

PROGRAM STRUCTURE, CREDITS, AND SYLLABUS

Of

BACHELOR IN COMPUTER APPLICATIONS (BCA)

AS PER AICTE MODEL CURRICULUM-2024

(NATIONAL EDUCATION POLICY – 2020)

(To be effective from the session 2025-2026)

CHAUDHARY CHARAN SINGH UNIVERSITY, MEERUT



CHAUDHARY CHARAN SINGH UNIVERSITY MEERUT

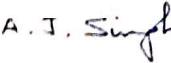

(Mukesh Kumar Sharma)


(Bhupendra Kumar)

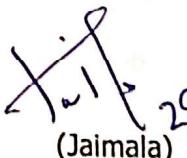

(M. P. Thapliyal)


(M.N. Hoda)


(T. V. Vijaykumar)


(A.J. Singh)


(Sunil Pandey)


29-04-2025
(Jaimala)

14. Semester-wise Structure, Courses, and Credits

Note: In Course Code 'T' denotes theory Paper Code, 'P' denotes practical Paper Code, and 'R' denotes Internship/Project/Dissertation Paper Code.

TABLE-VI

14.1. SEMESTER-I

Sl. No.	Course Type	Course Code	Course Title	(L+T) TH	P	Credit
1	CC-I	BCA 1001T	Mathematical Foundation for Computer Science – I	3	0	3
2	CC-II	BCA 1002T BCA 1002P	Computer Architecture	3	4	5
3	GEC-I	BCA 1003 T	Indian Knowledge System	2	0	2
4	SEC-I	BCA 1004T BCA 1004P	Problem Solving Techniques	3	4	5
5	AEC-I%	BCA 1005 T	General English – I	2	0	2
6	VAC-I	BCA 1006 T	Environmental Science and Sustainability	2	0	2
TOTAL						19

Note: AEC% : Alternative NPTEL/SWAYAM Course

TABLE-VII

14.2 SEMESTER-II

Sl. No.	Course Type	Course Code	Course Title	(L+T) TH	P	Credit
1	CC-III	BCA 2001 T	Mathematical Foundation for Computer Science – II	3	0	3
2	CC-IV	BCA 2002T BCA 2002P	Data Structures	3	4	5
3	CC-V	BCA 2003T BCA 2003P	Operating Systems	3	2	4
4	SEC-II	BCA 2004T BCA 2004P	Object Oriented Programming Using Java	3	4	5
5	SEC-III	BCA 2005T BCA 2005P	Web Technologies	1	2	2
6	VAC-II	BCA 2006 T	Indian Constitution	2	0	2
TOTAL						21

14.2.1 To exit with Certificate in Computer Applications, additional 4 credits through a field relevant Skill-Based Course/ Work-Based Vocational Course/ Social Responsibility & Community Engagement/ Internship/ Apprenticeship of a minimum duration of 8 weeks/ 120 hours in the summer break after the second semester are mandatory. (Code - BCA 2007R)

6. Reshma Ramdhoni, Ancient Indian Culture and Civilisation, Star Publication, 2018
7. Supriya Lakshmi Mishra, Culture and History of Ancient India (With Special Reference of Sudras), 2020.
8. Gambirananda, Swami, Tr. Upanishads with the Commentary of Sankaracharya, Kolkata: Advaita Ashram publication Department, 2002.
9. Ranganathananda, Swami. The Message of the Upanishads. Bombay: Bharatya Vidya Bhawan, 1985.
10. Om Prakash, Religion and Society in Ancient India, Bhartiya Vidhya Prakashan, 1985
11. J. Auboyer, Daily Life in Ancient India from Approximately 200 BC to AD 700, Munshi Ram Manoharlal Publication, 1994.
12. DK Chakrabarty, Makkhan Lal, History of Ancient India (Set of 5 Volumes), Aryan Book International publication, 2014
13. Dr. Girish Nath Jha, Dr. Umesh Kumar Singh and Diwakar Mishra, Science and Technology in Ancient Indian Texts, DK Print World limited,
14. Swami BB Vishnu, Vedic Science and History - Ancient Indian's Contribution to the Modern World, goswami publication, 2015
15. Chatterjee, S.C. The Nyaya Theory of Knowledge. Calcutta: University of Calcutta Press, 1950.
16. Dasgupta, Surendra. A History of Indian Philosophy. Motilal Banarsi Dass Publishing House, 1991. Vols. III & IV.
17. Mercier, Jean L. From the Upanishads to Aurobindo. Bangalore: Asian Trading Corporation, 2001.
18. M. Hiriyanna. Essentials of Indian Philosophy. London: Diane Publications, 1985.
19. Hume, Robert Ernest, Tr. The Thirteen Principal Upanishads. Virginia: Oxford.
20. Radhakrishnan, S. Principal Upanishads. New York: Harper Collins, 1963.
21. Satprakashananda. The Methods of Knowledge according to Advaita Vedanta. Calcutta: Advaita Ashram, 2005.
22. Potter, K.H. Encyclopaedia of Indian Philosophies, Vol. III. Delhi: Motilal Banarsi Dass Publishing House 2015.

Code: BCA-1004T (For Theory) BCA-1004P (For Practical)	SEC-I	Problem Solving Techniques	3L+T:4P	5 Credits (45 hours theory and 60 hours practical)
---	--------------	-----------------------------------	----------------	---

Max Marks; Theory: 100 (Int: 25; Ext: 75); Practical: 100

Course Outcomes: Upon completion of the course, students will be able to

- CO1:** Understand basic terminology of computers, problem solving, programming languages and their evaluation
- CO2:** Develop algorithms, flowcharts and pseudo code to solve computational problems using structured approaches.
- CO3:** Implement structured programming concepts and control structures using the C language.
- CO4:** Solve numerical and statistical problems using control structures and C programming.
- CO5:** Apply modular programming, recursion, and array/matrix operations in C programs.

Unit	Topics	Purposed lectures
I	Problem-Solving and Algorithm Development: Problems And Problem Instances, Generalization and Special Cases, Types of Computational Problems, Classification of Problems, Analysis of Problems, Solution Approaches, Algorithm Development, Analysis of Algorithm, Efficiency, Correctness, Role of Data Structures in Problem Solving, Problem-Solving Steps (Understand the Problem, Plan, Execute, And Review), Breaking the Problem into Subproblems, Input/output Specification, Input Validation, Pre and Post Conditions.	11
II	Structured Programming Concepts: Sequence (Input/Output/Assignment); Selection (If, If-Else) And Repetition For, While, Do-While) Statements, Control Structure Stacking and Nesting. Different Kinds of Repetitions: Entry Controlled, Exit Controlled, Counter Controlled, Definite, Indefinite and Sentinel-Controlled repetitions. Pseudocode and Flowcharts, Definition And Characteristics of algorithms,	11

	<p>Standard Algorithm Format.</p> <p>Problems Involving Iteration and Nesting: Displaying Different Patterns and Shapes Using Symbols and Numbers, Generating Arithmetic and Geometric Progression, Fibonacci and Other Sequences, Approximate Values For n, $\sin(x)$, $\cos(x)$, Etc. Using Taylor Series.</p> <p>Different Kinds of Data in The Real World and How They are Represented in The Computer Memory. Representation of Integers: Signed Magnitude Form, 1's Complement And 2's Complement. Representation of Real Numbers: IEEE 754 Floating Point Representation. Representation of Characters: ASCII, UNICODE.</p> <p>C Language and Basic Programming Constructs:</p> <p>Introduction To Programming Languages, Different Generations of Programming Languages. Typed Vs Typeless Programming Languages, History of C Language, An Empty C Program. C Language Counterparts For Input (<code>scanf()</code>), Output (<code>printf()</code>) Statements, Assignment, Arithmetic, Relational and Logical Operators. If, If-Else Statements, For, While, Do-While Statements. Data Types. Translating Pseudocode/Algorithm to C Program. Incremental Compilation and Testing of The C Program. Simple Problems Involving Input, Output, Assignment Statement, Selection and Repetition. Good Coding Practices.</p>																					
III	<p>Problems on Numbers and Basic Statistical Operations:</p> <p>Extracting Digits of a Number (Left to Right and Right to Left), Palindrome, Prime Number, Prime Factors, Amicable Number, Perfect Number, Armstrong Number, Factorial, Converting Number from One Base to Another. Statistics (Maximum, Minimum, Sum and Average) on a Sequence of Numbers which are Read using Sentinel Controlled Repetition using only a few Variables. C Language: else-if Ladder, switch Case, Increment/Decrement Operators, break and continue Statements.</p>	11																				
IV	<p>Modular Programming and Arrays :</p> <p>Modular Programming, Top-Down and Bottom-Up Approaches to Problem Solving. Recursion.</p> <p>Problems on Arrays: Reading and Writing of Array Elements, Maximum, Minimum, Sum, Average, Median and Mode. Sequential And Binary Search. Any one Sorting Algorithm. Matrix Operations.</p> <p>Implementation in C Language: Function Definition and Declaration (Prototype), Role of Return Statement, Recursion, One Dimensional and Two-Dimensional Arrays. String Functions. Other Operators, Operator Precedence and associativity.</p> <p>Debugging: identify and fix errors. Different types of debugging techniques.</p>	12																				
Lab Programs	<p>UNIT-II- Basic Problem-Solving Techniques</p> <ol style="list-style-type: none"> 1. Converting degrees Celsius to Fahrenheit and vice versa? 2. Display three input numbers in sorted(non-decreasing) order? 3. Given a positive integer value $n (>= 0)$ display number, square and cube of numbers from 1 to n in a tabular format? 4. Given an input positive integer number, display odd numbers from in the range $1, n]$? 5. Display first mathematical tables, each table upto 10 rows? Generalise this to display first $n (> 0)$ mathematical tables up to m ($m > 0$) rows? 6. Display following patterns of n rows ($n > 0$), For the below examples $n = 5$? For each pattern write a separate algorithm/program? <table border="1" style="margin-left: auto; margin-right: auto;"> <tbody> <tr> <td>\$</td> <td>\$</td> <td>12345</td> <td>12345</td> </tr> <tr> <td>\$\$</td> <td>\$\$</td> <td>1234</td> <td>1234</td> </tr> <tr> <td>\$\$\$</td> <td>\$\$\$</td> <td>123</td> <td>123</td> </tr> <tr> <td>\$\$\$\$</td> <td>\$\$\$\$</td> <td>12</td> <td>12</td> </tr> <tr> <td>\$\$\$\$\$</td> <td>\$\$\$\$\$</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	\$	\$	12345	12345	\$\$	\$\$	1234	1234	\$\$\$	\$\$\$	123	123	\$\$\$\$	\$\$\$\$	12	12	\$\$\$\$\$	\$\$\$\$\$	1	1	
\$	\$	12345	12345																			
\$\$	\$\$	1234	1234																			
\$\$\$	\$\$\$	123	123																			
\$\$\$\$	\$\$\$\$	12	12																			
\$\$\$\$\$	\$\$\$\$\$	1	1																			

7. Display the following patterns of n rows($n > 0$), for the below examples $n=5$?

Hollow square pattern:	Triangle Patterns with numbers:	Square with diagonals:	Diamond Pattern
<pre>##### # # # # # # # # # #####</pre>	<pre>1 121 12321 1234321 123454321</pre>	<pre>***** * * * * * * * * * * * * * * * * *</pre>	<pre>* * *** * ***** * *** * *</pre>

8. Given the first term (a), difference/multiplier (d) and number of terms ($n > 0$), display the first n terms of the arithmetic/geometric progression?
9. Display the first $n(n > 0)$ terms of the Fibonacci sequence?
10. Display the first $n(n > 0)$ terms of the Tribonacci sequence?
11. Given two positive integer numbers n_1 and n_2 check if the numbers are consecutive numbers of the Fibonacci sequence?
12. Compute approximate value of π considering first $n(n > 0)$ terms of the Taylor series for π ?
13. Compute approximate value of e^x considering first $n(n > 0)$ terms of the Taylor series for e^x ?
14. Compute approximate value of $\sin(x)/\cos(x)$ considering first $n(n > 0)$ terms of the Taylor series for $\sin(x)/\cos(x)$?

UNIT-III Problems on Numbers:

15. Extract digits of an integer number(left to right and right to left)?
16. Given a sequence of digits form the number composed of the digits. Use sentinel Controlled repetition to read the digits followed by -1. For example, for the input 2 7 3 2 9 the output number is 27329?
17. Check if a given positive integer number is a palindrome or not?
18. Compute character grade from the marks ($0 \leq \text{marks} \leq 100$) of a subject. Grading Scheme 80-100 : A, 60 - 79: B, 50 - 59: C, 40-49: D, 0-39: F? Solve this using both else-if ladder and switch case?
19. Compute the sum of a sequence of numbers entered using sentinel controlled repetition?
20. Check if a given positive integer number is a prime number or not?
21. Compute prime factors of a positive integer number?
22. Check if two positive integer numbers are amicable numbers or not?
23. Check if a given positive integer number is a perfect number or not?
24. Check if a given positive integer number Armstrong number or not?
25. Converting a positive integer number ($n > 0$) from one base (input Base) to another base (output Base) ($2 \leq \text{input Base}, \text{output Base} \leq 10$). Input number should be validated before converting to make sure the number uses only digits allowed in the input base?
26. Write a program to display a number in text form. For example If the number is 5432 the

Boke
M
Op
556

- output should be "FIVE FOUR THREE TWO"?
27. Using the grading scheme described in the question 4(UNIT III), Compute how many Students awarded each grade and display the frequency as a bar chart (horizontal) using Single "*" for each student. Use sentinel controlled repetition (-1 a sentinel value) in reading the students marks. Use else-if ladder/switch case to compute the grade and the corresponding frequency.
28. Sample bar chart when the class has 7-A, 10-B, 3-C, 7-D and 1-F grades.
- A:

- B:

- C:***
- D:

- F:*
29. Compute maximum, minimum, sum and average of a sequence of numbers which are read using sentinel controlled repetition using only few variables?
30. Compute body mass index, $BMI = \text{weight in KGs}/(\text{Height in Meters} * \text{Height in Meters})$, Both weight and height values are positive real numbers. Your program should display BMI value followed by whether the person is Underweight, Normal, Overweight or Obese using the below ranges:
- BMI Values
Underweight: less than 18.5 Normal: ≥ 18.5 and < 25
Overweight: ≥ 25 and < 30
Obese: ≥ 30
- UNIT IV :**
31. Design a modularized algorithm/program to check if a given positive integer number is a circular prime or not?
32. Design a modularized algorithm/program to compute a maximum of 8 numbers?
33. Design a modular algorithm/program which reads an array of n integer elements and outputs mean (average), range (max-min) and mode (most frequent elements)?
34. Design a modular algorithm/program which reads an array of n integer elements and outputs median?
35. Implement your own string length and string reversal functions?
36. Design algorithm/program to perform matrix operations addition, subtraction and transpose?
37. Write a recursive program to count the number of digits of a positive integer number?
38. Recursive solutions for the following problems:
- Factorial of a number
 - Display digits of a number from left to right and right to left
 - Compute x^y using only multiplication?
 - To print a sequence of numbers entered using sentinel controlled repetition in reverse order?

Text Books:

1. Harvey Deitel and Paul Deitel, C How to Program, 9th edition, Pearson India, 2015.
2. Dromey, R. G. How to Solve It by Computer. Pearson Education, 1982.
3. Programming in C- Balaguruswamy, Mc Graw Hills
4. Kanetkar, Yashavant. Let Us C. BPB Publications, 2020.
5. Venkatesh, Nagaraju Y., Practical C Programming for Problem Solving, Khanna Book Publish Company, 2024.

Reference Books:

1. Brian W. Kernighan and Dennis Ritchie, The C Programming Language, 2nd edition, Pearson, 2015.
2. Jeri Hanly and Elliot Koffman, Problem Solving and Program Design in C, 8th edition, Pearson, 2015.
3. Goyal K. K., Sharma M. K., and Thapliyal M. P., Concept of Computer and C Programming, Univer Science Press.
4. Exploring C – YashwantKanetkar, BPB Publications
5. Programming with C -K.R .Venugopal, TATA McGrawHill
6. Computer Programming in C, V. Rajaraman, PHI
7. Programming with C, Byron Gottfried, TATA McGrawHill

Code: BCA- 1005	AEC-I	General English - I	2L+T: 0P	2 Credits
(30 hours theory)				

Max Marks: 100; Theory: 100 (Int: 25; Ext: 75)

Course Outcomes: Upon completion of the course, students will be able to

CO1: Develop a strong vocabulary foundation and understand word formation, prefixes, suffixes, synonym and antonyms to improve English language proficiency.

CO2: Apply basic writing skills, including sentence structures, paragraph organization, coherence, ar punctuation, to enhance written communication.

CO3: Identify and correct common grammatical errors in writing, including subject-verb agreeemer misplaced modifiers, and redundancy.

CO4: Demonstrate the ability to write effectively using descriptive, definitional, and classificatory technique including providing examples and writing structured content.

CO5: Improve written communication through comprehension, précis writing, and essay writing techniques.

CO6: Develop oral communication skills, including pronunciation, intonation, workplace communication, ar formal presentations, through interactive language lab sessions.

Unit	Topics	Purposed lectures
I	Vocabulary Building The concept of Word Formation, Root words from foreign languages and their use in English, Acquaintance with prefixes and suffixes from foreign languages in English to form derivatives, Synonyms, antonyms, and standard abbreviations.	6
II	Basic Writing Skills Sentence Structures, Use of phrases and clauses in sentences, Importance of proper punctuation, Creating coherence, Organizing principles of paragraphs in documents, Techniques for writing precisely.	6
III	Identifying Common Errors in Writing Subject-verb agreement, Noun-pronoun agreement, Misplaced modifiers, Articles, Prepositions, Redundancies, Tenses.	4
IV	Nature and Style of Sensible Writing Describing, Defining, Classifying, providing examples or evidence, Writing introduction and conclusion.	4
V	Writing Practices Comprehension, Précis Writing, Essay Writing.	4
VI	Oral Communication Listening Comprehension, Pronunciation, Intonation, Stress and Rhythm, Common Everyday Situations: Conversations and Dialogues, Communication at Workplace, Interviews, Formal Presentations.	6

Blue

2/10/2024