SRM VALLIAMMAI ENGINEERING COLLEGE

SRM Nagar, Kattankulathur – 603 203

(AN AUTONOMOUS INSTITUTION)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

QUESTION BANK

II SEMESTER

1901006-Programming in C

Regulation – 2019

Academic Year 2022 – 2023
(EVEN Semester)

Prepared by

Dr. S.K.Saravanan, AP (Sel.G) / CSE	Dr. S. Parthasarathy, AP(Sel.G) / CSE
Mrs. G. Sathya, AP(O.G) / CSE	

SRM VALLIAMMAI ENGINEERING COLLEGE

SRM Nagar, Kattankulathur – 603 203.

DEPARTMENT OF GENERAL ENGINEERING

QUESTION BANK

SUBJECT : **Programming in C**

SEM / YEAR: Second Semester / First Year

<table>
<tr><td colspan="4">UNIT I - BASICS OF C PROGRAMMING</td></tr>
<tr><td colspan="4">SYLLABUS</td></tr>
<tr><td colspan="4">Introduction to algorithm: Flowchart-Pseudo code- Introduction to programming paradigms- C programming: Data Types -Keywords-Variables and Constants–Operators and Expressions: Expressions -precedence, -associativity-Input/Output Statements-Decision making and looping: Branching statement, Iterative statement - Compilation process.</td></tr>
<tr><td colspan="4">PART – A</td></tr>
<tr><td>Q.No</td><td>Questions</td><td>BT Level</td><td>Competence</td></tr>
<tr><td>1.</td><td>Define an Algorithm</td><td>BTL -1</td><td>Remember</td></tr>
<tr><td>2.</td><td>What are the characteristic of an algorithm?</td><td>BTL -1</td><td>Remember</td></tr>
<tr><td>3.</td><td>Distinguish between an algorithm and a flowchart</td><td>BTL -2</td><td>Understand</td></tr>
<tr><td>4.</td><td>List the various symbols used to draw flowchart.</td><td>BTL -1</td><td>Remember</td></tr>
<tr><td>5.</td><td>Write a pseudo code to find maximum of three numbers.</td><td>BTL -1</td><td>Remember</td></tr>
<tr><td>6.</td><td>List the various keywords used to write pseudo code.</td><td>BTL -1</td><td>Remember</td></tr>
<tr><td>7.</td><td>Discuss the concept of the programming paradigm.</td><td>BTL -2</td><td>Understand</td></tr>
<tr><td>8.</td><td>What is meant by a data type? Give its classifications.</td><td>BTL –1</td><td>Remember</td></tr>
<tr><td>9.</td><td>Analyze why keywords are not to be used as identifiers.</td><td>BTL -4</td><td>Analyze</td></tr>
<tr><td>10.</td><td>Differentiate between variable and constant.</td><td>BTL -4</td><td>Analyze</td></tr>
<tr><td>11.</td><td>Write the use of ternary or conditional operator.</td><td>BTL - 3</td><td>Apply</td></tr>
<tr><td>12.</td><td>Evaluate the following C expression:
A = 6+10**2/50-2+4</td><td>BTL -6</td><td>Create</td></tr>
<tr><td>13.</td><td>Summarize the various types of C operators.</td><td>BTL - 5</td><td>Evaluate</td></tr>
<tr><td>14.</td><td>Analyze Increment and Decrement Operators with an example.</td><td>BTL - 4</td><td>Analyze</td></tr>
</table>

15.	Generalize the types of I/O statements available in 'C'.	BTL -6	Create
16.	Show the differences between while and do-while statements.	BTL -3	Apply
17.	Differentiate the switch and nested-if statement.	BTL -2	Understand

18.	Invent what type of loop and how?
	#include <stdio.h>
	int main () {
	for(; ;) {
	printf("This loop will run forever.\n");
	   }
	   return 0;
	}
		BTL -5	Evaluate

19.	Show the general form of if – else – if statement.	BTL -3	Apply
20.	Describe the order of precedence of operators.	BTL - 3	Apply
21.	Analyze the significance of Sizeof operator.	BTL - 4	Analyze
22.	Write a program to determine whether a person is eligible to vote.	BTL -5	Evaluate
23.	Provide the significance of break statement in loops.	BTL -2	Understand
24.	Discuss the compilation process	BTL -2	Understand
	PART - B.		
1.	(i). Explain the characteristics and need of an algorithm. (ii). Write an algorithm to find the first N natural numbers. (6 + 7)	BTL -1	Remember
2.	(i). Draw a flowchart to find the max of three numbers and explain. (7) (ii). Write the pseucode for finding simple interest. (6)	BTL -3	Apply
3.	Explain the importance of pseudo code with its keywords and discuss the advantages and disadvantages. (13)	BTL -4	Analyze
4.	Describe the structure of a C program with an example. (13)	BTL -1	Remember
5.	Illustrate about the various data types in 'C' and write a C program to find the sum of 10 non-negative numbers entered by the user. (13)	BTL-3	Apply

6.	Explain the following:
	    i. Keywords (4)
	    ii. C character set (4)
	    iii. Constants (5).
		BTL -5	Evaluate

7.	(i). Explain the different types of operators used in 'C'. (ii)Write a C program to check the integer is Palindrome or not. (7 + 6)	BTL -1	Remember
8.	Discuss in detail about various operators used in C and its precedence with an example. (13)	BTL -2	Understand
9.	Write a C program for the following: (i). To check whether a given year is leap or not. (5) (ii). To find the roots of a quadratic equation. (8)	BTL -2	Understand

10.	Develop a C program for the following: (i). To find the area and circumference of a circle with radius r. (6) (ii). To find the sum of first 100 integers. (7)	BTL -6	Create
11.	Write a C program for the following: (i). To find the sum of the digits of a number. (123 =>1+2+3=6.) (7) (ii).To find the sum of all odd / even numbers between 1 and100. (6)	BTL -4	Analyze
12.	Write a C program for the following: (i). To generate the first n numbers in a Fibonacci series. (7) To find the factorial of a given number. (6)	BTL -1	Remember
13.	Discuss the various Conditional Branching Statements used in C with its syntax and flow diagram.	BTL -2	Understand
14	Explain the importance of the following loop control statements with example. (i). the break statement (5) (ii). the continue statement (4) (iii). the goto statement (4)	BTL -3	Apply
15.	Explain in detail about the variables, constants and its types used in C	BTL -5	Evaluate
16.	Write the various Input and Output statements used in C with example. (13)	BTL -4	Analyze
17.	Explain in detail the While and Do…While looping statements in C with suitable examples. (13)	BTL -2	Understand

PART-C			
1.	Develop a C program for the following: (i) To check whether a number is prime or not. (8) (ii). To convert the temperature given in Fahrenheit to Celsius and vice versa. (7)	BTL -6	Create
2.	Compare and contrast branching and looping statements used in C. (15)	BTL -5	Evaluate
3.	Develop a C Program to find the power of a Number Using the while loop and using pow () function. (15)	BTL -6	Create
4.	Develop a menu driven program in C to perform various arithmetic operations. (15)	BTL -6	Create
5.	Summarize the algorithm, flowchart and pseudo code with an example. (15)	BTL -5	Evaluate

SUBJECT   : Programming in C

SEM / YEAR: Second Semester / First Year

	UNIT II - ARRAYS AND STRINGS		
	SYLLABUS		
	Introduction to Arrays: One dimensional array: Assigning an array to another array –Equating an array with another array-Two dimensional Arrays: Declaration-usage of two-dimensional array-reading, storing and accessing elements in two dimensional array-memory representation-String operations: String library functions- list of strings-command line.		
	PART - A		
Q.No	**Questions**	**BT Level**	**Competence**
1.	What is an array? Write the syntax for array.	BTL -1	Remember
2.	Define a float array of size 5 and assign 5 values to it.	BTL -1	Remember
3.	Why are arrays needed?	BTL -4	Analyze
4.	List out the advantages of Arrays.	BTL -1	Remember
5.	What will happen when you access the array more than its dimension?	BTL -1	Remember
6.	How to create a two-dimensional array?	BTL -2	Understand
7.	Point out an example code to express two-dimensional array.	BTL -4	Analyze
8.	What is the method to equate an array?	BTL -3	Apply
9.	What are the different ways of initializing array?	BTL -2	Understand
10.	Distinguish between one dimensional and two-dimensional arrays.	BTL -4	Analyze
11.	What is the use of '\0' and '%s'?	BTL -1	Remember
12.	Is address operator used in scanf( ) statement to read an array? Why?	BTL -6	Create
13.	What is the role of strrev( ) function?	BTL -3	Apply
14.	Show a C function to compare two strings.	BTL -3	Apply
15.	How to initialize a string? Give an example.	BTL -6	Create

No.	Question	BTL	Level

16.	What will be the output of following program?
	``c
	#include <stdio.h>
	int main ()
	{
	char str[8]="SRMVALLIAMMAI";
	printf("%s",str);
	return 0;
	}
	``
		BTL -2	Understand

17.	Write the output of the following Code:
	``c
	main()
	{
	    char x;
	    x = 'a';
	    printf("%d\n", x);
	}
	``
		BTL -5	Evaluate

18.	Write the list of operations on arrays.	BTL -3	Apply
19.	How can putchar( ) be used to print a string?	BTL -2	Understand
20.	How is an array represented in memory?	BTL -2	Understand
21.	If str[ ] = "Welcome to the world of programming", then SUBSTRING(str, 15, 5) = ?	BTL -4	Analyze
22.	Specify any two applications of Array.	BTL -5	Evaluate
23.	List out the any four functions that are performed on character strings.	BTL -1	Remember

24.	Write the output of the following Code:
	``c
	main( )
	{
	    static char name[ ] = "SRMVEC";
	    int i = 0;
	    while(name[i]!='\0')
	    {
	        printf("%c",name[i]);
	        i++;
	    }
	}
	``
		BTL -5	Evaluate

	PART -B		
1.	(i). Write a C Program to take 5 values from the user and store them in an array. (6) (ii). Write a C program to re-order a one-dimensional array of numbers in descending order. (7)	BTL -1	Remember
2.	(i). How to access elements out of its bound explain with an example. (5) (ii). Write C program to count the array elements using sizeof( ) operator. (8)	BTL-5	Evaluate
3.	Write a C program to insert a number at a given location in an array. (13)	BTL -1	Remember
4.	Explain with an example how to copy all elements of an array into another array. (13)	BTL -4	Analyze
5.	Describe the following with suitable examples. (i) Initializing a 2 Dimensional Array (6) (ii) Memory Map of a 2 Dimensional Array. (7)	BTL -2	Understand

6.	Write a C program for transpose of a matrix. (13)	BTL -1	Remember
7.	Write a C program to add two 3 x 3 matrices. (13)	BTL -4	Analyze
8.	Write a C program to sort the n numbers using selection sort. (13)	BTL -2	Understand
9.	Develop a C program to search an element from the array. (13)	BTL -6	Create
10.	Develop a C program to delete an element from the array. (13)	BTL -3	Apply
11.	Write a C program to sort names of students in a class. (13)	BTL-5	Evaluate
12.	List and explain the functions used for reading and writing strings. (13)	BTL -2	Understand
13.	Explain about the Arrays of Strings and its manipulation in detail. (13)	BTL -1	Remember
14.	Write a C program to find whether the given string is palindrome or not without using string functions. (13)	BTL -3	Apply
15.	Write the C program to find the number of Vowels, Consonants, Digits and white space in a string. (13)	BTL -2	Understand
16.	Illustrate with an example of command line arguments. (13)	BTL -3	Apply
17.	Explain about various string handling functions with an example. (13)	BTL -4	Analyze
	PART-C		
1.	Write a C program to find average marks obtained by 30 students in a test. (15)	BTL -6	Create
2.	Explain the various operations on two- dimensional Array. Also write a program to do the matrix multiplication.	BTL -6	Create
3.	Write a C program to merge two unsorted arrays (15)	BTL -6	Create
4.	(i) Write a C program to strcpy ( ) function. (7) (ii) Compare and contrast gets() and puts(). (8)	BTL -5	Evaluate
5.	Write a C program to check whether a given number is Armstrong number or not using command line argument. (15)	BTL -5	Evaluate

SUBJECT   : Programming in C

SEM / YEAR : Second Semester / First Year

UNIT III - FUNCTIONS		
SYLLABUS		
Introduction to functions: Classification of functions- function definition-function call -function with inputs and outputs - recursive function - library functions - scope of variables.		
PART - A		
Q.No	**Questions**	**BT Level**	**Competence**

Q.No	Questions	BT Level	Competence
1.	Define function with its syntax.	BTL -1	Remember
2.	What are the various parts of functions?	BTL -1	Remember
3.	Express the difference between function declaration and function definition.	BTL -2	Understand
4.	What is function call?	BTL -1	Remember
5.	Write any two applications of recursive function.	BTL -1	Remember
6.	Differentiate between call by value and call reference.	BTL -4	Analyze
7.	Why is scope of variable necessary in function?	BTL -2	Understand
8.	Write a function in C to find the square root of any number.	BTL -1	Remember
9.	Point out the meaning of user-defined function.	BTL -4	Analyze
10.	What is meant by library function?	BTL -4	Analyze
11.	Develop no argument and no return value in a function.	BTL -6	Create

12.	Determine output:
	``c
	main( )
	{
	int i = abc(10);
	printf("%d", - - i);
	}
	int abc(int i)
	{
	    return(i++);
	}
	``
		BTL -3	Apply

13.	What will be output of the following?
	``c
	#include <stdio.h>
	int incr (int i)
	{
	   static int count = 0;
	   count = count + i;
	   printf("count=%d\n",count);
	}
	``
		BTL -3	Apply

1.	void main( )
	{
	int i, j;
	for (i = 0; i<=4; i++)
	    j = incr(i);
	}		

14.	Invent the output of the following code:
	#include<stdio.h>
	int A=2;
	int B=3;
	int Add ( )
	{
	    return A + B;
	}
	int main( )
	{
	   int answer;
	   A = 5;
	   B = 7;
	   answer = Add( );
	   printf("%d\n",answer);
	   return 0;
	}
		BTL -6	Create

15.	What is function prototype?	BTL -2	Understand
16.	Mention the need for 'return' statement in function.	BTL -3	Apply
17.	List any 6 functions used in math library.	BTL -4	Analyze
18.	Write a program to calculate GCD using recursive functions.	BTL -5	Evaluate
19.	What is a recursive function?	BTL -2	Understand
20.	Specify the need for function.	BTL -5	Evaluate

21.	Point out the error in the program
	#include<stdio.h>
	int main()
	{
	   int a=10;
	   void f( );
	   a = f( );
	   printf("%d\n", a);
	   return 0;
	}
	void f( )
	{
	   printf("a");
	}
		BTL -5	Evaluate

22.	Differentiate between formal parameters and actual parameters.	BTL -1	Remember
23.	Narrate how to apply user-defined function.	BTL -3	Apply
24.	Mention the advantage of pass by reference.	BTL -2	Understand
PART – B			
1.	Describe about user defined function and predefined function with an example. (13)	BTL -1	Remember
2.	Write a code in C to get the smallest element of an array using function. Analyze the code with sample input	BTL -4	Analyze

	34, 2,6,11 and 46. (13)		
3.	Apply recursive function in C for reverse a sentence. (13)	BTL -3	Apply
4.	Discuss about the classification of functions depending upon their inputs and output (parameters). (13)	BTL -2	Understand
5.	Explain in detail about Pass by Value and Pass by reference. (13)	BTL -1	Remember
6.	Discuss about passing arrays to function. (13)	BTL -2	Understand
7.	Explain in detail about recursive function with sample code. (13)	BTL -5	Evaluate
8.	Analyze with example code in C for global and local variables. (13)	BTL -4	Analyze
9.	Write notes on fixed argument functions and variable argument functions. (13)	BTL -1	Remember
10.	Write a program to sum the following series : $1/1! + 1/2! + 1/3! + \ldots\ldots+ 1/n!$	BTL -3	Apply
11.	Write a menu driven program in C for calculator using built-in functions. (13)	BTL -4	Analyze
12.	Write a function in C to find the greatest among three numbers. (13)	BTL -5	Evaluate
13.	What is function? Why we need function? Explain the function declaration and definition with its syntaxes.	BTL -2	Understand
14.	(i). Write a program to find whether a number is divisible by two or not using functions. (6) (ii). Write a program to find the simple interest using functions. (7)	BTL -3	Apply
15.	Write a C program for cube of a number using pass by reference. (13)	BTL -1	Remember
16.	Write a C program to sort the given N names using function. (13)	BTL -2	Understand
17.	Explain about any 4 library functions in C with an example. (13)	BTL -6	Create
PART-C			
1.	Develop a C program for binary search using recursive function. (15)	BTL -6	Create
2.	Examine with example program to display all prime numbers between two intervals using functions. (15)	BTL -5	Evaluate
3.	(i). Write a program to calculate area of a circle using function. (7) (ii). Write a program to convert time to minutes using functions. (8)	BTL -5	Evaluate
4.	Develop a C program for towers of Hanoi using recursive function. (15)	BTL -6	Create
5.	Illustrate the C coding for swapping of two numbers using pass by value and pass by reference. (15)	BTL -5	Evaluate

SRM VALLIAMMAI ENGINEERING COLLEGE

SRM Nagar, Kattankulathur – 603 203.

DEPARTMENT OF GENERAL ENGINEERING
QUESTION BANK

SUBJECT : **Programming in C**

SEM / YEAR: Second Semester / First Year

UNIT-IV- STRUCTURES AND UNIONS
SYLLABUS **Introduction to Structures: Array of structures – Nested structure-functions and Structures-Introduction to union-: practical applications of union —typedef and structures-enumerated data type.**

PART – A

Q.No	Questions	BT Level	Competence
1.	What is structure?	BTL -1	Remember
2.	Where is Union used in C?	BTL -1	Remember
3.	How the members of structure object is accessed?	BTL -2	Understand

4.	How many bytes in memory taken by the following C structure?
	#include <stdio.h>
	struct test
	{ int k;
	    char c;
	  };
		BTL -1	Remember

5.	What is a nested structure?	BTL -1	Remember
6.	How typedef is used in structure?	BTL -4	Analyze
7.	Define the term Union in C.	BTL -2	Understand

8.	What is the output of the following program?
	#include<stdio.h>
	void main ( )
	{
	enum days {MON=-1, TUE, WED=4, THU,FRI,SAT};
	printf("%d, %d, %d, %d, %d, %d", MON, TUE, WED, THU, FRI, SAT);
	}
		BTL -1	Remember

9.	Point out the meaning of array of structures.	BTL -4	Analyze
10.	Examine the output of the following program: #include<stdio.h> void main ( ) { enum status {pass, fail, absent};	BTL -4	Analyze

	enum status stud1, stud2, stud3;
	  stud1 = pass;
	  stud2 = absent;
	 stud3 = fail;
	 printf("%d %d %d\n", stud1, stud2, stud3);
	}		

11.	Specify the use of typedef.	BTL -3	Apply

12.	Determine the output of the following program:
	      #include<stdio.h>
	      void main ()
	      {
	         union a
	         {
	              int i;
	              char ch[2];
	         };
	         union a u;
	         u.ch[0]=3;
	         u.ch[1]=2;
	         printf("%d, %d, %d\n", u.ch[0], u.ch[1], u.i);
	      }
		BTL -6	Create

13.	Discover the meaning of enum.	BTL -3	Apply
14.	Differentiate Structures with Arrays.	BTL -3	Apply

15.	Invent the application of size of operator to this structure.
	Consider the declaration:
	struct
	{
	    char name;
	    int num;
	}student;
		BTL -6	Create

16.	How the structure members are initialized?	BTL -2	Understand
17.	Differentiate between structure and a union.	BTL -3	Apply
18.	How members of a union are accessed?	BTL -4	Analyze
19.	What do you mean by self – referential structure?	BTL -5	Evaluate
20.	Is it mandatory that the size of all elements in a union should be same?	BTL -2	Understand
21.	Develop a structure namely Book and create array of Book structure with size 10.	BTL -5	Evaluate

22.	What will be the output of the C program?
	#include<stdio.h>
	int main ()
	{
	    enum numbers {n1 = 1.5, n2 = 0, n3, n4, n5, n6};
	    printf("%d %d\n", n1, n2);
	}
		BTL -5	Evaluate

23.	What is the difference between enum and macro?	BTL -1	Remember
24.	Can we declare function inside structure of C programming? Justify.	BTL -2	Understand

	PART – B		
1.	Describe about the functions and structures. (13)	BTL -1	Remember
2.	Explain about the structures and its operations. (13)	BTL -2	Understand
3.	Examine the differences between nested structures and array of structures. (13)	BTL -3	Apply
4.	Write a C program using structures to prepare the students mark statement. (13)	BTL -2	Understand
5.	Write a C program to enter two points and then calculate the distance between them. (13)	BTL -2	Understand
6.	Write a C program to read the details of book name, author name and price of 200 books in a library and display the total cost of the books and the book details whose price is above Rs.500. (13)	BTL -1	Remember
7.	(i). Express a structure with data members of various types and declare two structure variables. Write a program to read data into these and print the same. (8) (ii). Justify the need for structured data type. (5)	BTL -3	Apply
8.	(i) Does structure bring additional overhead to a program? Justify. (7) (ii)Write short note on structure declaration. (6)	BTL -1	Remember
9.	(i). How to access enumerated data type and explain with an example program. (7) (ii) Create enum of week days. Write a program in C, use this enum and display it. (6)	BTL -6	Create
10.	Explain with an example the self-referential structure. (13)	BTL -4	Analyze
11.	Explain nested structure and write C Program to implement the same. (13)	BTL -5	Evaluate
12.	Explain passing structures to a function with respect to the following: (i). Passing individual members. (ii). Passing entire structure.	BTL -3	Apply
13.	Explain passing structures to a function with respect to passing the address of the structure.	BTL -5	Evaluate
14.	Define Union. Describe how to declare, initialize and access members of Union with a programming example. (13)	BTL -2	Understand
15.	Compare with example code for Structure and Union. (13)	BTL -4	Analyze
16.	Illustrate a C program to store the employee information using structure and search a particular employee details. (13)	BTL -1	Remember

17.	Define a structure called student that would contain name, regno and marks of five subjects and percentage. Write a program to read the details of name, regno and marks of five subjects for 30 students, calculate the percentage and display the name, regno, marks of the subjects and percentage of each student. (13)	BTL -4	Analyze

PART-C

1	Write a structure to store the name, account number and balance of customers (more than 10) and store their information. Write a function to print the names of all the customers having balance less than Rs.200. (15)	BTL -6	Create
2	Write a C program using structures to prepare the employee pay roll of a company. (15)	BTL -5	Evaluate
3.	Write a program to read and display the information of all the students in a class using array of structures.	BTL -5	Evaluate
4.	Write a C program for passing structures as function arguments and returning a structure from a function. (15)	BTL -5	Evaluate
5.	Develop a C Program to use the arrays inside union variables. (15)	BTL -6	Create

SUBJECT   : Programming in C

SEM / YEAR: Second Semester / First Year

UNIT-V- STORAGE CLASS AND PREPROCESSOR DIRECTIVE			
SYLLABUS **Introduction to storage classes: Types of storage classes- C preprocessor Directives: Types of preprocessor directives-Pragma Directive-conditional directive.**			
PART – A			
Q.No	**Questions**	**BT Level**	**Competence**
1.	Define storage class.	BTL -1	Remember
2.	Mention the different types of storage classes.	BTL -1	Remember
3.	Distinguish between auto and register storage class variables.	BTL -2	Understand
4.	How static variables differ with auto variables?	BTL -3	Apply
5.	Write a simple program for static variable.	BTL -4	Analyze

6.	What will be the answer for the following code:
	#include<stdio.h>
	int a;
	int main( )
	{
	    printf("a = : %d", a)
	    return 0;
	}
		BTL -5	Evaluate

7.	Where the storage class variables stored?	BTL -2	Understand
8.	What is meant by extern variable? Give an example.	BTL -1	Remember

9.	What is storage class for variable A in below code? Justify.
	int main ()
	{
	int A;
	A = 10;
	printf("%d", A);
	return 0;
	}
		BTL -1	Remember

10.	What will the SWAP macro in the following program be expanded to on preprocessing? Will the code compile?
	#include<stdio.h>
	#define SWAP (a, b, c) (c t; t=a, a=b, b=t)
	int main ()
	{
	int x=10, y=20;
	SWAP (x, y, int);
	printf("%d %d\n", x, y);
	return 0;
	}
		BTL -4	Analyze

11.	How are preprocessor directives written in C?	BTL -6	Create
12.	How can you avoid including a header more than once?	BTL -1	Remember
13.	What are the types of C preprocessor Directives?	BTL -4	Analyze
14..	Compare the pragma and conditional directive.	BTL -4	Analyze
15.	Examine the six pragma directives.	BTL -3	Apply
16.	Write the syntax of pragma directive.	BTL -2	Understand
17.	Is there any difference that arises if double quotes, instead of angular brackets are used for including the standard header file?	BTL -3	Apply
18.	Identify the use of pragma directive in c.	BTL -3	Apply
19.	List out the seven conditional directives in c.	BTL -1	Remember
20.	What is the use of #if directive?	BTL -2	Understand
21.	Write a note on define macro.	BTL -5	Evaluate
22.	Evaluate the advantages of a macro over a function.	BTL -5	Evaluate
23.	Develop an example for conditional compilation.	BTL -6	Create
24.	What does #undef, #pragma indicate in c?	BTL -2	Understand
PART – B			
1.	Describe about the register and static storage classes with example program. (13)	BTL -2	Understand

2.	Distinguish between the following macro.
	    (i). Object like macro. (7)
	    (ii). Function like macro. (6).
		BTL -2	Understand

3.	Illustrate and explain about unconditional preprocessor directive. (13)	BTL -3	Apply
4.	Explain about #if, #else, #elif directive with an example program. (13)	BTL -2	Understand
5.	Describe the defined operator and #error directive. (13)	BTL -1	Remember
6.	Define a macro in C to check whether a given three-digit number is an Armstrong number or not. Illustrate the use of this macro in a program. (13)	BTL -1	Remember

7.	Write about conditional preprocessor directive with an example. (13)	BTL -3	Apply
8.	Define a macro in C to check whether a given number is odd or even. Illustrate the use of this macro in a program. (13)	BTL -1	Remember
9.	Write about all pragma directives and explain in detail. (13)	BTL -6	Create
10.	Write the C coding for finding the average of number using any of the storage class declarations. (13).	BTL -4	Analyze
11.	Explain the various ways in which a source file inclusion directive can be written. (13)	BTL -5	Evaluate
12.	Examine the various stages a program undergoes before execution. (13)	BTL -4	Analyze
13.	File inclusion can be nested. Justify this statement with the help of a suitable example. (13)	BTL -3	Apply
14.	Comment on the restrictions imposed on the conditional expression of #if directive. (13)	BTL -2	Understand
15.	(i). What do you understand by the term preprocessor directive? (7) (ii). Can we have a C program that does not use any preprocessor directive? (6)	BTL -5	Evaluate
16.	Write a C Program to calculate the factorial of a number by using the keyword static. (13)	BTL -1	Remember

17.	Write a C Program to generate Fibonacci series by using keyword auto. (13)	BTL -4	Analyze
PART-C			
1	Describe all storage class variables with programming examples.	BTL -6	Create
2	Write preprocessor directives code in C for roots of a quadratic equation. (15)	BTL -5	Evaluate
3	Develop a C Program based on conditional directive to display the Distinction, First class and Second class based on the student mark is above 70, between 60 to 70 and between 40 to 60 respectively otherwise Fail. (15)	BTL -6	Create
4.	Summarize and Compare the preprocessor directives with all Pragma directives. Give example code.	BTL -6	Create
5.	Summarize of storage classes with respect to various parameters storage location, initial value, lifetime and linkage. (15)	BTL -5	Evaluate