GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Bachelor of Computer Applications

Level: Under Graduate

Course / Subject Code:

Course / Subject Name: Fundamental of Programming

w. e. f. Academic Year:	June-2024
Semester:	1
Category of the Course:	Core Course

Prerequisite: Basic Mathematics and knowledge about number systems

| **Rationale:** | The course content outlined encompasses fundamental concepts and practical skills essential for anyone aspiring to understand and develop proficiency in programming using the C language. Beginning with the basics of programming languages, students will grasp the foundational concepts of compilers, interpreters, linkers, and loaders, which are crucial for translating and executing code. This knowledge provides a solid groundwork for understanding how programs are processed and executed by computers, fostering a deeper appreciation of software development principles.

Moving into the specifics of C programming, the curriculum covers essential elements such as data types, variables, operators, and expressions. These topics are fundamental to constructing meaningful algorithms and executing logical operations within programs. Understanding decision control statements and iterative constructs equips learners with the ability to create programs that can make decisions and iterate through tasks efficiently. These skills are further honed through hands-on practice in implementing arrays, strings, and user-defined functions, enhancing both problem-solving abilities and code organization skills.

Moreover, the inclusion of pointers and structures introduces students to more advanced concepts in memory management and data structuring. Pointers, arrays, and structures enable efficient manipulation and organization of data, essential for optimizing program performance and managing complex data structures effectively. Additionally, the course provides practical experience in file handling, including reading from and writing to files, which is crucial for developing applications that require persistent data storage and retrieval.

Overall, this comprehensive course not only covers the syntax and semantics of the C programming language but also emphasizes practical application through numerous coding exercises and projects. By mastering these concepts and techniques, students gain a solid foundation in C programming, empowering them to tackle real-world programming challenges with confidence and proficiency. |
|---|---|

Course Outcome:

After Completion of the Course, Student will able to:

No	Course Outcomes	RBT Level*
1	Formulate algorithm/ flowchart for given arithmetic and logical problem.	UN
2	Translate algorithm/flowchart into C program using correct syntax of Operator, conditional, branching, iteration and execute it.	AP
3	Write programs using the concepts of array and functions.	AP
4	Write programs using the concepts of pointers, structure and union.	AP
5	Develop an application using the concepts file management to solve problems	CR

*RM: Remember, UN: Understand, AP: Apply, AN: Analyze, EL: Evaluate, CR: Create

Teaching and Examination Scheme:

Teaching Scheme (in Hours)			Total Credits L+T+ (PR/2)	Assessment Pattern and Marks				Total Marks
				Theory		Tutorial / Practical		
L	**T**	**PR**	**C**	ESE (E)	PA / CA (M)	PA/CA (I)	ESE (V)	
3	0	2	4	70	30	20	30	150

Course Content:

Unit No.	Content	No. of Hours	% of Weightage
1.	Programming Language, Compiler, Interpreter, Liker, Loader, Classification of Programming, Algorithm, Flowchart, Structure of a C Program, First C Program, Comments, C Tokens, Basic data types in C, Variables, Operators & Expression in C, Type conversion and Typecasting.	7	15
2.	Basic Screen & Keyboard Input & Output, Introduction to Decision Control Statements, Conditional execution & selection statements, Iterative Statements, Nested Loops, Special Control Statement.	9	20%
3.	Introduction to Array, One Dimensional Array, Multi-Dimensional Array, Introduction to String, Character & string Functions, Array of String Introduction to User Defined Functions, Using Functions, Function Declaration/Function Prototype, Function Definition, Function call, return statement, Passing Parameters to the function, scope of variables, Storage classes, Recursive Functions.	9	20%
4.	Introduction, Understanding Memory Addresses, Address Operator (&), Introduction to Pointers, void Pointer, Null Pointer, Use of Pointers, Arrays and Pointers, Pointer and String, Pointer Arithmetic, Pointers to Pointers, Array of Pointers, Pointers to an Array, Pointers to Functions. Structure, Structure Declaration, Assessing member of structure, Initialization of Structure, Nesting of Structure, Array of Structure, Array within structure, Structure and Function, Structure and Pointer,	9	25%

GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Bachelor of Computer Applications
Level: Under Graduate
Course / Subject Code:
Course / Subject Name: Fundamental of Programming

	Union.		
5.	Using Files in C, Declaration, Opening and Closing of a File, Working with Text Files and Binary Files, Character Input and Output, End of File (EOF), feof() Function, Files of Records, Random Access to Files of Records.	8	20%
	Total	**42**	**100**

Suggested Specification Table with Marks (Theory):

Distribution of Theory Marks (in %)					
R Level	**U Level**	**A Level**	**N Level**	**E Level**	**C Level**
0	20	60	0	0	20

Where R: Remember; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create (as per Revised Bloom's Taxonomy)

References/Suggested Learning Resources:
Text Books:

1. PradipDey, Manas Ghosh, "Programming in C", 2nd Edition, 2018, Oxford University Press, ISBN: 978-01-9949-147-6.

Reference Books:

2. E. Balaguruswamy, "Programming in ANSI C", 8th Edition, 2019, McGraw Hill Education, ISBN: 978-93-5316-513-0.
3. Yashavant P. Kanetkar, "Let Us C", 16th Edition, 2019, BPB Publications, ISBN: 978- 93-8728-449-4.
4. ReemaThareja. "Programming in C", 2nd Edition, Oxford University Press.
5. C: The Complete Reference, by Herbert Schildt, Publisher – Tata McGraw Hill.

Suggested Course Practical List:

1. Write a program to that performs as calculator (addition, multiplication, division, subtraction).
2. Write a program to find area of triangle(a=h*b*.5)

    a = area   h= height   b = base.
3. Write a program to calculate simple interest (i = (p*r*n)/100)

    i = Simple interest   p= Principal amount   r= Rate of interest   n=Number of years.
4. Write a C program to inter change two numbers.
5. WriteaCprogramtoenteradistanceintokilometreandconvertitintometer,   feet,   inches   and centimeter.
6. Write a program to compute Fahrenheit from centigrade(f=1.8*c +32).
7. Write a C program to find that the accept ednumber isNegative, Positive or Zero.
8. Writeaprogramtoreadmarksofastudentfromkeyboardwhetherthestudentis pass or fail (using

ifelse)

9. Write a program to read three numbers from keyboard and find out maximum out of these three.(nested ifelse)

10. Write a c program to prepare pay slip using following data. Da = 10% of basic, Hra = 7.50% of basic, MA= 300, Pf= 12.50%ofbasic, Gross=basic+Da+ Hra+Ma,Nt= Gross–Pf.

11. WriteaCprogramtofindouttheMaximumandMinimumnumberfromgiven10 numbers

12. Write a C program to find factorial of a given number.

13. Write a program to reverse a number.

14. Write a program to generate first *n* number of Fibonacci series

15. Write a program to calculate average and total of5 studentsfor3subjects(use nested *for* loops)

16. Read five persons height and weight and count the number of person having height greater than 170andweightlessthan 50,

17. Write a program to check whether the give n number is prime or not.

18. Write a program to evaluate the series $1^2+2^2+3^2+\ldots\ldots+n^2$

19. Write a program to print following patterns:

```
        i)   *              ii)   *****
             * *                  ****
             * * *                ***
             * * **               **
             * * ** *             *
```

20. Write a program to print following patterns:

```
        i)      1         ii)   12345
                12              1234
                123             123
                1234            12
                12345           1
```

21. Write a program to print following patterns:

```
i)  AAAAA                    ii)ABCDE
    BBBB                        ABCD
    CCC                         ABC
    DD                          AB
    E                           A
```

22. Write a C program to read and store the rollno and marks of 20students using array.

23. Write a program to findmaximumelementfrom1-Dimensionalarray.

24. Write a program to sort given array in ascending order.

25. Write a program to reverse string.

26. Write a program that defines a function to add first *n* numbers.

27. Writeafunctionintheprogramtoreturn1ifnumberisprimeotherwisereturn0

28. Write a program to find factorial of a number using recursion.

29. Write a function that will scan a character string passed as an argument and convert all lowercase character into their upper case equivalents

30. Define a structure type *struct* personal that would contain person name, joining date and salary using this structure to read this information of 5 people and print the same on screen.

31. Define structure data type called time_struct containing three member's integer hour, integer minute and integer second. Develop a program that would assign values to the individual number and display the time inthefollowingformat:16:40:51

32. Design a structure student record to contain name, branch and total mark obtained. Develop aprogramtoreaddatafor10studentsinaclass and print them.

33. Write a program to print address of variable using pointer.

34. Write a C program to swap the two values using pointers.

35. Write a program to access elements using pointer.

36. Write a program to read, print and addition of two Matrices using pointer and user define functions.

37. Write a program for sorting using pointer.

38. Write a program to read n integer number from keyboard and store them into a file All.txt. Read All.txt file, separate even and odd numbers and store them into files Even.txt and Odd.txt respectively and display contents of all the three files.

39. Write a program to accept the contents from the user and store it in the file one line at a time and print the contents of the file.

40. Write a program to merge two files into the third file.

CO- PO Mapping:

Semester-1	Fundamental of Programming (10610102)										
	POs										
Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	3	3	-	3	-	-	-	-	-	-
CO2	3	3	3	-	3	-	-	-	-	-	-
CO3	3	3	3	-	3	-	-	-	-	-	-
CO4	3	3	3	-	2	-	-	-	-	-	-
CO5	3	3	3	3	3	-	-	-	-	-	-
Ave	3.00	3.00	3.00	3.00	2.80	-	-	-	-	-	-

Legend: '3' for high, '2' for medium, '1' for low and '-' for no correlation of each CO with PO.

* * * * * * *