



GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Engineering

Level: Degree

Branch: ALL

Course / Subject Code: BE01000121

Course / Subject Name: Programming for Problem Solving

w. e. f. Academic Year:	2024-25
Semester:	1 st Year
Category of the Course:	ESC

Prerequisite:	<ul style="list-style-type: none">• Basic Computer Skills• Logical Thinking and Problem-Solving Abilities• Understanding of mathematical logic and problem-solving strategies
Rationale:	The "Programming for Problem Solving" course is a foundational skill that significantly enhances an engineer's problem-solving abilities, career prospects, and capacity for innovation. Integrating programming into the curriculum ensures that engineering programs prepare students to meet the challenges of the modern technological landscape and contribute effectively to their respective fields.

Course Outcome:

After Completion of the Course, Student will able to:

No	Course Outcomes	RBT Level
01	Identify fundamental programming constructs such as variables, data types, operators, expressions, control structures, functions and basic data structures	Remember
02	Explain the principles of programming and software development, including the structure and operation of algorithms, flowcharts, and pseudocode.	Understand
03	Develop simple programs using appropriate data structures and standard libraries.	Apply
04	Apply programming constructs such as loops, conditional statements, and functions to solve basic engineering problems.	Apply
05	Debug and troubleshoot programming errors by systematically testing and refining code.	Analyze



GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Engineering

Level: Degree

Branch: ALL

Course / Subject Code: BE01000121

Course / Subject Name: Programming for Problem Solving

Teaching and Examination Scheme:

Teaching / Learning Scheme (in Hours per semester)					Total Credits = TH/30	Assessment Pattern and Marks					Total Marks
L	T	P	PBL*	TH		Theory		Tutorial / Practical			
						ESE (E)	PA (M)	PA/ (I)	TW/ SL (I)	ESE (V)	
30	00	60	30	120	04	70	30	20	30	50	200

Where L = Lecture, T= Tutorial, P= Practical, TW/SL = Term-Work / Self-Learning, TH = Total Hours, PA = Progressive Assessment, ESE = End-Semester Examination

* Problem Based Learning (PBL) aims to accommodate learning beyond syllabus as per clause 9.4 of NBA manual.

Course Content:

Unit No.	Content	No. of Hours	% of Weightage
1.	Introduction to Programming <ul style="list-style-type: none"> Basics of Computers: Overview of computer systems, hardware, and software. Introduction to Programming Languages: Types of programming languages, language translators (compilers, interpreters). Flowcharts and Algorithms: Basics of creating flowcharts and writing algorithms for problem-solving. 	03	10
2.	Basics of C Programming <ul style="list-style-type: none"> Structure of a Program: Basic syntax, semantics, and structure of C programs. Data Types and Variables: Primitive data types, variables, and constants. Operators: Arithmetic, relational, logical, bitwise, and assignment operators. 	03	10
3.	Control Structures <ul style="list-style-type: none"> Conditional Statements: if, if-else, nested if-else, switch-case. Loops: for, while, do-while loops. Break and Continue Statements: Usage in loop control. 	03	10
4.	Functions and Modular Programming <ul style="list-style-type: none"> Defining Functions: Syntax, return types, and parameter passing. Library Functions: Standard library functions and header files (for C). Recursion: Basic concepts and examples. 	03	10
5.	Arrays and Strings <ul style="list-style-type: none"> Arrays: One-dimensional and multi-dimensional arrays, array operations. 	03	10



GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Engineering

Level: Degree

Branch: ALL

Course / Subject Code: BE01000121

Course / Subject Name: Programming for Problem Solving

	<ul style="list-style-type: none">Strings: String handling functions, basic string operations.		
6.	Pointers <ul style="list-style-type: none">Pointers: Basics of pointers, pointer arithmetic, pointers and arrays, pointers to functions (for C).	03	10
7.	Structures and Unions <ul style="list-style-type: none">Structures: Defining and using structures, array of structures, pointer to structures.Unions: Basics and usage.	03	10
8.	File Handling <ul style="list-style-type: none">File Operations: Opening, closing, reading, writing files.File Pointers: File pointers and basic file operations (for C).	03	10
9.	Dynamic Memory Allocation <ul style="list-style-type: none">Memory Allocation: malloc, calloc, realloc, and free functions. Usage and examples.	03	10
10.	Debugging and Testing <ul style="list-style-type: none">Debugging Techniques: Common debugging practices, use of debugging tools.Testing: Writing test cases, unit testing.	03	10
	Total	30	100

Suggested Specification Table with Marks (Theory):

Distribution of Theory Marks (in %)					
R Level	U Level	A Level	N Level	E Level	C Level
20	20	40	20	00	00

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

References/Suggested Learning Resources:

(a) Books:

- C Programming Language by Brian W. Kernighan and Dennis M. Ritchie, Latest Edition (for C programming)
- Let Us C by Yashavant Kanetkar, Latest Edition (for C programming)
- Problem Solving and Program Design in C by Jeri R. Hanly and Elliot B. Koffman, Latest Edition (for C programming)



GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Engineering

Level: Degree

Branch: ALL

Course / Subject Code: BE01000121

Course / Subject Name: Programming for Problem Solving

(b) Open-source software and website:

1. OnlineGDB : [GDB online Debugger | Compiler - Code, Compile, Run, Debug online C, C++ \(onlinegdb.com\)](http://onlinegdb.com)
2. Compiler Explorer (Godbolt): [Compiler Explorer \(godbolt.org\)](http://godbolt.org)
3. JDoodle: [JDoodle - Free online cloud coding platform IDE to practice, teach and learn programming](http://jdoodle.com)

Suggested Course Practical List:

1. Write a program that performs basic arithmetic operations (addition, subtraction, multiplication, and division) and demonstrates the use of different data types.
2. Create a program that uses if, else, and switch statements to implement a simple menu-driven application. Use loops (for, while, and do-while) to repeat tasks.
3. Develop a program that calculates the factorial of a number using both iterative and recursive functions.
4. Write a program to perform various operations on arrays (e.g., sorting, searching) and strings (e.g., concatenation, comparison).
5. Implement a program that uses pointers to create and manipulate dynamic arrays, demonstrating the use of malloc, calloc, realloc, and free.
6. Design a student record system using structures that store and display information such as name, roll number, and grades.
7. Write a program to read from and write to files, such as creating a simple text editor that performs basic file operations.
8. Implement a singly linked list with operations like insertion, deletion, and traversal.
9. Develop programs to simulate stack operations (push, pop, peek) and queue operations (enqueue, dequeue) using arrays and linked lists.
10. Provide students with a program containing intentional errors and inefficiencies. Have them use debugging tools (like gdb) to find and fix the errors and optimize the code for better performance.

List of Laboratory/Learning Resources Required:

1. Computer System/Laptop with latest configurations.
2. Internet Connectivity



GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Engineering

Level: Degree

Branch: ALL

Course / Subject Code: BE01000121

Course / Subject Name: Programming for Problem Solving

Suggested Project List:

- Library Management System** with the Objectives of
 - Practice basic C syntax and operations,
 - Use control structures effectively,
 - Implement functions and modular programming,
 - Manipulate arrays and strings,
 - Handle file operations for data storage and retrieval
- Institute Hall Management System**, Project Objectives:
 - To manage the allocation of halls for different events.
 - To maintain records of bookings, cancellations, and availability of halls.
 - To provide an interactive user interface for managing hall reservations.

Suggested Activities for Students:

- To enhance the programming skills of students in a C Programming and Problem-Solving course, the instructor can assign various activities that progressively build their understanding and ability to apply concepts.

List of suggested activities for Problem Based Learning::

Sl. No.	Name of the activity	No. of hours	Evaluation Criteria
1.	Industry/Research laboratory visit	Visit = 5h, Report preparation = 5h Total = 10h	Based on report submitted. Report should contain observations and calculations based on industry/ lab data.
2.	Technical Video based learning related to the subject	Duration of video = 5h Report preparation = 5h Total = 10h	Report /presentation based on the video learning outcomes.
3.	Assignment writing. Numericals based assignment is preferable.	5 assignments of 2h each. Total = 10h	Based on the assignment submitted.
4.	Problem solving/Coding using C, C++, Python, SCILAB, MATLAB, MS-EXCEL or any other relevant software	5 small coding based assignment of 2h each. Total = 10h	Based on the coding solution submitted.
5.	Self learning on-line course	Minimum duration of the course should be 10h.	Examination based assessment at the end of course. Based on the certificate produced.
6.	Complex problem solving	Maximum 2 problem.	Based on the depth of the



GUJARAT TECHNOLOGICAL UNIVERSITY

Program Name: Engineering

Level: Degree

Branch: ALL

Course / Subject Code: BE01000121

Course / Subject Name: Programming for Problem Solving

		Study of the problem and solution finding, Total = 10h	solution submitted.
7	Videos on Industrial safety aspects based on subject	Duration of video = 5h Report preparation = 5h Total = 10h	Based on quiz/report submitted
8	Discussion on research paper based on relevant subject	5 research paper = 20 h	Summarize research paper and evaluation critical parameters
9.	Poster/chart/power point preparation on technical topics	Duration = 6 h	Based on poster/chart preparation and presentation skills
10	Working/non-working model on technical topics	Working = 12 h Non- working = 8 h	Based on inter department/external evaluation
11	Industrial exposure for 2-3 days to observe and provide tentative solutions on society/environment/health/any other issue	Duration = 15 h for industrial exposure Problem identification and tentative solution = 10 h Total = 20 h	Based on evaluation of critical problems and solutions
12	Group Discussion on emerging/trending technical topics based on subject	Duration = 1 h each	Based on performance in group discussion, technical depth, knowledge etc.
13.	Real world case studies based learning	Duration of data collection/study = 5h Report preparation = 5h Total = 10h	Based on in-depth study, technical depth, data collected, fact finding, etc.
14.	Application/Software development	Duration = 10 h	Depending on the complexity of the Application/Software

Note:

1. All the suggested activity should be related to the subject.
2. The number of hours are suggestive. Faculty can sub-divide the number of hours based on the activity. However, total number of hours is fixed.
3. Rubrics for the evaluation can be prepared by the faculty.
