Structure & Syllabus for Semester-I

BACHELOR OF COMPUTER APPLICATIONS (BCA) Programme

Gujarat University

2023 – 2024

As per NEP 2020 CURRICULUM AND CREDIT FRAMEWORK FOR UNDERGRADUATE PROGRAMMES, UGC

&

Resolution No. KCG/admin/2023-24/0607/kh.1

of

Education Department, Govt. of Gujarat

STRUCTURE FOR SEMESTER - 1
GUJARAT UNIVERSITY
Bachelor Of Computer Applications (B.C.A.)
(as per NEP 2020)

COURSE:	**Bachelor Of Computer Applications (B.C.A.)**
MAJOR:	**COMPUTER SCIENCE**
MINOR:	**1. COGNITIVE MATHS**
	2. DATA SECURITY
	3. WEB TECHNOLOGIES

SEMESTER - 1

MAJOR

CODE	COURSE	CREDITS
DSC-C-BCA-111T	Elements of Programming & Computer Organization	4
DSC-C-BCA-112P	C Programming Practical	4

MINOR

CODE	COURSE	CREDITS
DSC-M-BCA-113T	Mathematical Foundation	2
DSC-M-BCA-113P	Office Automation	2
	OR	
DSC-M-BCA-113T	Data Encryption & Compression Technique	2
DSC-M-BCA-113P	Data Encryption & Compression Technique Practical	2
	OR	
DSC-M-BCA-113T	Introduction to Web Technologies	2
DSC-M-BCA-113P	Web Designing	2

Semester - I

BACHELOR OF COMPUTER APPLICATIONS

MAJOR : COMPUTER APPLICATIONS

Course Name: Elements of Programming & Computer Organization

Course Code: DSC-C-BCA-111T

Credits: 04

Course Outcomes:

At the end of the course, the student will be able to

- CO1: Develop proficiency in the logic building
- CO2: Develop Programming skills
- CO3: Get a good understanding of a basic computer design and its organization
- CO4: Develop an understanding of logic circuits, digital components, registers and micro-operations of a basic computer system.

Prerequisites:

Fundamentals of a Computer System, Logical Reasoning Ability

Contents:

No.	Particulars	Hours	Credits
Unit 1.	**Pre-Programming Techniques** Introduction to Programming Languages: Introduction to Machine level language, Introduction to Assembly language, Introduction to Higher level language, Limitations and Features, Classification of Computer Language: Procedural Language and Non Procedural Language, Tools and Techniques of Problem Analysis: Algorithm Development and Flow Chart with numerous examples, Getting Started With 'C' Language: Basic Structure of C, Executing C program, Character set & C Tokens, Identifiers & Keywords, Data Types, Constants and Variables, Type Casting, Comments.	15 Hrs	01
Unit 2.	**C Language Operators and Decision Making** Operators & Expression: Types of Operators and Expression, Precedence and Associativity, Console Based I/O: printf(), scanf(), getch(), getchar(), putchar() and concept of Header File	15 Hrs	01

and #include, #define, Decision Making Structure: If, If-else, Nested If-else, Multiple If Structure, Switch, break, goto statement

Unit 3. **Digital Computers: Components and Micro-operations** **15 Hrs** **01**

Digital Logic Circuits: Digital Computer (Block diagram), Logic Gates (truth-table and graphic symbol): AND, OR, NOT, BUFFER, NAND, NOR, X-OR, X-NOR, Half Adder: truth-table and logic diagram, Full Adder: truth-table and logic diagram, Flip-Flops (graphic symbol, logic diagram and characteristic table): SR, D, JK & T, Digital Components: Decoders (logic diagram and truth-table) – 3-to-8 line decoder, Encoders (truth-table and Boolean functions): Octal-to-Binary, Multiplexer (logic diagram and truth-table): 4-to-1 line multiplexer, Registers: definition, 4-bit register with logic diagram, 4-bit shift register with logic diagram, Binary Counters: 4-bit synchronous binary counter, Data Representation: Complements: (r-1)'s and (r)'s, Fixed-point representation, Floating point representation with Normalization Error (Overview), Register Transfer and Micro-operations: Bus system for four registers, Three-state bus buffers (graphic symbol and logic diagram), Arithmetic Micro-operations: 4-bit binary adder, 4-bit adder-subtractor, 4-bit binary incrementer, function table of 4-bit arithmetic circuit, Logic Micro-operations: list of logic Micro-operations, Shift Micro-operations (Overview), Arithmetic Logic Shift Unit with function table only

Unit 4. **Digital Computers: Design and Organization** **15 Hrs** **1**

Basic Computer Organization and Design: List of registers for basic computer and their functions, 16-bit common bus system, Computer instructions: format of memory reference, register reference and input-output instructions, Instruction cycle (flow-chart), Interrupt cycle with flow-chart, Central Processing Unit: Instruction formats, Addressing modes (Brief Overview), Input-Output Organization: Input-output interface, connection of I/O bus to input-output devices, isolated versus memory-mapped I/O, Asynchronous data transfer: handshaking, Modes of transfer: programmed I/O, Interrupt-initiated I/O, Direct memory access, DMA with block diagram of DMA controller, DMA transfer, Memory Organization: Associative memory – block diagram, Cache memory and mappings, Virtual memory (definition only)

References:

1. Reema Thareja, “Introduction to C Programming”, Oxford Publication
2. E. Balaguruswamy, “Programming in ANSI C”, 5th Edition, McGraw Hill Publication
3. M. Morris Mano, “Computer System Architecture”, 3rd Edition, Pearson Publication
4. Pradip Dey, Manas Ghosh, “Computer Fundamentals & Programming in C”, Oxford Publication
5. B. Govindrajalu, “Computer Architecture and Organization”, 2nd Edition, TMH Publication
6. Online References:
    - https://www.tutorialspoint.com/cprogramming
    - http://www.javatpoint.com/c-programming-language-tutorial
    - https://www.programiz.com/c-programming
    - http://www.cprogramming.com/tutorial/c-tutorial.html
    - http://www.programmingsimplified.com/c-program-examples
    - https://www.tutorialspoint.com/computer_logical_organization/index.htm
    - https://en.wikipedia.org/wiki/Computer_architecture
    - http://nptel.ac.in/courses/106103068/#

Accomplishments of the student after completing the Course:

After completion of this course, students will be able to

- Build logic reasoning to solve problem definitions
- Develop basic programming skills
- Understand a design and organization of a basic computer system
- Design logic circuits, differentiate digital components, perform micro-operations on registers used in a basic computer system.

Course Name : C Programming Practical

Course Code : DSC-C-BCA-112P

Course Outcome :

The aim of this course is to enable students to

- CO1: Acquire knowledge of advanced C programming concept.
- CO2: To understand Array, Loop, Structure, Union and Pointers.

Prerequisites : Knowledge of Computers

Contents :

Unit No.	Particulars	Hours	Credits
Unit 1.	**Loops & Decision Making Control**	**30 Hrs**	**01**

1. Demonstrate printing Table of given number using loop.
2. Demonstrate printing Tables for the range given by user.
   E.g. Start no : 1, Ending no : 10
   Program should print table(s) of 1 to 10 numbers.
3. Demonstrate to find sum of N numbers.
4. Demonstrate to find factorial of given number.
5. Demonstrate to find maximum from given N inputs by user.
6. Demonstrate to find reverse of a given number.
7. Demonstrate to find sum of the digits entered by the user.
8. Demonstrate to generate Fibonacci series up to N numbers.
9. Demonstrate to find GCD and LCM of given 2 numbers.
10. Demonstrate to find the sum of first 100 odd nos. and even nos.
11. Demonstrate to check whether given is Palindrome or not.
12. Demonstrate to check whether the given number is Prime or not.
13. Demonstrate to check whether the given number is Armstrong or not.
14. Demonstrate to print all the prime numbers ranging from 50 to 100.
15. Demonstrate to find x1+x2+x3+x4+ ….+xn.

Demonstrate to find 1+1/2+1/3+1/4+ …+1/n.

Unit 2 Patterns, Arrays, String & Functions 30 Hrs 01

1. Print the following pyramid. [Left aligned, Right Aligned, Center Aligned]

```
*
* *
* * *
* * * *
```

2. To accept an integer N, if the integer N = 4, then print the pyramid :

```
1
121
12321
1234321
```

3. To accept an integer N, if the integer N =4, then print the pyramid :

Print the following Pattern:

```
A
B  C
D  E  F
G H  I  J
```

4. Print the following pattern:

```
A
ABA
ABCBA
ABCDCBA
```

5. Print the following pattern:

```
1
0 1
0 1 0
1 0 1 0
```

6. Display this kind of output on screen.

```
C
CP
CPR
. .
CPROGRAMING
.
.
CPR
CP
C
```

7. A program which will take 10 numbers from user and stored it in the array. It will print all the numbers, their sum and average of it.
8. Demonstrate to sort an array.
9. Demonstrate to search an element from the array.
10. Demonstrate to find addition of two matrices of 3*3.
11. Demonstrate to find multiplication of two matrices of 3*3.
12. Input a string from the user and check whether the string is palindrome or not.
13. Demonstrate to find sum, average of two numbers passed to user defined functions called sum and average.
14. Demonstrate to print factorial of a given number by recursive user defined function fact(int).
15. Demonstrate to print Fibonacci series using recursive UDF.
16. Demonstrate to find length of the given string (without including string.h).
17. Demonstrate to convert lowercase string to uppercase string (without including string.h).
18. Accept two strings from the user and print the message that the strings are same or not.
19. Take a lowercase string from the user and print its length and uppercase string.
20. A program that uses function digit(n, k) that return the value of the kth digit the right of the number N. For e.g. The function call digit (254693,2) should return 9.
21. Demonstrate to find if the given no. is prime or not. The function should accept the number as an argument and return if the no. is prime or not.

Unit 3 **Structures and Union** **30 Hrs** **01**

1. Define structure called state having member variables as state name, number of districts and total population. Input and display 5 State data.
2. Define a structure called Item having member variables: Item code, Item name, item price. Create an array of structure to store five Items. Create a function which accepts the Item array and modifies each element with an increase of 10% in the price.
3. Define a structure to represent a date. Use your structures that accept two different dates in the format mm dd of the same year. Write a C program to display the month names of both dates.
4. Define a structure that can describe a Hotel. It should have members that include name, address, grade, room

charges, grade and no of rooms. Write a function to print out all hotel details with room charges less than a given value.

5. Accept records of different states using array of structures. The structure should contain state and number of engineering colleges, medical colleges, management colleges and universities. Calculate total colleges and display the state, which is having highest number of colleges.
6. Define a structure by name time with members seconds, minutes and hours. If time1 and time2 are two variables of the structure type, write a program to find the difference in time using a function.
7. Declare structure having state name, population, literacy rate and per capita income. Input 5 records. Display the state whose literacy rate is highest and whose per capita income is highest.
8. Define a structure employee with member variables having employee name, basic pay, dearness allowance, house rent, net salary. Store data of 5 employees. Write a function which calculates the net salary of employees and prints all employee details in descending order of their net salary.
9. Define a structure with tag population with fields Men and Women. Create structure with in structure using state and population structure. Read and display the data.
10. Demonstrate the use of Union.

Unit 4 **Pointers** **30 Hrs** **01**

1. Demonstrate a user-defined function which will swap the values of two variables using pointer (Call by reference).
2. Demonstrate a user defined function calc(), which will return the sum, subtraction, multiplication, and division values of two variable locally declared in the main function use of pointer.
3. Demonstrate UDF which will return the length of the string declared locally in the main (use pointer).
4. Write a program in C to find the maximum number between two numbers using a pointer.
5. Write a C program to input and print array elements using pointer.
6. Write a C program to copy one array to another using pointers.
7. Write a C program to compare two strings using pointers.
8. Write a C program to find reverse of a string using

pointers.

9. Write a C program to sort array using pointers.
10. Write a C program to concatenate two strings using pointers

References :

1. Reema Thareja, “Introduction to C Programming”, Oxford.
2. E. Balaguruswamy, “Programming with ANSI C”, 5th Edition, McGraw Hill.
3. Computer Fundamentals & Programming in C Publication: Oxford By Pradip Dey, Manas Ghosh
4. Yashavant Kanetkar, “Let Us C: Authentic guide to C programming language”, BPB
5. Online References:
   - https://www.tutorialspoint.com/cprogramming
   - http://www.javatpoint.com/c-programming-language-tutorial
   - https://www.programiz.com/c-programming
   - http://www.cprogramming.com/tutorial/c-tutorial.html
   - http://www.programmingsimplified.com/c-program-examples

Accomplishments of the student after completing the course:

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

- Use programming environment to create, compile, debug and execute C programs
- Acquire logical thinking, write efficient programs for solving problems