GUJARAT TECHNOLOGICAL UNIVERSITY, AHMEDABAD, GUJARAT

COURSE CURRICULUM

Course Title: Advanced Computer Programming
(Code: 3320702)

Diploma Programmes in which this course is offered	Semester in which offered
Computer Engineering, Information Technology	Second Semester

1. RATIONALE

Students in the previous semester have learned procedure for developing programs to solve simple problems using basic features of very popular language i.e. structured programming language `C'. This course deals with some advanced features of the 'C' language. The programming skills thus acquired can be used for developing programs with advance level programming features which in turn will be helping in developing practical applications for the scientific, research and business purposes.

2. COMPETENCY

The course content should be taught and implemented with the aim to develop different types of skills so that students are able to acquire following competency:

i. **Develop structured, modular and memory efficient programs in 'C' using arrays, functions, pointers and data files.**

3. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (L+T+P)	Examination Scheme				
				Theory Marks		Practical Marks		Total Marks
L	T	P	C	ESE	PA	ESE	PA	200
3	0	4	7	70	30	40	60	

Legends: L-Lecture; **T** – Tutorial/Teacher Guided Student Activity; **P** - Practical; **C** – Credit;; **ESE** - End Semester Examination; **PA** - Progressive Assessment.

Note: It is the responsibility of the institute heads that marks for **PA of theory** & **ESE and PA of practical** for each student are entered online into the GTU Portal at the end of each semester within the dates specified by GTU.

4. DETAILED COURSE CONTENTS

Unit	Major Learning Outcomes	Topics and Sub-topics
Unit – I Arrays	1a.Develop, debug and execute programs which use reading, writing and manipulating single dimensional and multidimensional arrays.	1.1Declaring and initializing One-Dimensional Array and array Operations i. Insertion ii. Searching iii. Merging iv. Sorting v. Deletion 1.2 Introduction of String as array of characters Declaration and Initialization of String 1.3Two-Dimensional Array and its Operations i. Insertion, Deletion ii. Matrix addition operation 1.4 Multi-Dimensional Arrays 1.5 sscanf() and sprintf() Functions 1.6 Drawbacks of Linear Arrays
Unit– II Pointers	2a. Develop, debug and execute programs to perform memory access using Pointers	2.1 Introduction and Features of Pointers 2.2 Declaration of Pointer 2.3 Void Pointers 2.4 Array of Pointers 2.5 Pointers to Pointers
Unit– III Functions	3a. Develop, debug and execute modular programs by writing and using Functions	3.1 Basics of Functions 3.2 Built-in and user defined Functions 3.3 Using String, Math and other built-in functions 3.4 Advantages of using Functions 3.5 Working of a Function 3.6 Declaring, Defining and calling user defined Functions- 3.7 The return Statement 3.8 Call by Value and call by Reference 3.9 Function as an Argument 3.10 Recursion 3.11 Advantages and Disadvantages of Recursion
Unit– IV Preprocessor Directives	4a. Appreciate use of various header files 4b. Define, test and implement constant and Macro	4.1 Introduction 4.2 #define and #undef Directives 4.3 #include ,#line Directive 4.4 Predefined macros in ANSI C 4.5 Standard I/O Predefined Streams in stdio.h 4.6 Predefined macros in ctype.h
Unit– V Structure and	5a. Implement different data types under a	5.1 Introduction and Features of Structures 5.2 Declaration and Initialization of Structures

Unit	Major Learning Outcomes	Topics and Sub-topics
Union	single structure 5b. Utilize memory effectively using Union	5.3 Array of Structures 5.4 Pointers to Structure 5.5 typedef 5.6 Enumerated Data Type 5.7 Union 5.8 Union of Structures
Unit– VI Files	6a. Develop, debug and execute programs to read and write data from secondary storage devices	6.1 Introduction 6.2 File Operations i. Opening a File ii. Reading a File iii.Closing a File 6.3 Text Modes 6.4 Binary Modes 6.5 File Functions i. fprintf() ii. fscanf() iii.getc() iv.putc() v. fgetc() vi.fputc() vii.fseek() viii. feof() 6.6 Command Line Arguments

5. SUGGESTED SPECIFICATION TABLE WITH HOURS & MARKS (THEORY)

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks (Duration – 48 Hours)			
			R Level	U Level	A Level	Total
1.	Arrays	8	4	5	6	15
2.	Pointers	8	4	5	5	14
3.	Functions	8	4	5	6	15
4.	Preprocessor Directives	4	1	2	2	5
5.	Structure and Union	6	2	3	4	9
6.	Files	8	3	4	5	12
	Total	**42**	**18**	**24**	**28**	**70**

Legends: R = Remember; U = Understand; A = Apply and above levels (Bloom's revised taxonomy)

Note: This specification table shall be treated as only general guideline for students and teachers. The actual distribution of marks in the question paper may vary from above table.

6. SUGGESTED LIST OF PRACTICAL/EXERCISES

Write, test, debug and execute following programs to develop different types of skills leading to the achievement of desired competency as mentioned. Out of the following enough practical/Exercise should be selected from each unit to give total workload of 56 hours to students.

S. No.	Unit No.	Practical/ Exercises	Approx Hrs. Required
1	I	Write, test, debug and execute minimum five programs with array operations like insertion, searching, merging, sorting and deletion.	06
2	I	Write, test, debug and execute minimum five programs using two Dimensional and Multi-Dimensional arrays.	06
3	II	Write, test, debug and execute four programs using Pointers.	06
4	II	Write, test, debug and execute programs using array of Pointers and pointers of pointers.	06
5	III	Write, test, debug and execute programs using String functions strlen(), strcpy, strcmp(), strlwr(),strupr(), strchr(), strcat() and common math and other functions like sqrt(), pow(), ceil(), round(), sin(), cos(), tan(), div(), abs() etc .	06
6	III	Write, test, debug and execute programs using functions and passing function arguments.	06
7	III	Write, test, debug and execute programs using recursive functions.	04
8	IV	Write, test, debug and execute programs for implementing Preprocessor Directives such as constants and Macros	02
9	V	Write, test, debug and execute programs with various features of Structures	04
10	V	Write, test, debug and execute programs using Union and Union of structures	06
11	VI	Write, test, debug and execute programs using elementary read/write file operations.	06
12	VI	Write, test, debug and execute programs using fprintf(), fscanf(), getc(), putc(), fgetc(), fputc(), fseek(), feof() functions.	12
		Total	**70**

7. SUGGESTED LIST OF PROPOSED STUDENT ACTIVITIES

7.1 Students will prepare file for the above mentioned Practical

7.2 Prepare presentation and deliver seminar on various topics covered like String functions, Pointers, Arrays, File Functions, Structures and Unions,

7.3 Students are expected to develop minimum one program of particular topic as an example to exhibit real life application.

8. SUGGESTED LEARNING RESOURCES

A. List of Books

Sr.No.	Author	Title of Books	Publication
1	Kamthane,A.N.	Programming in 'C'	Pearson,2012
2	Balaguruswami,E.	Programming in ANSI C	TMH,2012
3	Kanetkar, Yashavant	Let us 'C'	BPB publications,2010

B. List of Major Equipment/ Software

1 Computer System with latest configuration

2 'C' Compiler

C. List of Software/Learning Websites

1 'C' Programming Language: http://www.w3schools.in/c-programming-language/intro/

2 Learn C Online**:** http://www.learnconline.com/

3 **'**C' Frequently Asked Questions**:** http://www.c-faq.com

4 **'**C' Programming: http://www.cprogramming.com

5 Sams Teach Yourself C in 24 Hours: http://aelinik.free.fr/c/

9. COURSE CURRICULUM DEVELOPMENT COMMITTEE

Faculty Members from Polytechnics

1. **Dr. P.P.Kotak** Head Computer Engg. Dept,AVPTI, Rajkot
2. **Prof. K. N. Raval** Head Computer Engg. Dep, RCTI , Ahmedabad
3. **Prof. R. M Shaikh** Head Computer Engg. Dept, KD Polytechnic, Patan.
4. **Prof. S. D. Shah** Lect. Computer Engg. Dept, RCTI, Ahmedabad

Co-ordinator and Faculty Members from NITTTR Bhopal

1. **Dr. K. J. Mathai**, Associate Professor Dept. of Computer Engineering and Applications
2. **Dr. R. K. Kapoor**, Associate Professor Dept. of Computer Engineering and Applications