SKE Society's
G.S.S. College(Autonomous)
Belagavi

STRUCTURE & SYLLABUS OF BACHELOR OF SCIENCE

(B.Sc COMPUTER SCIENCE)

1ST TO 2ND Semesters

w.e.f.

Academic Year 2024-25 and Onwards

Submitted by

Chairman,
Board of Studies (UG),
Bachelor of Computer Science
G.S.S. College(Autonomous)Belagavi

Curriculum Structure for B.Sc. Computer Science of G.S.S. College w.e.f 2024-25								
Category	**Course Code**	**Title of the Paper**	**Marks**			**Hours / Week**	**Credits**	**Duration of Exams (Hrs.)**
			IA	**SEE**	**Total**			
SEMESTER I								
DSC 1	SEPBSCCST01	Fundamentals of Computer with Programming in C	20	80	100	04	3	03
	SEPBSCCSP01	C Programming Lab	10	40	50	04	2	03
TOTAL: HOURS / CREDIT					**150**	**08**	**5**	
SEMESTER II								
DSC 2	SEPBSCCST02	Data Structures using C	20	80	100	04	3	03
	SEPBSCCSP02	Data Structures using C Lab	10	40	50	04	2	03
TOTAL: HOURS / CREDIT					**150**	**08**	**5**	

Year	I	Course Code: SEPBSCCST01	Credits	03
Semester	I	Course Title: Fundamentals of Computer with Programming in C	Hours	48

Course Pre-requisites, if any	NA	
Formative Assessment Marks: 20	Summative Assessment Marks: 80	Duration of ESA: 03hrs.

Course Outcomes	***After completing this course satisfactorily, a student will be able to:*** ○ Confidently operate Laptop/Desktop Computers to carry out computational tasks. ○ Understand working of Hardware and Software and the importance of operating systems. ○ Understanding the concept of programming languages. ○ Read, understand and trace the execution of programs written in C language. ○ Write the C code for any complex and real-time problems. ○ Write programs that perform operations matrices using Arrays & Strings. ○ Write programs using pointer concepts for efficient memory usage. ○ Create user defined functions to strengthen the C Library.

Unit No.	Course Content	Hours
UNIT I	**Fundamentals of Computers**: Introduction, Definition, History of Computers, Generations of Computers, Classification of Computers, Characteristics of Computers, Basic Organization of a Computer, Software, Types of Software, Overview of Operating System, introduction to Linux, GPU commands. **Introduction to C Programming**: History of C, Features of C; Programming Languages and its Classification; Compiler, Interpreter, Linker, and Loader; Problem Solving Techniques; Flowcharts; Algorithms; Basic Structure of a C Program; Executing a C Program; Examples of flow charts and algorithms - Largest of three numbers, Reversing the digits of an integer, GCD of two integers, generating prime numbers, computing n^{th} Fibonacci numbers.	10
UNIT II	**Overview of C:** C Character set; C Tokens - keywords, identifiers, constants, and variables; Data types; Declaration & initialization of variables; Declaration of Storage Class; Assigning Values to Variables, Symbolic constant, Declaring a Variable as Constant, binary representation of Integer **Operators & Expression:** Introduction; Arithmetic Operators; Relational Operators; Logical Operators; Assignment Operators; Increment & Decrement Operators; Conditional Operator; Bitwise Operators; Special Operators; Arithmetic Expressions; Evaluation of Expressions; Type Conversion; Operator Precedence and Associatively; Standard Mathematical functions.	10

UNIT III	**Input/output Functions**: Formatted I/O Functions - printf and scanf; Unformatted I/O Functions – getch(), putch(), getche(), getchar(), putchar(), gets(), puts(). **Control Structures:** **Decision Making & Branching:** Introduction; Decision making with Simple if; if-else; Nesting of if-else; else-if ladder, switch; and goto statement **Decision Making & Looping:** Introduction; while loop; do-while loop; for loop, Nested loops; jumps in loops - break, continue, return, exit.	10

UNIT IV	**Arrays**: Introduction; One-dimensional Arrays; Declaration of One-dimensional Arrays; Initialization of One-dimensional Arrays; Two-dimensional Arrays; Initializing Two-dimensional Arrays; Multi-dimensional Arrays; Dynamic Arrays; Passing Arrays to Functions. **Strings:** Introduction; Declaring & Initializing String Variables; Reading Strings from Terminal; Writing Strings to Screen; String Handling Functions - strlen, strcmp, strcpy, strrev and strcat; Character handling functions - toascii, toupper, tolower, isalpha, isnumeric etc.	08
UNIT V	**Pointers in C:** Introduction; Understanding Pointers; Accessing the Address of a Variable; Declaring Pointer Variables; Initialization of Pointer Variables; Accessing a Variable through its Pointer; **User defined functions:** Introduction; Need for user defined functions; Elements of User-Defined Functions; Definition, Categories of Functions - No Arguments and No Return Values, Arguments but No Return Values, Arguments with Return Values, No Arguments but Returns a Value; Nesting of Functions; Recursion; The Scope, Visibility and Lifetime of Variables. **User defined structure:** Defining a Structure; Declaring Structure Variables; Accessing Structure Members; Structure Initialization; Arrays of Structures; Unions; Difference between Structure & Union; Bit Fields.	10

Recommended Leaning Resources

Text Books:

1. Computer Concepts and Programming in C (UPTU): E. Balaguruswamy.
2. Computer Fundamentals and Programming in c, "Reema Thareja", Oxford University, Second edition, 2017.
3. Brian W. Kernighan and Dennis M. Ritchie - The „C" Programming Language, Prentice Hall of India.

Reference Books:

1. Pradeep K. Sinha and Priti Sinha: Computer Fundamentals (SixthEdition), BPB Publication
2. Kamthane: Programming with ANSI and TURBO C (Pearson Education)
3. Kernighan & Ritche: The C Programming Language(PHI)
4. Yashwant Kanitkar: Let us C

Year	I	Course Code: SEPBSCCSP01	Credits	02
Semester	I	Course Title: C Programming Lab	Hours	48

Course Pre-requisites, if any	Knowledge of C Programming Language	
Formative Assessment Marks: 10	Summative Assessment Marks: 40	Duration of ESA: 03hrs.

PRACTICE LABS

- The following activities be carried out/ discussed in the lab during the initial period of the semester.
  - Basic Computer Proficiency
  - Familiarization of Computer Hardware and Software.
  - Basic Computer Operations and Maintenance.
  - Do"s and Don"ts, Safety Guidelines in Computer Lab
- Familiarization of Basic Software – Operating System, Word Processors, Internet Browsers, Integrated Development Environment (IDE) with Examples.
- 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 read a number, find the sum of the digits, reverse the number andcheck it for palindrome
6. Write a C Program to display the following by reading the number of rows as input,

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

7. An electricity board charges the following rates for the use of electricity: for the first 200 units 80 paise per unit: for the next 100 units 90 paise per unit: beyond 300 units Rs 1 per unit. All users are charged a minimum of Rs.100 as meter charge. If the total amount is more than Rs 400, then an additional surcharge of 15% of total amount is charged. Write a program to read the name of the user, number of units consumed and print out the charges.
8. Write a C Program to find the roots of quadratic equation (demonstration of switch-case statement)
9. Write a C program to read marks scored by n students and find the average of marks (Demonstration of single dimensional array)
10. Program to perform addition and subtraction of Matrices

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

Instructions:

1. Certified Journal is mandatory for every student to appear the practical examination.
2. Student has to execute a minimum of 8 programs from each part to complete the Lab course.
3. Based on practical internal test of 10 marks shall be awarded.

Year	II	Course Code: SEPBSCCST02	Credits	03
Semester	II	Course Title: Data Structure Using C	Hours	48

Course Pre-requisites, if any	Knowledge of C Programming.	
Formative Assessment Marks: 20	Summative Assessment Marks: 80	Duration of ESA: 03hrs.

Course Objectives	After completing this course satisfactorily, a student will be able to: ○ To explain Fundamentals of Data Structures and their Applications. ○ To illustrate representation of Different Data Structures such as Arrays, Stack, Queues, Linked Lists, Trees and Graphs. ○ To Design and Develop Solutions to problems using Linear Data Structures ○ To discuss applications of Nonlinear Data Structures in problem solving. ○ Problem solving using Dynamic memory allocation concepts. ○ To introduce advanced Data structure concepts such as Binary Search Trees

Unit No.	Course Content	Hours
UNIT I	**Introduction to Data Structures**: Definition; Classification of Data Structures - Primitive & Non-primitive, Linear and Non-linear; Operations on Data Structures; Abstract Data Type; Different Approaches to Designing an Algorithm; Control Structures Used in Algorithms; Time and Space Complexity. **Arrays:** Storing Values in Arrays; Operations on arrays; Types of arrays; Arrays as Abstract Data Types (ADT);	10
UNIT II	Representation of Linear Arrays in memory; Traversing Linear Arrays; Inserting and Deleting elements; Sorting – Bubble sort, Selection sort, Merge Sort, Insertion sort, Searching - Sequential Search, Binary search; Iterative and Recursive searching; **Stacks:** Introduction; Definition and Array Representation of Stacks; Operations on Stacks; Applications of Stacks; Infix, Postfix and Prefix notations; Conversion from infix to postfix using stack; Evaluation of postfix expression using stack;	9
UNIT III	**Queues:** Introduction; Definition and Array Representation of queues; Types of Queues – Simple Queues, Circular Queues, Double ended Queues, Priority Queues; Operations on Ordinary queue; **Recursion:** Definition; Types of recursions; Recursion Technique Examples - Fibonacci numbers, GCD, Binomial coefficient ^{n}Cr, Towers of Hanoi; Comparison between Iterative and Recursive functions.	9

UNIT IV	**Dynamic Memory Allocation:** Static & Dynamic Memory Allocation; Memory allocation and De-allocation functions - malloc, calloc, realloc and free. **Linked List:** Introduction - Basic Terminologies, Linked Lists versus Arrays, Memory Allocation and De-allocation for a Linked List; Representation of Linked List, Types of Linked Lists - Singly Linked List, Doubly Linked List, Circular Linked List, Circular Doubly Linked List and Header Linked List; Representation of Linked list in Memory; Operations on Singly Linked Lists – Traversing, Searching, Inserting, Deleting;	10
UNIT V	**Trees:** Definition; Tree Terminologies – Node, Root Node, Leaf Node, Path, Ancestors of a Node, Siblings, Descendant node, Terminal & Non-Terminal Nodes, Degree of a Node, Level Number, In-degree, Out-degree; Types of Trees; **Binary tree:** Terminology; Types of Binary Trees - Strict Binary Tree, Complete Binary Tree, Binary Search Tree and Heap Tree; Representation of Binary Tree in Memory. Traversal Of Binary Tree – Pre-order, In-order and Post-order Traversal;	10

Recommended Leaning Resources

Text Book:

1. Data Structures Using C - Reema Thareja, 2nd Edition, Published in India by Oxford University Press.
2. Data Structures Using C - Samir Kumar Bandyopadhyay, Kashi Nath Dey, Pearson Education.
3. Data Structures with C – Seymour Lipschutz, Tata McGraw-Hill Education Private Limited.
4. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014

Reference Books:

1. Padma Reddy: Data Structure Using C, 2019, CENGAGE INDIA
2. Tanenbaum: Data structures using C (Pearson Education)
3. Ellis Horowitz and SartajSahni: Fundamentals of Data Structures
4. Kamathane: Introduction to Data structures (Pearson Education)
5. Y. Kanitkar: Data Structures Using C(BPB)
6. Kottur: Data Structure Using C
7. Sudipa Mukherjee: Data Structures using C – 1000 Problems and Solutions (McGraw Hill Education,2007)

Year	II	Course Code: SEPBSCCSP02	Credits	02
Semester	II	Course Title: Data Structures Using C Lab	Hours	48
Course Pre-requisites, if any		Knowledge of C Programming		
Formative Assessment Marks: 20		Summative Assessment Marks: 80	Duration of ESA: 03hrs.	

PART A:

1. Write a C program to read one dimensional array, and print sum of all elements along with inputted array elements using Dynamic Memory Allocation.
2. Write a C Program to Display Pascal Triangle using Binomial function.
3. Write a C Program to find GCD using recursive function.
4. Write a C Program to generate n Fibonacci numbers using recursive function.
5. Write a C Program to implement Towers of Hanoi.
6. Write a C Program to read the names of cities and arrange them alphabetically.
7. Write a C Program to implement dynamic array, find smallest and largest element of the array.
8. Write a C Program to search an element using linear search technique.
9. Write a C Program to search an element using recursive binary search technique.
10. Write a C Program to sort the given list using selection sort technique.

PART B:

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

Instructions:

1. Certified Journal is mandatory for every student to appear for the examination.
2. Student has to execute a minimum of 8 programs in each part to complete the Lab course.
3. Based on practical test 10 marks of practical shall be awarded.

Semester End Exam Question Paper Pattern

Duration of the examination: 3hour **Max. Marks: 80**

Section A

Answer any TEN Questions 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 FOUR from the following questions each carries 10 marks.

(The Question may consist two sub-questions)

18. ---------- **[4X10=40]**
19. ----------
20. ----------
21. ----------
22. ----------

CIA for Practical	
Assessment Type	**Marks**
Test1	10
Total	**10 Marks**

Instructions:

1. Based on Two practical Tests 10 marks of practical shall be awarded.

Note: *Guidelines given by the university from time-to-time shall be followed for IA.*

SEE Scheme of Evaluation for Lab Examination

Assessment Criteria	Marks
Writing of 2 Programs (Each from Part A & Part B)	15
Execution (Includes program code modification and execution result)	15
Journal	05
Viva Voice	05
Total	**40 Marks**

Instructions:

1. Certified Journal is mandatory for appearing the examination.
2. Students shall be given two programming assignments taking into consideration of duration of the time allotted to students for writing, typing and executing the programs.

CIA for Theory	
Assessment Type	**Marks**
Test 1 / Seminar / Activity	10
Test 2 / Seminar / Activity	10
Total	**20 Marks**