. ----------
17. ----------

Section C

Answer Any two from the following questions each carries 10 marks.

(The Question may consist two sub-questions)

[2X10=20]

18. ----------
19. ----------
20. ----------

Theory Paper IA 40 Marks distribution

	C1	C2	Total Marks
First IA	Test-1: 15marks	Assignment/Activity-1: 05Marks	**20**
Second IA	Test-2: 15marks	Assignment/Activity-2 : 05Marks	**20**
	30	**10**	**40**

Theory Paper IA 20 Marks distribution

	C1	C2	Total Marks
First IA	Test-1: 10 marks	---	**10**
Second IA	Test-2: 5marks	Assignment/Activity-2 : 05	**10**
	15	**05**	**20**

Rani Channamma University Belagavi

Vidyasangama, NH-04, Bhutaramanahatti, Belagavi – 591 156

Bachelor of Science (B.Sc.) in Computer Science

Syllabus for V and VI Semester

(as per National Education Policy – 2020)

2023-24 onwards

RANI CHANNAMMA UNIVERSITY

Vidyasangama, NH-04, Bhutaramanahatti, Belagavi – 591 156

SYLLABUS

Bachelor of Science (B.Sc.) in Computer Science

(as per National Education Policy – 2020)

Submitted by

Dr. Parashuram Bannigidad
Chairperson BoS (UG) – Rani Channamma University, Belagavi

Members of Board of Studies

1. Dr. Vijayalaxmi M. Belagumpi — Member
   Assistant Professor,
   Dept. of Computer Science,
   GFGC Vijayapur
2. Dr. Bhagirathi Halalli — Member
   Assistant Professor,
   Dept. of Computer Science,
   GFGC Raibag
3. Dr. Basavaraj K Galagali — Member
   Assistant Professor,
   Coordinator of BCA
   BLDEA'S Commerce, BHS
   Arts and TGP Science
   College, Jamakhandi

Curriculum Structure for B.Sc. (Computer Science) V and VI Sem Program of RCUB as per NEP 2020 w.e.f. 2023-24

SEMESTER-V										
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
DSC5	21BSC5CSMJ1L	Programming in PYTHON	40	60	100	4	-	-	4	2
	21BSC5CSMJ1P	PYTHON Programming lab	25	25	50	-	-	4	2	3
DSC6	21BSC5CSMJ2L	Computer Networks	40	60	100	4	-	-	4	2
	21BSC5CSMJ2P	Computer Networks Lab	25	25	50	-	-	4	2	3
DSC5	Another Department Subject	Another Department Course Title	40	60	100	4	-	-	4	2
	Another Department Subject Practical	Another Department Course Title lab	25	25	50	-	-	4	2	3
DSC6	Another Department Subject	Another Department Course Title	40	60	100	4	-	-	4	2
	Another Department Subject Practical	Another Department Course Title lab	25	25	50	-	-	4	2	3
SEC4	21BSC6SE4CS3	Cyber Security	20	30	50	2	-	2	3	1h.30min
Total Marks					**650**	**Semester Credits**			**27**	

Curriculum Structure for B.Sc. (Computer Science) VI Sem Program of RCUB as per NEP 2020 w.e.f. 2023-24

SEMESTER-VI										
Category	**Course code**	**Title of the Paper**	**Marks**			**Teaching hours/week**			**Credit**	**Duration of exams (Hrs)**
			IA	**SEE**	**Total**	**L**	**T**	**P**		
DSC7	21BSC6CSMJ1L	Web Technologies	40	60	100	4	-	-	4	2
	21BSC6CSMJ1P	Web Technology lab	25	25	50	-	-	4	2	3
DSC8	21BSC6CSMJ2L	Statistical Computing & R Programming	40	60	100	4	-	-	4	2
	21BSC6CSMJ2P	R Programming Lab	25	25	50	-	-	4	2	3
DSC7	Another Department Subject	Another Department Course Title	40	60	100	4	-	-	4	2
	Another Department Subject Practical	Another Department Course Title lab	25	25	50	-	-	4	2	3
DSC8	Another Department Subject	Another Department Course Title	40	60	100	4	-	-	4	2
	Another Department Subject Practical	Another Department Course Title lab	25	25	50	-	-	4	2	3
Proj 1	21BSC6SE4CS1	Project	25	25	50				2	2
Total Marks					**650**	**Semester Credits**			**26**	
Students Exiting the programme after 3-years will be awarded UG degree in Disciplines A and B as double majors upon securing 136 credits and satisfying the minimum credit requirements under each category of courses prescribed										

Syllabus for BSc V Semester

Semester: V

Program Name	**B.Sc.**		Semester	**V**
Course Title	**Programming in Python (Theory)**			
Course Code:	**DSC5**		No. of Credits	**04**
Contact hours	**52 Hours**		Duration of SEA/Exam	**2 hours**
Formative Assessment Marks	**40**		Summative Assessment Marks	**60**

Course Pre-requisite(s): None

Course Outcomes (COs): After the successful completion of the course, the student will be able to:

CO1	Setup Python to develop simple applications
CO2	Understand the basic concepts in Python Programming
CO3	Learn how to write, debug and execute Python programs
CO4	Understand and demonstrate the use of advanced data types such as tuples, dictionaries and lists, Tuples and Sets
CO5	Design solutions for problems using object-oriented concepts in Python
CO6	Use and apply the different Python Libraries for GUI Interface, Data Analysis and Data Visualisation.
C07	Extend the knowledge of python programming to build successful career in software development.

Unit	Contents	52 Hrs
Unit I	**Introduction**: Introduction, Overview, Features and Applications of Python; Python Versions; Getting Started With Python; Python Command Line mode and Python IDEs; Indentation; Comments: **Python Basics:** Identifiers; Keywords; Variables; Data Types; Operators; Precedence and Association; Statements and Expressions; **Python Control Flow:** Types of Control Flow; Control Flow Statements- if, else, elif, while loop, break, continue statements, for loop Statement; range () and exit () functions.	12
Unit II	**Functions:** Introduction; Types of Functions; Built- in Functions- Console Input and Console Output, Type Conversions; Python Libraries; Importing Libraries with Examples. User Defined Functions-Parameters, arguments, function calls, return statement, Scope and Lifetime of Variables in Functions, Writing Python Scripts using functions. **Strings:** Creating and Storing Strings; Accessing Sting Characters; the str() function; Operations on Strings- Concatenation, Comparison, Slicing and Joining, Traversing; Format Specifiers; Escape Sequences; Raw and Unicode Strings; Python String Methods.	10

Unit III	**Lists:** Creating Lists; Built-in Operations on Lists; Built-in Functions on Lists; Implementation of Stacks and Queues using Lists; Nested Lists. **Dictionaries:** Creating Dictionaries; Operations on Dictionaries; Built-in Functions on Dictionaries; Dictionary Methods; Populating and Traversing Dictionaries. **Tuples and Sets:** Creating Tuples; Operations on Tuples; Built-in Functions on Tuples; Tuple Methods; Creating Sets; Operations on Sets; Built-in Functions on Sets; Set Methods.	10
Unit IV	**Exception Handling:** Types of Errors; Exceptions; Exception Handling using try, except and finally. **File Handling:** File Types; Operations on Files– Create, Open, Read, Append and Write, Close Files; File Names and Paths; Format Operator. **Object Oriented Programming:** Classes and Objects; Creating Classes and Objects; Constructor Method; Classes with Multiple Objects; Objects as Arguments; Objects as Return Values; Inheritance- Single and Multiple Inheritance, Multilevel and Multipath Inheritance; Encapsulation- Definition, Private Instance Variables; Polymorphism- Definition, Operator Overloading.	10
Unit V	**GU Interface:** The Tkinter Module-; Widgets-Button, Canvas, Check button, entry, frame, label, List Box, Menu Button, Radio Button, Message, Scale, Scrollbar, Text, Spin Box, Message Box, Label Frame, Paned Window ; Layout Management- pack, grid and place. **Python SQLite:** The SQLite3 module; SQLite Methods- connect, cursor, execute, close; Connect to Database; Create Table; Operations on Tables- Insert, Select, Update. Delete and Drop Records.	10

Pedagogy: Lecture/ PPT/ Videos/ Animations/ Role Plays/ Think-Pair-Share/ Predict-Observe-Explain/ Demonstration/ Concept mapping/ Case Studies examples/ Tutorial/ Activity/ Flipped Classroom/ Jigsaw/ Field based Learning/ Project Based Learning/ Mini Projects/ Hobby Projects/ Forum Theatre/ Dance/ Problem Based Learning/ Game Based Learning/ Group Discussion/ Collaborative Learning/ Experiential Learning / Self Directed Learning etc.

References	
1	**Introduction to Computing and Problem Solving Using Python**, E Balaguruswamy, McGrawHill, First Edition
2	**Think Python How to Think Like a Computer Scientist,** Allen Downey et al., 2nd Edition, 2015, Green Tea Press. Freely available online @ https://www.greenteapress.com/thinkpython/thinkCSpy.pdf
3	**Introduction to Python Programming**, Gowrishankar S et al.,2019, CRC Press
4	**Python Data Analytics**: Data Analysis and Science Using Pandas, matplotlib, and the Python Programming Language, Fabio Nelli, 2015, Apress®
5	**Advance Core Python Programming**, Meenu Kohli, 2021, BPB Publications
6	**Core PYTHON Applications Programming**, Wesley J. Chun, 3rd Edition, 2012, Prentice Hall
7	**Automate the Boring Stuff**, Al Sweigart, 2015, No Starch Press, Inc.
8	**Data Structures and Program Design Using Python**, D Malhotra et al., 2021, Mercury Learning and Information LLC
9	http://www.ibiblio.org/g2swap/byteofpython/read/
10	https://docs.python.org/3/tutorial/index.html

Course Title	**Python Programming Lab (Practical)**	Practical Credits	**02**
Course Code	**DSC5-Lab**	Contact Hours	**04** **Hours/week**
Formative Assessment	**25 Marks**	Summative Assessment	**25 Marks**

Practical Content

Part-A

1. Write a Python function to calculate the factorial of a number
2. Write a Python to generate Fibonacci Sequence
3. Write a Python program to get the sum of digits of a non-negative integer.
4. Write a Python program to create a module Calculation.py that contains functions to perform basic arithmetic operations.
5. Write a python program to reverse a string without using built-in functions.
6. Write a python program to generate random numbers.
7. Write a python program to display Multiplication Tables
8. Demonstrate importing the math module and perform any five math functions.
9. Write a Python class named Circle constructed by a radius and two methods which will compute the area and the perimeter of a circle.
10. Write a Python class named Rectangle constructed by a length and width and a method which will compute the area and perimeter of rectangle.

Part-B

1. Demonstrate usage of basic regular expression with match (), search (), findall (), sub () and split ().
2. Find the largest and smallest element in the list
3. Demonstrate use of Dictionaries to store and retrieve contact information.
4. Create SQLite Database and Write a Python program to demonstrate modification of an existing table data from SQLite Database
5. Write a python program that prompts the user for a number and handles a “ValueError”
6. Inherit a class Box that contains additional method volume. Override the perimeter method to compute perimeter of a Box.
7. Write a Python program to read a file line by line store it into an array.
8. Write a python program to create a class representing a basic bank account class with deposit and withdrawal methods.
9. Design Student Registration form using any 5 widgets using Tkinter Module.
10. Write a python program to create a GUI interface for temperature converter using Tkinter

Program Name	**B.Sc.**	Semester	**V**
Course Title	**Computer Networks (Theory)s**		
Course Code:	**DSC6**	No. of Credits	**04**
Contact hours	**52 Hours**	Duration of SEA/Exam	**2 hours**
Formative Assessment Marks	**40**	Summative Assessment Marks	**60**

Course Pre-requisite(s):

Course Outcomes (COs): After the successful completion of the course, the student will be able to:

CO1	Define various data communication components in networking.
CO2	Describe networking with reference to different types of models and topologies.
CO3	Understand the need for Network and various layers of OSI and TCP/IP reference model.
CO4	Explain various Data Communications media.
CO5	Describe the physical layer functions and components
CO6	Identify the different types of network topologies and Switching methods.
CO7	Describe various Data link Layer Protocols.
CO8	Identify the different types of network devices and their functions within a network.
CO9	Analyze and Interpret various Data Kink Layer and Transport Layer protocols.
CO10	Explain different application layer protocols.

Unit	Contents	52 Hrs
UNIT I	**Introduction:** Computer Network: Definition, Goals, Structure; Broadcast and Point-To-Point Networks; Network Topology and their various Types; Types of Network, Network software, Design issues for the layers, Connection-oriented vs. Connectionless service, Applications of Computer network, Protocols and Standards, The OSI Reference Model, The TCP/IP Protocol suite, Comparison between OSI and TCP/IP Reference model.	12
UNIT II	**Physical Layer:** Functions of Physical Layer**,** Analog signals, Digital signals, Transmission Impairment, Data Rate Limits, and Performance. Data Transmission Media: Guided Transmission Media, Magnetic Media, Twisted Pairs, Coaxial Cable, Power Lines, Fiber Optics, Wireless Transmission, Electromagnetic Spectrum, Radio Transmission, Microwave Transmission, Infrared Transmission, Light Transmission, Digital Modulation and Multiplexing, Public Switched Telephone Networks. Switching: Circuit switching, Message switching & Packet switching	10
UNIT III	**Data Link Layer:** Functions of Data Link Layer, Data Link Control: Framing, Flow and Error Control, Error Detection and Correction, High-Level Data Link Control (HDLC) & point — to — Point protocol (PPP), Channel Allocation Problem, Multiple Access: Radom Access (ALOHA, CSMA, CSMA/CD, CSMA/CA), Controlled Access(Reservation, Polling, Token Passing), Channelization(FDMA, TDMA, CDMA),	10

UNIT IV	**Wired LAN:** Ethernet Standards and FDDI, Wireless LAN: IEEE 802.1 Ix and Bluetooth Standards. **Transport Layer:** Functions of Transport Layer, Elements of Transport Protocols: Addressing, Establishing and Releasing Connection, Flow Control & Buffering, Error Control, Multiplexing & De-multiplexing, Crash Recovery,	10
UNIT V	**User Datagram Protocol (UDP):** User Datagram, UDP Operations, Uses of UDP, RPC, Principles of Reliable Data Transfer: Building a Reliable Data Transfer Protocol, Pipelined Reliable Data Transfer Protocol, Go Back-N(GBN), Selective Repeat (SR). **Application layer :** Functions of Application layer, Application Layer Protocols: DNS, DHCP, WWW, HTTP, HTTPs, TELNET, FTP, SMTP, POP, IIMAP	10

Pedagogy: Lecture/ PPT/ Videos/ Animations/ Role Plays/ Think-Pair-Share/ Predict-Observe-Explain/ Demonstration/ Concept mapping/ Case Studies examples/ Tutorial/ Activity/ Flipped Classroom/ Jigsaw/ Field based Learning/ Project Based Learning/ Mini Projects/ Hobby Projects/ Forum Theatre/ Dance/ Problem Based Learning/ Game Based Learning/ Group Discussion/ Collaborative Learning/ Experiential Learning / Self Directed Learning etc.

References	
1	Andrew S Tanenbaum, David. J. Wetherall, –Computer Networks‖, Pearson Education, 5th Edition,
2	Behrouz A. Forouzan, "Data Communications and Networking", Tata McGraw-Hill, Fourth Edition
3	Kurose and Ross, Computer Networking- A Top-Down approach, Pearson, 5th edition
4	William Stallings, Data and Computer Communications, 7th Edition,PHI.
4	http://highered.mheducation.com/sites/0072967757/index.html
7	Larry L. Peterson, Bruce S. Davie, –Computer Networks: A Systems Approach‖, Morgan Kaufmann Publishers, Fifth Edition, 2011.
8	Brijendrasingh, Data Communication and Computer Networks,PHI.

Course Title	**Computer Networks Laboratory (Practical)**	Practical Credits	**02**
Course Code	**DSC6 Lab**	Contact Hours	**04 Hours/week**
Formative Assessment	**25 Marks**	Summative Assessment	**25 Marks**

Practical Content

Part A:

1. Prepare hardware and software specification for basic computer system and Networking.
2. Study of different types of Network cables and practically implement the cross-wired cable and straight through cable using clamping tool.
3. Identifying the networking devices on a network.
4. Configure the IP address of the computer.
5. Create a basic network and share file and folders.
6. Study of basic network command and Network configuration commands.
7. Installation process of any open source network simulation software.

Part B:

1. Implement connecting two nodes using network simulator.
2. Implement connecting three nodes considering one node as a central node using network simulator. Implement a network to connect three nodes considering one node as a central node using network simulator
3. Implement bus topology using network simulator.
4. Implement star topology using network simulator.
5. Implement ring topology using network simulator.
6. Demonstrate the use of wireless LAN using network simulator.
7. Implement FTP using TCP bulk transfer using network simulator.

Links for open source simulation software:

- NS3 software: https://www.nsnam.org/releases/ns-3-30/download/
- Packet Tracer Software: https://www.netacad.com/courses/packet-tracer
- GNS3 software: https://www.gns3.com/

Syllabus for B. Sc VI Semester

Program Name	**B.Sc.**	Semester	**VI**
Course Title	**Web Technologies (Theory)**		
Course Code:	**DSC8**	No. of Credits	**04**
Contact hours	**52 Hours**	Duration of SEA/Exam	**2 hours**
Formative Assessment Marks	**40**	Summative Assessment Marks	**60**

Course Pre-requisite(s): Basic Knowledge About Programming and Internet and Web Browsing

Course Outcomes (COs): After the successful completion of the course, the student will be able to:

- Understand basics of Internet technology
- Use of HTML in designing static web pages.
- Use of CSS in designing attractive web pages
- Use of Java Script in designing dynamic web pages.
- Students are able design a own website at the end of the course.

Unit	Contents	52 Hrs
UNIT I	**Fundamentals of Web**: Internet, WWW, Web Browsers, Web Protocols and Web Servers, URLs, MIME, HTTP, Security, the Web Programmers Toolbox. **HTML5 and XHTML:** Origins and evolution of HTML5 and XHTML, Basic syntax, Standard XHTML document structure, Basic Text Markup, HTML5 Page Layout and Navigation, Hypertext Links, Lists, Tables, Forms, Frames in HTML5 and XHTML, Syntactic differences between HTML5 and XHTML, Images, audio and video.	12
UNIT II	**Introduction to XML:** Introduction; Syntax; Document structure; Document Type Definitions (DTD); XSLT style sheets; XML Processors; Web services. **Cascading Style Sheets:** Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The Box model, Background images, The and tags.	10
UNIT III	**Introduction to Java Script:** Overview of JavaScript; Object orientation and JavaScript; General syntactic characteristics; Primitives, Operations, and expressions; Screen output and keyboard input; Control statements; Object creation and Modification; Arrays; Functions; Constructor; Pattern matching using expressions; Errors in scripts; Examples.	10
UNIT IV	**Java Script and HTML Documents:** The JavaScript execution environment; The Document Object Model; Element access in JavaScript; Events and event handling; Handling events from the Body elements, Button elements, Text box and Password elements; The DOM 2 event model; The navigator object; DOM tree traversal and modification. Dynamic Documents with JavaScript	10
UNIT V	**Dynamic Documents with JavaScript:** Introduction to dynamic documents; Positioning elements; Moving elements; Element visibility; Changing colors and fonts; Dynamic content; Stacking elements; Locating the mouse cursor; Reacting to a mouse click; Slow movement of elements; Dragging and dropping elements.	10

Pedagogy: Lecture/ PPT/ Videos/ Animations/ Role Plays/ Think-Pair-Share/ Predict-Observe-Explain/ Demonstration/ Concept mapping/ Case Studies examples/ Tutorial/ Activity/ Flipped Classroom/ Jigsaw/ Field based Learning/ Project Based Learning/ Mini Projects/ Hobby Projects/ Forum Theatre/ Dance/ Problem Based Learning/ Game Based Learning/ Group Discussion/ Collaborative Learning/ Experiential Learning / Self Directed Learning etc.

References	
1.	Robert W Sebesta, "Programming the World Wide Web", 4th Edition, Pearson Education, 2008.
2	M.Deitel, P.J.Deitel, A.B.Goldberg, "Internet & World Wide Web How to program", 3rd Edition, Pearson Education / PHI, 2004.
3	Chris Bates, "Web Programming Building Internet Applications", 3rd Edition, Wiley India, 2006.
4	Xue Bai et al, "The Web Warrior Guide to Web Programming", Thomson, 2003
5	Sklar, "The Web Warrior Guide to Web Design Technologies", 1st Edition, Cengage Learning India.
6	Internet and World Wide Web – How to program, Dietel and Nieto, Pearson.

Program Name	**B.Sc.**	Semester	**VI**
Course Title	**Web Technology Lab**		
Course Code:	**DSC 8- Lab**	No. of Credits	**02**
Contact hours	**04 Hours per week**	Duration of SEA/Exam	**3 hours**
Formative Assessment Marks	**25**	Summative Assessment Marks	**25**

Part A

1. Design web pages for your college containing college name and Logo, departments list using href, list tags.
2. Create a class timetable using table tag.
3. Write a HTML code to design Student registrations form for your college Admission
4. Design Web Pages with includes Multi-Media data (Image, Audio, Video, GIFs etc)
5. Create a web page using frame.
6. Write code in HTML to develop a webpage having two frames that divide the webpage intotwo equal rows and then divide the row into equal columns fill each frame with a different background color.
7. Write CSS code to Use Inline CSS to format your ID Card.
8. Using HTML, CSS create display a text called –Hello India !‖ on top of an image of India- Map using an overlay.

Part B

1. JavaScript Program to perform Basic Arithmetic operations.
2. JavaScript Program to implement all string operations.
3. JavaScript Program to Check Prime Number.
4. JavaScript Program to implement Java script Object Concept
5. JavaScript Program to Create Array and inserting Data into Array
6. JavaScript Program to Validate an Email Address.
7. Create a form for Student information. Write JavaScript code to find Total, Average, Result and Grade.
8. Write a program for implementing XML document for Employee Details

Program Name	**B.Sc.**	Semester	**VI**
Course Title	**Statistical Computing & R Programming (Theory)**		
Course Code:	**DSC-8**	No. of Credits	**04**
Contact hours	**52 Hours**	Duration of SEA/Exam	**2 hours**
Formative Assessment Marks	**40**	Summative Assessment Marks	**60**

Course Pre-requisite(s): Basic Knowledge About Programming and Web browsers	
Course Outcomes (COs): After the successful completion of the course, the student will be able to	
CO1	Explore fundamentals of statistical analysis in R environment.
CO2	Describe key terminologies, concepts and techniques employed in Statistical Analysis.
CO3	Define Calculate, Implement Probability and Probability Distributions to solve a wide variety of problems.
CO4	Conduct and interpret a variety of Hypothesis Tests to aid Decision Making.
CO5	Understand, Analyze, and Interpret Correlation Probability and Regression to analyse the underlying relationships between different variables.

Contents	**52 Hrs**
UNIT-I: Introduction to R: Overview and History of R, R Console Input and Evaluation, R Objects and Attributes, Data Types – Vectors, Lists, Matrices, Arrays, Factors, String and Data Frame, Variables and Constants, Variable scope, Getting User Input and Output, R-Operators.	12
UNIT-II: R-Controls Structures: Decision Making in R-if, if-else, if-else ladder, switch statement. Loops in R-repeat, while, for; Loop Control Statements-Break, next; R-functions-Definition, function calls, lazy evaluation of a function, return, multiple returns, recursive functions	10
UNIT-III: Classes and Objects: Class Definition, Creating objects from constructors, Methods and generic functions, Creating Own Methods for S3-class, S4-Class and Reference Class, Error handling in R, Packages in R-Programming.	10
UNIT-IV: Exploratory Data Analysis: R-statistics-Mean, median, mode, variance, standard deviation, Descriptive analysis, Linear Regression, Normal Distribution, Binomial Distribution, Poisson distribution,	10
UNIT-V: Data Analysis and Visualization with R: T-Tests , ANOVA Test, Covariance and Correlation, Hypothesis Testing, Pie-Charts, Bar charts, Boxplots, Histograms, Line Graphs, Scatter Plots	10

Preferred Text Books	
1	Tilman M. Davies, “The book of R: A first course in programming and statistics”, San Francisco, 2016.
2	Vishwas R. Pawgi, “Statistical computing using R software”, Nirali prakashan publisher, e1-edition, 2022.
3	Daniel Bell-R-Programming A step by step guide for absolute guide
4	https://www.youtube.com/watch?v=KlsYCECWEWE https://www.geeksforgeeks.org/r-tutorial/ https://www.tutorialspoint.com/r/index.htm

References	
1	Introductory Statistics with R (Statistics and Computing) Dalgaard, Peter (Author) English (Publication Language) 267 Pages - 02/10/2004 (Publication Date) - Springer (Publisher) Read more at: https://examupdates.in/statistics-with-r-programming-notes/
2	Statistics: An Introduction using R Crawley, Michael J. (Author) English (Publication Language) 342 Pages - 03/11/2005 (Publication Date) - Wiley–Blackwell (Publisher) Read more at: https://examupdates.in/statistics-with-r-programming-notes/
3	A Handbook of Statistical Analyses using R RC Press Hothorn, Torsten (Author) English (Publication Language) 304 Pages - 06/25/2014 (Publication Date) - Chapman and Hall/CRC (Publisher) Read more at: https://examupdates.in/statistics-with-r-programming-notes/
4	A First Course in Statistical Programming with R Braun, W. John (Author) English (Publication Language) 230 Pages - 07/18/2016 (Publication Date) - Cambridge University Press (Publisher) Read more at: https://examupdates.in/statistics-with-r-programming-notes
5	Statistical Analysis with R For Dummies Schmuller, Joseph (Author) English (Publication Language) 464 Pages - 05/16/2017 (Publication Date) - For Dummies (Publisher) Read more at: https://examupdates.in/statistics-with-r-programming-notes

Program Name	**B.Sc.**	Semester	**VI**
Course Title	**R Programming Lab**		
Course Code:	**DSC 8-Lab**	No. of Credits	**02**
Contact hours	**04 Hours per week**	Duration of SEA/Exam	**3 hours**
Formative Assessment Marks	**25**	Summative Assessment Marks	**25**

PART A

1. Write a R program to find Area and Circumference of Circle
2. Write an R program Illustrate with if-else statement and how does it operate on vectors of variable length.
3. Write an R program Illustrate with for loop and stop on condition, to print the error message.
4. Write an R Program to find Factorial of given number.
5. Write an R Program to append a value to given empty vector .
6. Implementation of Vectors data objects operations.
7. Implementation of matrix, arrays and factors objects operations.
8. Write an R Program to Find Mean, Mode & Median.

PART B

1. Write an R Program to implement T-Test.
2. Write an R Program Compute mean values for vector aggregates defined by factors tapply and sapply.
3. Write an R Program to find Unique element of a given string and unique value from vector.
4. Write a R program to demonstrate Binomial Distribution.
5. Write a R program to demonstrate Normal Distribution.
6. Write an R Program Illustrate Reading & Writing Files.
7. Write a R program for simple bar plot for 5 subject marks
8. Implementation of Data visualization using ggplot.

Program Name	**B.Sc.**	Semester	**VI**
Course Title	**Project Work**		
Course Code:	**Proj 1**	No. of Credits	**02**
Contact hours	**02 Hour per week**	Duration of SEA/Exam	**3 hours**

Project Work Guidelines for B. Sc Graduate Programme

1. Students are expected to work out a real life project in some industry/research and development laboratories/educational institutions/software companies, it is suggested that the project is to be chosen which should have some direct relevance in day-to-day activities of the candidates in his/her institution. However, it is not mandatory for a student to work on a real life project. The student can formulate a project problem with the help of Guide.
2. Project mentor/supervisor shall avail work allotment during 6th semester is 2 hour per week and Maximum 20 hours in a semester.
3. The student should submit the final project report to the college through the mentor for completion of the project work.

Formative Assessment for Practical	
Assessment Occasion/ type	**Marks**
Practical Test	20
Journal	05
Total	**25 Marks**
Formative Assessment as per NEP guidelines are compulsory	

Evaluation Scheme for Lab Examination

Assessment Criteria		Marks
Program – 1 from Part A	Writing the Program	03
	Execution and Formatting	07
Program -2 from Part B	Writing the Program	03
	Execution and Formatting	07
Viva Voice		05

	Formative Assessment for Theory	
Components	**Assessment Occasion/ type**	**Marks**
C1	Test	10
	Seminar/Activity	10
C2	Test	10
	Assignment/Projects/Quiz	10
	Total	**40 Marks**

Summative Assessment for Theory

Semester End Exam Question Paper Pattern

Duration of the examination: 2hour **Max. Marks:60**

Section A

Answer any TEN from the following, each carries 2 marks: **[10X2=20]**

1. ----------
2. ----------
3. ----------
4. ----------
5. ----------
6. ----------
7. ----------
8. ----------
9. ----------
10. ----------
11. ----------
12. ----------

Section B

Answer any FOUR from the following questions each carries 5 marks.

[4X5=20]

13. ----------
14. ----------
15. ----------
16. ----------
17. ----------

Section C

Answer any two from the following questions each carries 10 marks.

(The Question may have consist two sub-questions)

[2X10=20]

18. ----------
19. ----------
20. ----------