

COURSE TITLE- PROGRAMMING LANGUAGE
 COURSE CODE 6X203

PROGRAMME & SEMESTER

Diploma Programme in which this course is offered	Semester in which offered
ELECTRONICS AND TELECOMMUNICATION ENGINEERING	Second

1. RATIONALE

C forms the basics of C++, Visual C / C++ etc. which is current requirement in the fields of computer science (CS) and information technology (IT). It combines features of both the high level and low level language. C is used for creating computer applications that are used in writing embedded software/ firmware for various micro-controller based products in electronics, communications and in industry. It is widely used to develop system programming & operating systems. C is also used in developing simulators, verification software and test code for various applications and hardware products. It is therefore vital for electronic engineers to develop interest and skill in C language.

2. COMPETENCY

"Develop programs in 'C' language."

3. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (Hours/ Credits)			Total Credits (L+T+P)	Examination Scheme (Marks)				
				Theory		Practical		Total
L	T	P	C	ESE	PT	ESE @ (PR)	PA (TW)	
2	-	4	6	80	20	25	25	150
Duration of the Examination (Hrs)				3	1	2	--	

Legends : L-Lecture; T-Tutorial/Teacher Guided Theory Practice ; P- Practical; C- Credits; ESE- End Semester Examination; PT – Progressive Test, PA- Progressive Assessment, OR – Oral Examination, TW - Term Work, # External, @ Internal

COURSE OUTCOMES

- I Distinguish between various data types.
- II Develop programming logic for given problem.
- III Implement looping in programs.
- IV Create own functions and use library functions.
- V Highlight effectiveness of Arrays, Structures and Pointers for data access.
- VI Develop projects using file management.



5. DETAILED COURSE CONTENTS :

Unit	Major Learning Outcomes (Cognitive Domain Only)	Topics And Sub-Topics
Unit - I Concepts, Constants, Variables and Data Types	1a. Prepare flowcharts 1b. Develop algorithms 1c. Describe concept of constants and variables 1d. Distinguish between various data types	1.1 Concepts of programming methodology : Flowchart Algorithm 1.2 Character set 1.3 'C' tokens 1.4 Keywords, Identifiers 1.5 Constants & Variables 1.6 Declaration of variables 1.7 Assigning values to variables 1.8 Data types 1.9 Storage class
Unit - II Operators and Expressions	2a. Create arithmetic and logical expressions 2b. Use input and output Functions 2c. Develop programming logic for given problem	2.1 Assignment operators 2.2 Relational operators 2.3 Logical operators. 2.4 Arithmetic operators 2.5 Conditional operators 2.6 Increment and Decrement operator 2.7 Bitwise operators and special Operators 2.8 Evaluation of arithmetic and logical expressions 2.9 Formatted input & output 2.10 I/O functions : scanf(), printf(), getch(), getch(), gets(), puts() 2.11 Programming exercises based on arithmetic and logical expressions
Unit - III Branching and Looping	3a. Develop decision making Programming routines. 3b. Implement looping in programs	3.1 GO TO statement 3.2 IF statement. 3.3 IF... else statement 3.4 Nested 'IF... else 'statement. 3.5 Else if ladder routine 3.6 SWITCH statement 3.7 The '?' Operator. 3.8 Programming based on decision Making. 3.9 FOR statement 3.10 WHILE statement 3.11 DO & DO... WHILE statements 3.12 Jumps in loops 3.13 Need of BREAK, CONTINUE statements in looping 3.14 Looping exercises in complex programming problems

CO6	Develop projects using file management.	2	2	1						2	4
TOTAL STRENGTH		3	1	2	0					2	32

Course Curriculum Design Committee

Dr. S. D. Yardi, Lecturer in Electronics and telecommunication, Government Polytechnic, Aurangabad.

HOD ET (Chairman, PBOS)

Co-ordinator(CDIC)Member Secretary



11. LIST OF MAJOR EQUIPMENTS AND MATERIALS REQUIRED :

S. N.	Name of equipment	Brief specification
1.	Personal Computers	32.bit
2.	Turbo C/ C++ Compiler	
3.	Multimedia projector, Tutorial Video CD (Programming in C), Expert video lectures.	

12. LEARNING WEBSITE & SOFTWARE

- 1 Software/tools : Turbo C or Borland C, Visual Studio
- 2 Theory and programming concepts: www.nptel.iitm.ac.in
- 3 <http://www2.its.strath.ac.uk/courses/c/>
- 4 <http://www.iu.hio.no/~mark/C/Tutorial/C-Tut-4.02.pdf>
- 5 www.nptelvideos.com/programming/c_programming_videos.php
- 6 www.ocw.mit.edu (Practical Programming in C - MIT Open Course Ware)
- 7 www.cprogramming.com

MAPPING OF PROGRAMME OUTCOMES (POs) AND PROGRAMME SPECIFIC OUTCOMES (PSOs) WITH COURSE OUTCOMES (COs)

SNo	Course Outcome	POs										PSOs	No. of hours allocated in curriculum				
		1	2	3	4	5	6	7	8	9	10			01	02		
CO1	Distinguish between various data types.	3	2	0													6
CO2	Develop programming logic for given problem.	3	3	2											2		6
CO3	Implement looping in programs.	3	3											3			6
CO4	Create own functions and use library functions	3	3											1			4
CO5	Highlight effectiveness of Arrays, Structures and Pointers for data access.	3	3											3			6

Unit - IV ARRAYS and Pointers	4a. Handle large volume of data of similar type. 4b. Use array, structures & pointers to relate it to real world data types	4.1 Introduction to ARRAYS and Strings 4.2 One dimensional , two dimensional arrays of integer, float and characters 4.3 String related functions: strcat(), strcpy(), strlen(), strcmp(), strchr(), strstr() 4.3 programming exercises based on Arrays 4.4 Introduction to Pointers 4.5 Initialization and declaration of Pointers 4.6 Programming exercises based on Pointers 4.7 Introduction to Structures
Unit - V User defined and library functions	5a. Understand the importance of User Defined Functions [UDF] 5b. Develop own functions and select suitable library functions for a programming logic	5.1 Introduction to User Defined Functions [UDF] 5.2 Call by reference and call by value 5.3 Library Functions: clrscr(), abs(), sqrt(), isdigit(), isalpha(), toupper(), tolower(), strlen(), strcat(), strcpy(), strcmp 5.4 Difference between library functions and UDF. 5.5 Recursive function (ex. factorial) 5.6 Programming exercises based on Functions
Unit - VI File Management	6a. Describe file operations 6b. Operate files in given projects	6.1 Introduction of file management. 6.2 Opening, closing, defining a file viz.: fopen(), fclose(), fscanf(), fprintf(), fseek(), rewind(), ftell(). 6.3 Input and output operations on files.



6. SUGGESTED SPECIFICATION TABLE FOR QUESTION PAPER DESIGN

Unit No	Title Of Unit	Teaching Hours	Distribution Of Theory Marks			
			R Level	U Level	A Level	TOTAL
1	Concepts , Constants, Variables and Data Types	6	06	06	00	12
2	Operators and Expressions	6	02	10	04	16
3	Branching and Looping	6	04	06	06	16
4	ARRAYS and Pointers	4	02	02	08	12
5	User defined and library functions	6	00	04	12	16
6	File Management	4	00	02	06	08
Total		32	14	30	36	80

Legends: R – Remember, U – Understand, A – Apply and above (Bloom's revised Taxonomy)

7. LIST OF PRACTICAL / LABORATORY EXPERIENCES/ TUTORIALS

Sr. No.	Unit	Title Practical/ Lab. Work/ Assignments/ Tutorials	Hours
1.	I	Use the C compiler software for editing, compiling and running programs. Use menu options and initialize local variables. Output and Input data using printf(), scanf() statement.	06
2.	II	Perform simple arithmetic programs using floating point Output and input data in proper format using printf(), scanf() Develop & test programs using conditional, relational or logical operators Develop programming logic for addition of digits of a 4 digit number	14
3.	III	Develop & test programs using control structures like: if, if... else, nested if... else, else... if ladder Develop & test programs using switch, break, continue statements Develop & test programs with while, do... while, for loops	12
4.	IV	Develop & test programs to declare & initialize arrays Develop & test programs using character type arrays Develop & test programs with pointer variables Develop logic to implement loops for arranging 10 numbers in ascending order	12
5.	V	Develop & test programs using library functions of C Develop & test programs with user defined functions of C Create a program to pass the value of local variables into user	10

Unit No	Title Of Unit	Teaching Hours	Distribution Of Theory Marks			
			R Level	U Level	A Level	TOTAL
6.	VI					10
Total						64

8. SUGGESTED STUDENTS ACTIVITIES

1. Prepare journals based on practical performed in laboratory.
2. Solve logical problems using different software tools.
3. Simulate programs for various applications & debug.
4. Develop mini project.
5. Identify use of 'C' programming in embedded systems applications.
6. Multiple choice questions, short questions and answers.
7. Rapid code development.
8. Technical quiz, debate & seminar.

9. SUGGESTED SPECIFIC INSTRUCTIONAL STRATEGIES

1. Arrange guided visits to automation industries.
2. Motivate students to use internet for applications of C, C++ programming.
3. Students may be helped to develop logic on individual basis
4. Students must be encouraged for self directed learning to improve LOs/ Cos.

10. SUGGESTED LEARNING RESOURCE

S.No.	Name of Book	Author	Publication
1.	Let Us C	Kanetkar Yashvant (Twelfth Edition)	BPB Publications, 2012
2.	Introduction to C Programming (With CD ROM support)	Reema Thareja (First Edition)	Oxford University Press, 2012
3.	Programming in C	Balagurusamy, E (Fifth Edition)	Tata McGraw-Hill, New Delhi, 2012
4.	Programming in C	Kernighan Brian and Ritchie Dennis (Second Edition)	Prentice Hall of India Pvt. Ltd., New Delhi, 2012