BAGALKOT UNIVERSITY

Mudhol Road, Jamkhandi-587301 Dist: Bagalkot

PROGRAM /COURSE STRUCTURE AND SYLLABUS
FOR
COMPUTER SCIENCE

as per the Choice Based Credit System (CBCS) designed in accordance with Learning Outcomes-Based Curriculum Framework (LOCF)

For

Bachelor of Arts
Computer Science

(General Degree)
I and II Semester

w.e.f.

Academic Year 2024-25

Preamble for UG Syllabus of Bagalkot University

Bagalkot University Jamkhandi has been established by the Government of Karnataka and has started functioning from the academic year 2023-24. All the degree colleges other than engineering and medical colleges in the district of Bagalkote, are affiliated to this university as per the Karnataka State Universities Act 2000, as modified by the 26th Act of 2022. The students taking admission to any of the colleges in the district of Bagalkote, from the academic year 2023-24 will be students of Bagalkot University. The Government of Karnataka has instructed all the Universities to revise the under graduate syllabus as per the Government order no. ED 166 UNE 2023 Bengaluru Dated 08-05- 2024 from the academic year 2024-25.

Hence the Bagalkot University has revised the syllabus as suggested by its Board of Studies and approved by Academic Council and Syndicate. The subject code format for all the subjects of the new syllabus is also revised.

The subject code format is described in the following.

Subject Code Format

1	2	3	4	5	6	7	8	9	10
VER	DEGREE	SEM	DISCIPLINE			SUB. TYPE	SL. NO. FOR SUB. TYPE		TH/LAB/F
2	A	1	C	H	E	M	0	1	T
2	B	1	P	O	L	M	0	1	T

[1] **The Version information gives the version of the syllabus. It can take values 1,2..9,a,b,...**

[2] **The UG degree codes to be provided as / The code applicable to all degrees**

Sl. No	Degree Code		Degree
1	B.Sc.	A	Bachelor of Science
2	B.A	B	Bachelor of Arts
3	B.Com.	C	Bachelor of Commerce
4	BBA	D	Bachelor of Business Administration
5	BCA	E	Bachelor of Computer Applications
6	BSW	F	Bachelor of Social Work
7.	-----------	S	Applicable to all degrees

[1] The Semester Information is provided as

Sl. No	Semester
1	1
2	2
3	3
....	

4-6]The Discipline Information to be provided as

Sl No	Degree	Discipline Code
1	B.Com.	XXX
2	BCA	XXX
3	BBA	XXX
4	BSW	XXX
5	B.A	'HIS', POL','GEO','KAN', 'HIN' etc. The detailed list is to be provided
6	B.Sc.	'PHY','CHE', 'BOT','ELE' etc. The detailed List is to be Provided

[7] The Subject Type to be provided as

Sl. No.	TYPE	Description
1	Major	M
2	Language	L
3	Constitutional Moral Values	C
4.	Elective	E
5.	Skill / Practical based learning	S
6.	Mini Project	P
7.	Internship	I
8.	Case study/ Survey using principles of Research methodology	R

[8-9] The Running Serial Number is to be provided for a particular Subject type 01 to 99

[10] This character specifies the category of the subject namely, T=Theory, L-Practical, P-Project Work, F-Field work, Viva-V, I-Internship, Dissertation-D

Computer Science I & II Semester Syllabus and 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. In India, 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, B.Sc and M.Sc programmes with specialization in Computer Science were introduced to train manpower in this highly demanding area.

The degree is aimed at undergraduate level training facilitating multiple career paths. Students so graduated, can take up postgraduate programmes in 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 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 studentslearn 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 tochanges in programming languages and learn new languages as they are developed. The present Curriculum Framework for BSc degrees is intended to facilitate thestudents 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 analyze 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 knowledgeon 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 program enables students to attain following additional attributes besides the afore-mentioned attributes, by the time ofgraduation:

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.

1. Apply standard Software Engineering practices and strategies in real -time software project development

Syllabus and Credits Structure under Choice Based Credit System [CBCS] *General Degree*

for the Three Years B.Sc. with Computer Science Undergraduate Programme with effect from 2024-25

Category	Course code	Title of the Paper	Marks			Teaching hours/week			Credit	Duration of exams (Hrs)	Teaching Department
			IA	SEE	Total	L	T	P			
L1	------	Language 1	20	80	100	3	-	-	3	3	-------------
L2	------	Language 2	20	80	100	3	-	-	3	3	-------------
Major	2B1COMM01T	Computer Fundamentals and Programming in C	20	80	100	3	-	-	3	3	Computer Science
	2B1COMM01L	C Programming Lab	10	40	50	-	-	4	2	3	Computer Science
Major	------	Theory Course Title	20	80	100	3	-	-	3	3	-------------
	------	Lab Course Title	10	40	50	-	-	4	2	3	-------------
Major	------	Theory Course Title	20	80	100	3	-	-	3	3	-------------
	------	Lab Course Title	10	40	50	-	-	4	2	3	-------------
Common	2S1XXXC01T	Constitutional Values	10	40	50	1	-	2	2	2	Constitutional Values: • Political Science
	2S1XXXC02T	Environmental Studies									Environmental Studies: • Chemistry/Geology/Geography/ Botany
			Total Marks		**700**	**Semester Credits**			**23**		

Category	Course code	Title of the Paper	Marks			Teaching hours/week			Credit	Duration of exams (Hrs)	Teaching Department
			IA	SEE	Total	L	T	P			
L3	----	Language 3	20	80	100	3	-	-	3	3	-----------
L4	----	Language 4	20	80	100	3	-	-	3	3	-----------
Major	2B2COMM02T	Data Structures using C	20	80	100	3	-	-	3	3	Computer Science
	2B2COMM02L	Data structures Lab	10	40	50	-	-	4	2	3	Computer Science
Major	----	Theory Course Title	20	80	100	3	-	-	3	3	-----------
	----	Lab Course Title	10	40	50	-	-	4	2	3	-----------
Major	----	Theory Course Title	20	80	100	3	-	-	3	3	-----------
	----	Lab Course Title	10	40	50	-	-	4	2	3	-----------
Common	2S1XXXC01T	Constitutional Values	10	40	50	1	-	2	2	2	Constitutional Values: • Political Science
	2S1XXXC02T	Environmental Studies									Environmental Studies: • Chemistry/Geology/Geography/ Botany
		Total Marks			700	Semester Credits			23		

Year	I	**Course Code:** 2B1COMM01T	**Credits**	03
Sem.	1	**Course Title:** Computer Fundamentals and Programming in C	**Hours**	42
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 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, Generations of Computers, Types of Computers, Basic Organization of a Digital Computer; Number Systems – Different types, Conversion from one integer 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 Application Software ;Operating System – Functions of O.S and Types of O.S ; Computer Languages - Machine Level, Assembly Level & High Level Languages, Translator Programs –Assembler, Interpreter and Compiler; Planning a Computer Program - Algorithm, Flowchart with Examples.	10
Unit II	**Introduction to C Programming**: Introduction, History and Features of C; Characteristics of C, Structure of C Program with Examples ; Creating and Executing a C Program; **C Programming Basic Concepts:** C Character Set, Data types, C tokens - keywords, identifiers, constants and variables Declaration& initialization of variables ; Symbolic constants. **Input and output with C**: Formatted I/O functions – printf() and scanf(),control strings and escape sequences, output specifications with printf();Unformatted I/O functions, to read and display single character and a string- getchar(), putchar(), gets() and puts() **Operators & Expressions**: Arithmetic operators; Relational operators; Logical operators; Assignment operators; Increment & Decrement operators; Bitwise operators; Conditional operator; Special operators; Operator Precedence and Associativity, Evaluation of arithmetic expressions; Type conversion.	11

Unit III	**Control Structures**: Decision making Statements - Simple if, if_else, nested if_else, else_if ladder, switch case, goto, break & continue statements; Looping Statements - Entry controlled and Exit controlled statements, while, do while, for loops, Nested loops with examples. **Arrays**: One Dimensional arrays - Declaration, Initialization and Memory representation; Two Dimensional arrays - Declaration, Initialization and Memory representation with examples.	11
Unit IV	**Strings**: Declaring & Initializing string variables; String handling functions – strlen(), strcmp(), strcpy() and strcat(); Character handling functions – toascii(), toupper(), tolower(), isalpha(), isnumeric() etc. **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, declaration & initialization, accessing union members, difference between Structures and Unions.	10
Recommended Learning Resources		

1. Pradeep K. Sinha and Priti Sinha: Computer Fundamentals(SixthEdition),BPB Publication
2. E. Balgurusamy: Programming in ANSIC (TMH)
3. Kamthane :Programming with ANSI and TURBO C(Pearson Education)
4. V. Rajaraman: Programming in C (PHI–EEE)
5. S. Byron Gottfried: Programming with C(TMH)
6. Kernighan & Ritche: The C Programming Language(PHI)
7. Yashwant Kanitkar: Let us C

Year	I	**Course Code:** 2B1COMM01L	**Credits**	02
Sem.	I	**Course Title:** C Programming Lab	**Hours**	50

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

	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 command to create a directory, change directory and 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 whether the given number is prime or not . 5. Write a C Program to generate n prime numbers. 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 two numbers and perform arithmetic operations usingswitch case. 9. Write a C program to read marks scored by n students and find the average of marks (Demonstration of single dimensional array. 10. Write a C Program to remove Duplicate Element in a single dimensional Array. 11. 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. 3. Write a C Program to check a number for prime by defining isprime() function. 4. Write a C Program to read, display and to find the trace of a square matrix. 5. Write a C Program to read, display and multiply two m x n matrices using functions. 6. Write a C Program to read a string and to find the number of alphabets, digits, vowels, consonants, spaces and special characters.

	7. Write a C Program to Reverse a String. 8. Write a C Program to Swap Two Numbers. 9. Write a C Program to demonstrate student structure to read & display records of n students. 10. Write a C Program to demonstrate the difference between structure & union. 11. Write a C Program to find the roots of quadratic equation (demonstration of else if ladder).

Year	I	**Course Code:** 2B2COMM02T		**Credits**	03
Sem.	2	**Course Title:** Data Structures using C		**Hours**	42
Course Pre-requisites, if any		NA			
Formative Assessment Marks: 20		SummativeAssessmentMarks:80		Duration : 3hrs.	

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 by algorithms • Describe common applications for arrays, records, linked structures, stacks, queues, trees, and graphs • Write programs that use arrays, records, linked structures, stacks, queues, trees, and graphs • Demonstrate different methods for traversing trees • Compare alternative implementations of data structures with respect to performance • Describe the concept of recursion, give examples of its use • Discuss the computational efficiency of the principal algorithms for sorting and searching

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 **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; Dynamic memory allocation: Static & Dynamic memory allocation; Memory allocation and de- allocation functions – malloc(), calloc(), realloc() and free().	10
Unit-II	**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. **Sorting:–** Selection sort, Bubble sort, Merge Sort, Quick sort, Selection sort, Insertion sort; **Searching -** Sequential Search, Binary search; Iterative and Recursive searching;	10

Unit III	**Stacks :**Basic Concepts Definition and 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; **Queues:** Basic Concepts–Definition and Representation of queues; Types of queues-Simple queues, Circular queues, Double ended queues,Priority queues; Operations on Simple queues; **Linked list:** Basic Concepts – Definition and Representation of linked list, Types of linked lists – Singly linked list, Doubly linked list, Circular linked list; Operations on Singly linked lists – Traversing, Searching, Insertion, Deletion;	12
Unit IV	**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, Traversal of binary tree; preorder, in order and Post order traversal; binary search tree and heap tree;	10
	Recommended Leaning Resources	

Reference Books:
1. Ellis Horowitz and Sartaj Sahni: 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. Padma Reddy: Data Structure Using C
6. Sudipa Mukherjee: Data Structures using C – 1000 Problems and Solutions (McGraw Hill Education,2007)

Year	I	Course Code: 2B2COMM02L	Credits	02
Sem.	II		Hours	50
		Course Title: Data Structure Lab		

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

Part A:
1. Write a C Program to find GCD using recursive function
2. Write a C Program to generate n Fibonacci numbers using recursive function.
3. Write a C Program to implement Towers of Hanoi.
4. Write a C Program to implement dynamic array, find smallest and largest element of the array.
5. Write a C Program to read the names of cities and arrange them alphabetically.
6. Write a C Program to sort the give n list using selection sort technique.
7. Write a C Program to sort the given list using bubble sort technique.
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.

PARTB:

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 implement Stack.
5. Write a C Program to convert an infix expression to postfix.
6. Write a C Program to implement simple queue.
7. Write a C Program to implement linear linked list.
8. Write a C Program to display traversal of a tree.
9. Write a C Program to construct a binary search tree

ASSESSMENT METHODS

Formative Assessment for Theory

Evaluation Scheme for Internal Assessment: Continuous Internal Assessment (CIA)

Assessment Criteria 20 marks		
1st Internal Assessment Test for 20 marks of 1 hour duration after 8 weeks and later marks should be reduced to 5	CIA : C1	5 Marks
2nd Internal Assessment Test for 40 marks 2 hours duration after 15 weeks and marks should be reduced to 10	CIA : C2	10 Marks
Assignment/ Activity	CIA : C3	05 Marks
	Total	20 Marks

Summative Assessment for Theory:

SEMESTER END EXAM : SEE	C4	80 Marks

Formative Assessment for Practical:

Assessment Criteria 10 marks		
Internal Test including basic understanding of the concept, Viva Voce, Journal. Test should be conducted for 50 marks and later it should be reduced for 10 marks	CIA : C1	10 Marks

Summative Assessment for Practical:

SEMESTER END EXAM : SEE	C2	40 Marks

Instructions to set the question paper and question paper pattern :

Instruction to set the question paper.

1. Question number 1 has 12 sub questions consisting of 3 questions from each unit. Each question carries two marks. Student has to answer any ten questions.

2. Question number 2 to 7 are from unit I to IV.
   Each question carries five marks. Student has to answer any four questions

3. Question number 8 to 12 are from unit I to IV.
   Each question carries ten marks. Student has to answer any four questions

Question Paper pattern

B.Sc. Degree Examination COMPUTER SCIENCE

Time: 3 hours   Max. Marks: 80

		Part- A	
1.		**Answer any <u>TEN</u> questions**	**10 x 2 = 20**
	a)		
	b)		
	c)		
	d)		
	e)		
	f)		
	g)		
	h)		
	i)		
	j)		
	k)		
	l)		
		Part-B	
		Answer any <u>Four</u> questions	**4 x 5 = 20**
	2		
	3		
	4		
	5		
	6		
	7		
		Part-C	
		Answer any FOUR questions	**4 X 10 = 40**
	8		
	9		
	10		
	11		
	12		