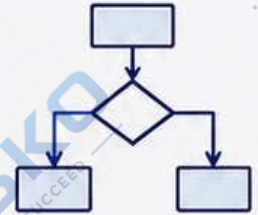


1010101010
0101010101
1010101010
0101010101



Question Bank

10110
01001
10101



Programming with C

Unit I - Fundamentals of C Programming

● Answers in one or two lines (1 mark questions)

1. Define an algorithm.
2. What is a flowchart?
3. What is a programming language?
4. Name the three types of programming languages.
5. Who developed the C programming language?
6. In which year was C language developed?
7. Expand ANSI.
8. What is the purpose of the C language?
9. What is the structure of a C program?
10. What is a character set in C?
11. Define an identifier.
12. What is a keyword?
13. How many keywords are available in C?
14. What is a variable?
15. What is a constant?
16. Define symbolic constant.
17. What is a data type?
18. What is type conversion?
19. What is an operator?
20. What is an expression?

● Short Answer Questions (3 marks questions)

1. Explain the algorithm with an example.
2. Explain the characteristics of a good algorithm.
3. Explain a flowchart with suitable symbols.
4. Draw the standard flowchart symbols and explain their uses.
5. Explain different types of programming languages.
6. Explain the advantages of the C programming language.
7. Explain the character set used in C.
8. Differentiate between keywords and identifiers
9. Explain variables and constants with examples.
10. Explain symbolic constants with syntax.
11. Explain arithmetic, relational and logical operators

- **Long Answer Questions (6 marks questions)**

1. Explain different types of programming languages with their advantages and disadvantages.
2. Describe the history, features, and advantages of C programming language.
3. Explain the structure of a C program with a neat example.
4. Explain all types of operators in C with suitable examples.
5. Write an algorithm and flowchart to calculate the sum of two numbers.
6. Draw a flowchart and write an algorithm to find the largest of two numbers.

Unit II - I/O Functions and Control Structure

● Answers in one or two lines (1 mark questions)

1. What is formatted input/output in C?
2. What is unformatted input/output?
3. What is the purpose of scanf()?
4. What is the purpose of printf()?
5. What is getchar()?
6. What is getch()?
7. What is gets()?
8. What is putchar()?
9. What is puts()?
10. What is a control structure?
11. Define branching statement.
12. What is an if statement?
13. What is an if-else statement?
14. What is a nested if?
15. What is a switch statement?
16. What is a loop?
17. Name the three looping statements in C.
18. What is the use of the break statement?
19. What is the use of the continue statement?
20. What is the comma operator?

● Short Answer Questions (3 marks questions)

1. Explain formatted input and output functions in C.
2. Differentiate between scanf() and printf().
3. Explain the functions getchar() and putchar().
4. Differentiate between getch() and getchar().
5. Explain branching statements in C.
6. Explain the if statement with an example.
7. Explain the if-else statement with syntax and example.
8. Explain nested if with an example.
9. Explain the switch statement with syntax.
10. Differentiate between if-else and switch.
11. Explain the while loop with an example.
12. Explain the do-while loop with an example.
13. Explain the for loop with syntax and example.
14. Explain the use of break, continue, goto, and the comma operator.

- **Long Answer Questions (6 marks questions)**

1. Explain formatted and unformatted input/output functions in C with suitable examples.
2. Explain all decision-making (branching) statements in C with syntax and examples.
3. Explain the if, if-else, nested if, and switch statements with suitable programs.
4. Explain all looping statements (while, do-while, and for) with syntax, flowchart, and examples.
5. Differentiate between while, do-while, and for loops with suitable examples.
6. Explain the use of break, continue, and goto statements with programs.
7. Compare if-else and switch statements with suitable examples.
8. Write a detailed note on control structures in C.
9. Explain the role of looping statements in problem solving with suitable examples.
10. Explain formatted and unformatted I/O functions along with their advantages and limitations.

Unit III- Arrays , Structure and Pointers

- **Answers in one or two lines (1 mark questions)**

1. Define an array.
2. What is a one-dimensional array?
3. What is a two-dimensional array?
4. What is array initialization?
5. What is a structure in C?
6. What is a union?
7. What is a pointer?
8. What is a pointer variable?
9. What is the address operator in C?
10. What is pointer initialization?
11. What is an array of structures?
12. What is a nested structure?
13. State one advantage of structures.
14. State one advantage of unions.
15. What is pointer arithmetic?
16. Which operator is used to access structure members?
17. Can a pointer store the address of another pointer?
18. Name any two applications of pointers.

- **Short Answer Questions (3 marks questions)**

1. What is Array ? Explain one-dimensional arrays with an example.
2. What is Array ? Explain two-dimensional arrays with an example.
3. Explain declaration and initialization of arrays.
4. What is Structure ? Explain the structure with syntax and examples.
5. Explain an array of structures.
6. Explain the declaration and initialization of pointers.
7. Explain pointer arithmetic with examples.

- **Long Answer Questions (6 marks questions)**

1. Explain one-dimensional and two-dimensional arrays with suitable programs.
2. Explain declaration, initialization, and operations on arrays with examples.
3. Explain pointers in C with declaration, initialization, and examples.
4. Explain pointer arithmetic with suitable examples.
5. Write a detailed note on pointers and their importance in C programming.

Unit IV- Functions and String Functions

● Answers in one or two lines (1 mark questions)

1. Define a function in C.
2. What is a function prototype?
3. What is a function call?
4. What is a function definition?
5. What is a formal argument?
6. What is an actual argument?
7. Define call by value.
8. Define call by reference.
9. What is a local variable?
10. What is a global variable?
11. What is recursion?
12. What is a recursive function?
13. What is the purpose of the return statement?
14. What is the use of the strlen() function?
15. What is the use of the strcpy() function?
16. What is the use of the strcmp() function?
17. What is the use of the strcat() function?

● Short Answer Questions (3 marks questions)

1. Explain functions in C with syntax.
2. Explain the function prototype with an example.
3. Explain different categories of functions in C.
4. Differentiate between actual and formal arguments.
5. Differentiate between call by value and call by reference.
6. Explain local and global variables.
7. Explain recursion with an example.
8. Explain function calling in C.
9. Explain functions with array arguments.
10. Explain the strlen() function with an example.
11. Explain the strcpy() function with an example.
12. Explain the strcmp() function with an example.
13. Explain the strcat() function with an example.
14. Explain function parameters and return values.

- **Long Answer Questions (6 marks questions)**

1. Explain functions in C with suitable syntax, examples, and advantages.
2. Explain all categories of functions with suitable examples.
3. Differentiate between call by value and call by reference with programs.
4. Explain actual arguments, formal arguments, local variables, and global variables with examples.
5. Explain recursion in C with suitable programs.
6. Explain functions with arrays using suitable examples.
7. Explain the string functions `strlen()`, `strcpy()`, `strcmp()`, and `strcat()` with syntax, examples, and outputs.