SANKALCHAND PATEL UNIVERSITY
।। अथातो ज्ञानजिज्ञासा ।।

FACULTY OF COMPUTER SCIENCE

Post Graduate Diploma in Computer Science and Application

In Effect from Academic Year 2016-17 **(Sem-I)**

Subject Code: 1CS4010101	**Subject Title: INTRODUCTION TO COMPUTER PROGRAMMING**
Pre-requisite	-

Course Objective:

This course is intended to develop problem solving skills in students with basics of programming using logic. Student is expected to learn problem solving using algorithm & flowchart techniques and implementation of problem using 'C' programming. The course aims to make the students formulate a problem in a programming language and solve it.

Teaching Scheme (Hours per week)				**Evaluation Scheme (Marks)**				
				Theory		**Practical**		
Lecture	**Tutorial**	**Practical**	**Credit**	**University Assessment**	**Continuous Assessment**	**University Assessment**	**Continuous Assessment**	**Total**
4	1	2	6	70	30	30	20	150

Subject Contents			
Sr. No.	**Topics**	**Total Hours**	**Weight (%)**
1	**Overview of C:** History of C, Importance of C, Basic Structure of C Programs, Executing C Program. **Constants, Variables and Data Types of C:** Character set, C tokens, keywords and identifiers, constants, variables, data types, declaration of variables, assigning value to variable, defining symbolic constants. **Operators and Expressions in C:** Introduction, Operators: arithmetic, relational, logical, assignment, increment and decrement, conditional, bitwise and special. Arithmetic expressions, evaluation of expressions, precedence of arithmetic operators, type conversions in expressions, operator precedence and associativity.	**9**	**20**
2	**Managing Input and Output Operators: Introduction**, Reading a character/number, Writing a character/number, Formatting Input and Output. **Decision Making Branching:** Introduction, Decision making with *if* statement, Simple *if* statement, the *if-else* statement, Nesting of *if-else* statements, the *else-if* ladder, the *switch-case* statement, *goto* statement.	**9**	**20**
3	**Loop Structures:** Introduction, *while*, *do-while* and *for* statement. *break* and *continue* statements in loop**.** Nesting of loops. **Arrays:** Introduction, One-dimensional arrays, Two-dimensional arrays, Initialization, Sorting and Searching using arrays, Concept of Multidimensional arrays.	**9**	**20**
4	**Character Array and Strings:** Declaring and initializing string variables, reading string from terminal, writing string to screen, string handling functions. **User-Defined Functions:** Need for user-defined functions, elements of user-defined functions, definition of functions, function calls, function declaration, category of functions, nesting of functions, recursion, the scope, visibility and lifetime of variables in functions.	**9**	**20**
5	**Structures and Unions:** Defining a structure, declaring structure variables, accessing structure members, structure initialization, arrays of structures, arrays within structures, unions. **Pointers:** Definition, accessing the address of variable, declaring and initializing pointer variables, accessing a variable through its pointer, pointer expressions, pointer increments and scale factor.	**9**	**20**

Course outcomes:

At the end of this course, the student would be able

- To have fundamental knowledge on flowcharts and algorithms
- To formulate the problem and express the same using flowcharts and algorithms
- To understand the basic terminology used in computer programming using C
- To study, analyze and understand logical structure of a computer program, and different construct to develop a program in 'C' language
- To write, compile and debug programs in C language
- To design programs involving decision structures, loops and functions

List of References:

1. Programming in ANSI C, By E Balagurusamy, Tata McGraw-Hill Publishing Company Limited.
2. Programming with C, By Bayron Gottfried, Tata McGraw-Hill Edition.
3. Let Us C, By Yashavant Kanetkar, BPB Publications.
4. Working with C, By Yashavant Kanetkar, BPB Publications.
5. The Spirit of C an Introduction to Modern Programming, By Mullish Cooper, Jaico Publishing House.
6. The C Programming Language, By Brian W. Kernighan and Dennis M. Ritchie, PHI.
7. C in Depth, By Suresh K. Srivastava, BPB Publications.
8. Programming in C, by Pradip Dey & Manas Ghosh, Publisher – Oxford
9. Programming in C, by Reema Thareja Publisher – Oxford

E-Resources / Web Links:

- http://www.cprogramming.com/

List of Experiments:

Note: The experiment list provided beneath is for reference only. The course teacher may change/formulate it as per his/her methodology and requirement.

1. Write a C program to display "Hello Friends!" on the screen.
2. Write a C program to print roll no, name and address of student.
3. Write a C program to implement the concept of symbolic constants.
4. Write a C program to perform arithmetic operations on two numbers.
5. Write a C program to find the area of circle using the formula Area=PI * r * r.
6. Write a C program to implement the concept of assignment operators.
7. Write a C program to implement the concept of increment and decrement operators.
8. Write a C program to implement the concept of conditional operators (? :).
9. Write a C program that reads a single character from screen and also print that character on the screen.
10. Write a C program to implement various formatting options for reading integers and real numbers.
11. Write a C program that shows the various formats for output of integers and real numbers.
12. Write a C program to display that the entered number is Negative, Positive or Zero.
13. Write a C program to check given number is odd or even number.
14. Write a C Program that will find the largest no from the given three nos.
15. Write a C program to implement the concept of else if ladder statement.
16. Write a C program to implement the concept of switch statement.
17. Write a C program to find the sum of first 100 natural nos.
18. Write a C program to find factorial of accepted numbers.
19. Write a C Program to find the sum of the digits of given number.
20. Write a C program to display first 25 Fibonacci nos.
21. Write a C Program that check whether the given no is prime or not.

22. Write a C program to print accepted no and its reverse number.
23. Write a C program to check whether the given number is palindrome or not.
24. Write a C program to display following output on the screen.

```
1
12
123
1234
```

```
1
22
333
4444
```

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

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

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

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

```
      1
    2   3
  4   5   6
7   8   9   10
11  12  13  14  15
```

```
C
CP
CPR
CPRO
:
CPROGRAMMING
:
CPRO
CPR
CP
C
```

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

```
11  12  13  14  15
  7   8   9   10
    4   5   6
      2   3
        1
```

25. Write a C program to calculate the sum and average of elements of an array.
26. Write a C program to find minimum and maximum number from given array.
27. Write a C program to arrange the accepted numbers in ascending order.
28. Write a C program to read a line of string from screen and also prints the same string on the screen.
29. Write a C program to find the length of given string.
30. Write a C program to copy one string into another string.
31. Write a C program to compare two strings together.
32. Write a C program to concate two strings.
33. Write a C program to implement function with no argument and no return value.
34. Write a C program to implement function with argument and no return value.
35. Write a C program to implement function with argument and a return value.
36. Write a C program to implement function with no argument and a return value.
37. Write a C program to implement the concept of nesting of functions.
38. Write a C program that shows the use of recursion.
39. Write a C program to implement the properties of static variable.
40. Write a C program to print detail of students like Rno, name, address, city, phone on screen. (Using structure.)
41. Write a C program to implement initialization of structure.
42. Write a C program to implement the concept of array of structure.
43. Write a C program to implement the concept of array within structure.
44. Write a C program to implement the concept of union.
45. Write a C program to print the address of variables along with their value.
46. Write a C program for accessing a variable through its pointer.
47. Write a C program that shows the use of pointers in arithmetic operations.
48. Write a C program to swap the values of two different nos. using UDF and pointer.