Module-2: Introduction to programming

Que1. Research and provide three real-world applications where C programming is extensively used, such as in embedded systems, operating systems, or game development.

Ans.

→1. Embedded Systems:-

- Description: Embedded systems are specialized computing systems that perform dedicated functions within larger mechanical or electrical systems.
- Use of C: C is widely used because it allows direct manipulation of hardware resources and has minimal runtime overhead.
- Example:
    - Microcontrollers in household appliances (like washing machines or microwave ovens).

→2. Operating Systems:-

- Description: Operating systems manage computer hardware and software resources and provide common services for application programs.
- Use of C: Most modern operating systems are either written in C or have C at their core because it provides low-level access while maintaining some abstraction.
- Examples:

- Linux kernel
- Windows
- macOS and Unix variants

→3. Game Development (Game Engines and Tools):-

- Description: Game development involves creating software for video games, including rendering engines, physics simulations, and hardware interfacing.
- Use of C: C and C++ are used for performance-critical parts of game engines due to their speed and memory control.
- Examples:
  - Doom and Quake
  - Unreal Engine

Que2. Install a C compiler on your system and configure the IDE. Write your first

program to print "Hello, World!" and run it.

Ans.

```
#include<stdio.h>
int main(){
    printf("\n Hello world");
    return 0;
}
```

Que-3. Write a C program that includes variables, constants, and comments. Declare

and use different data types (int, char, float) and display their values.

Ans.

```
#include<stdio.h>
/* constants*/
#define c 10
int main(){
    printf("%d",c);

    int roll,std;
    float per;
    char grade;
    long int fees;
    printf("\n enter your roll no=");
    scanf("%d",&roll);
    printf("\n enter your standard=");
    scanf("%d",&std);
    printf("\n enter your percentage=");
    scanf("%f",&per);
    printf("\n enter your grade=");
    scanf(" %c",&grade);
```

```
    printf("\n enter your fees=");
    scanf("%ld",&fees);

    printf("\n roll no=%d",roll);
    printf("\n standard=%d",std);
    printf("\n percentage=%f",per);
    printf("\n grade=%c",grade);
    printf("\n fees=%ld",fees);

    return 0;
}
```

Que-4. Write a C program that accepts two integers from the user and performs

arithmetic, relational, and logical operations on them. Display the results .

Ans.

```
#include<stdio.h>
int main(){
    int num1,num2;
    printf("\n Enter value in number 1=");
    scanf("%d",&num1);
```

```
printf("\n Enter value in number 2=");
scanf("%d",&num2);
//arithmetic operations

printf("\n Addition of %d and %d is
=%d",num1,num2,num1+num2);
printf("\n Subtraction of %d and %d is =%d",num1,num2,num1-
num2);
printf("\n Multiplication of %d and %d is
=%d",num1,num2,num1*num2);
printf("\n Division  of %d and %d is
=%d",num1,num2,num1/num2);
printf("\n Remaindor of %d and %d is
=%d",num1,num2,num1%num2);
printf("\n");

//relation operation

printf("\n Result of %d>%d is=%d",num1,num2,num1>num2);
printf("\n Result of %d>=%d is=%d",num1,num2,num1>=num2);
printf("\n Result of %d<%d is=%d",num1,num2,num1<num2);
printf("\n Result of %d<=%d is=%d",num1,num2,num1<=num2);
printf("\n Result of %d==%d is=%d",num1,num2,num1==num2);
printf("\n Result of %d!=%d is=%d",num1,num2,num1!=num2);
```

```
	printf("\n ");


	//logical operator


	printf("\n Result of %d>=%d && %d<=%d =
%d",num1,num2,num1,num2,num1>=num2 && num1<=num2);
	printf("\n Result of %d>=%d || %d<=%d =
%d",num1,num2,num1,num2,num1>=num2 || num1<=num2);
	printf("\n Result of !(%d>=%d)=
%d",num1,num2,!(num1>=num2));


	return 0;
}
```

Que-5. Write a C program to check if a number is even or odd using an if-else

statement. Extend the program using a switch statement to display the month

name based on the user's input (1 for January, 2 for February, etc.).

Ans.

```
#include<stdio.h>
int main(){
	//number even or odd check
	int num;
	printf("\n Enter the value of Number=");
```

```
scanf("%d",&num);
if(num%2==0){
	printf("\n Number is Even.");

}
else{
	printf("\n Number is Odd.");
}

// Display the months
printf("\n");

int month;
printf("\n enter value for month=");
scanf("%d",&month);

switch(month){
	case 1:
		printf("\n Month='January'");
		break;
	case 2:
		printf("\n Month='February'");
		break;
```

```
case 3:
        printf("\n Month='March'");
        break;
case 4:
        printf("\n Month='April'");
        break;
case 5:
        printf("\n Month='May'");
        break;
case 6:
        printf("\n Month='June'");
        break;
case 7:
        printf("\n Month='July'");
        break;
case 8:
        printf("\n Month='August'");
        break;
case 9:
        printf("\n Month='September'");
        break;
case 10:
        printf("\n Month='October'");
```

```
            break;
        case 11:
            printf("\n Month='November'");
            break;
        case 12:
            printf("\n Month='December'");
            break;
        default:
            printf("\n Invalid Month.");
    }
    return 0;
}
```

Que-6. Write a C program to print numbers from 1 to 10 using all three types of loops

(while, for, do-while).

Ans.

```
#include<stdio.h>
int main(){
    //while loop
    printf("\n while loop:");
    int num=1,i;
    while(num<=10){
        printf("\n number=%d",num);
```

```
        num++;
    }
    //for loop

    printf("\n");
    printf("\n for loop :");
    num=10;
    for(i=1;i<=num;i++){
        printf("\n number=%d",i);
    }

    //do..while loop
        printf("\n");
    printf("\n do...while loop :");
    num=1;
    do
    {
printf("\nnumber=%d",num);
num++;
    }while(num<=10);


    return 0;
```

}

Que-7. Write a C program that uses the break statement to stop printing numbers

when it reaches 5. Modify the program to skip printing the number 3 using the

continue statement.

*/

Ans.

```
#include<stdio.h>
int main(){
	int number,i;
	printf("\n Enter the Value Of Number=");
	scanf("%d",&number);
	for(i=1;i<=number;i++){
		if(i==5){
			break;
		}
		else if(i==3){
			continue;
		}
		else{
			printf("\n number=%d",i);
		}
```

```
    }

    return 0;
}
```

Que-8. Write a C program that calculates the factorial of a number using a function. Include function declaration, definition, and call.

Ans.

```
#include<stdio.h>
int fact();//declaration
int main(){
    int result;
    result=fact();//function call
    printf("factorial of given number=%d",result);

    return 0;
}
int fact()//definition
{
    int num,fact=1,i;
    printf("\n enter the number=");
    scanf("%d",&num);
    for(i=1;i<=num;i++){
    fact=fact*i;
```

```
    }
    return fact;
}
```

Que-9. Write a C program that stores 5 integers in a one-dimensional array and prints them. Extend this to handle a two-dimensional array (3x3 matrix) and calculate the sum of all elements.

Ans.

```
#include<stdio.h>
Int main(){
Int a[5]={1,2,3,4,5},i,j;
For(i=0;i<5;i++)
{

printf(“%d”,a[i]);
}
Int b[3][3];
For(i=0;i<3;i++){
For(j=0;j<3;j++){
Printf(“\n enter the element in b[i][j]=”,i,j);
Scanf(“%d”,&b[i][j]);
}
```

```
}
Int sum=0;
For(i=0;i<3;i++){
For(j=0;j<3;j++){
Sum=sum+b[i][j];
}}
Printf("\n matrix of 3*3=\n");
For(i=0;i<3;i++){
For(j=0;j<3;j++){
   Printf("%d",b[i][j]);
}
Printf("\n");
}
Printf("\n sum of array element is =%d",sum);
return 0;
}
```

Que-10. Write a C program to demonstrate pointer usage. Use a pointer to modify the value of a variable and print the result.

Ans.

```
#include<stdio.h>
int main(){
        int a=10;
        int *ptr=&a;
```

```
        printf("\n address of a=%p",ptr);
        *ptr=20;
        printf("\n value of a=%d",a);
        return 0;
}
```

Que-11. Write a C program that takes two strings from the user and concatenates them using strcat(). Display the concatenated string and its length using strlen().

Ans.

```
#include<stdio.h>
#include<string.h>
int main(){

   char str1[100],str2[100];
        printf("\n enter the string 1=");
        gets(str1);
        printf("\n enter the string 2=");
        gets(str2);

        printf("\n original string 1=%s",str1);
        printf("\n original string 2=%s",str2);
```

```
    strcat(str1,str2);
    printf("\n concatanated string=%s",str1);

    int result=strlen(str1);
    printf("\n Lenth of concatenated string =%d",result);
    return 0;
}
```

Que-12. Write a C program that defines a structure to store a student's details (name, roll number, and marks). Use an array of structures to store details of 3 students and print them.

Ans.

```
#include<stdio.h>
struct student{
    char name[100];
    int roll;
    int marks;
};
int main(){
    struct student s[3];
    int i;
    for(i=0;i<3;i++){
        printf("\n Enter Student[%d] Name=",i+1);
        scanf("%s",&s[i].name);
```

```
        printf("\n Enter Student[%d] Roll Number=",i+1);
        scanf("%d",&s[i].roll);
        printf("\n Enter Student[%d] Marks=",i+1);
        scanf("%d",&s[i].marks);
    }
    for(i=0;i<3;i++){
        printf("\n Student[%d] Name=%s",i+1,s[i].name);
        printf("\n Student[%d] Roll no=%d",i+1,s[i].roll);
        printf("\n Student[%d] Marks=%d",i+1,s[i].marks);
        printf("\n");
        printf("\n");
    }

    return 0;
}
```

Que-13. Write a C program to create a file, write a string into it, close the file, then open the file again to read and display its contents.

Ans.

```
#include<stdio.h>
int main(){
    FILE *fp;
    fp=fopen("demo.txt","a");
    fprintf(fp,"\nhello this is my assignment value");
```

```
    fclose(fp);

    char str[100];
    FILE *fr;
    fr=fopen("demo.txt","r");

    if(fr==NULL){
        printf("\n file doesn't exists.");
    }
    else{
        while(fgets(str,sizeof(str),fr)){
            printf("\n%s",str);
        }
    }

    fclose(fr);
    return 0;
}
```
