1-Create a C program that defines a function to increment an integer by 1. The function should demonstrate call by value, showing that the original value remains unchanged.

```
#include <stdio.h>

void increment(int );

int main(){
   int a;
   printf("Enter value for a:");
   scanf("%d",&a);
   increment(a);
   printf("the value back in main is %d\n",a);
   return 0;
}

void increment(int a){

   a=a+1;
   printf("the value after increment is %d\n",a);
}
```

2-Write a C program that attempts to swap two integers using a function that employs call by value. Show that the original values remain unchanged after the function call.

```
#include <stdio.h>

void swap(int , int);

int main(){
   int a;
```

```
   int b;
   printf("Enter value for a:");
   scanf("%d",&a);
   printf("Enter value for b:");
   scanf("%d",&b);
   swap(a,b);
   printf("back in main\n");
   printf("a=%d and b=%d\n",a,b);
   return 0;
}

void swap(int a,int b)
{

   int temp =a;
   a=b;
   b=temp;
   printf("new a=%d and new b=%d\n",a,b);
}
```

3-Develop a C program that calculates the factorial of a number using call by value

```
#include <stdio.h>

void factorial(int );

int main(){
   int a;
   printf("Enter number:");
```

```
   scanf("%d",&a);
   factorial(a);
   return 0;
}

void factorial(int a)
{

   int fact =1;
   for(int i=1;i<=a;i++)
   {
      fact=fact*i;
   }
   printf("the factorial is %d\n",fact);
}
```

4-Create a C program that defines a function to find the maximum of two numbers using call by value.

```
void maximum(int , int);

int main(){
   int a;
   int b;
   printf("Enter value for a:");
   scanf("%d",&a);
   printf("Enter value for b:");
   scanf("%d",&b);
   maximum(a,b);
```

```
    return 0;
}

void maximum(int a,int b)
{

    if(a>b)
    {
        printf("a is bigger\n");
    }
    else
    {
        printf("b is bigger\n");
    }
}
```

5- Problem Statement 1: Arithmetic Operations Calculator

Description: Write a C program that performs basic arithmetic operations (addition, subtraction, multiplication, and division) on two numbers provided by the user.

The program should use functions to perform each operation and demonstrate call by value.

Requirements: Create separate functions for addition, subtraction, multiplication, and division.

Each function should take two parameters (the numbers) and return the result.

Use appropriate data types for the variables.

Use operators for arithmetic calculations.

Example Input/Output: Enter first number: 10

Enter second number: 5

Addition: 15

Subtraction: 5

Multiplication: 50

Division: 2.0

```
#include<stdio.h>
void addition(int,int);
void subtraction(int,int);
void multi(int,int);
void division(int,int);

int main()
{
   char op;
   int inp1,inp2;
   float n;
   printf("Enter the first input: ");
   scanf("%d",&inp1);
   printf("\n");

   printf("Enter the operation: ");
   scanf(" %c",&op);
   printf("\n");

   printf("Enter the second input: ");
   scanf("%d",&inp2);
   printf("\n");

   switch(op)
```

```
{
   case '+':
      addition(inp1,inp2);
      break;

   case '-':
      subtraction(inp1,inp2);
      break;

   case '*':
      multi(inp1,inp2);
      break;

   case '/':
      if(inp2==0)
      {
         printf("Division by zero not possible");
         break;
      }
      else
      {
         division(inp1,inp2);
         break;
      }

   default:
      printf("Enter a valid opearator");
      break;
```

```
    }
    return 0;
}

void addition(int x, int y)
{
    int s=0;
    s=x+y;
    printf("sum is %d",s);
}

void subtraction(int x, int y)
{
    int s=0;
    s=x-y;
    printf("difference is %d",s);
}

void multi(int x, int y)
{
    int s=0;
    s=x*y;
    printf("product is %d",s);
}

void division(int x, int y)
{
```

```
   float s=0.0;
   s=(float)x/y;
   printf("quotient is %.2f",s);
}
```

6- Problem Statement 2: Temperature Conversion

Description: Develop a C program that converts temperatures between Celsius and Fahrenheit.

The program should use functions to handle the conversions and demonstrate call by value.

Requirements: Create two functions: one for converting Celsius to Fahrenheit and another for converting Fahrenheit to Celsius.

Each function should accept a temperature value as an argument and return the converted temperature.

Use appropriate data types for temperature values.

Use arithmetic operators to perform the conversion calculations.

Example Input/Output:

Enter temperature in Celsius: 25

Temperature in Fahrenheit: 77.0

Enter temperature in Fahrenheit: 77

Temperature in Celsius: 25.0

```
#include<stdio.h>
#include <string.h>

float tofahren(float );
float tocelcius(float );
int main()
{
```

```
    char unit[5];
    float temp;
    printf("Enter your temp unit(type cel or fah): ");
    scanf("%s",unit);
    printf("Enter the temperature: ");
    scanf("%f",&temp);
    if(strcmp(unit,"cel")==0)
    {
        float fa=tofahren(temp);
        printf("The temp in fahrenheit is %.2f",fa);
    }
    else
    {
        float cel=tocelcius(temp);
        printf("The temp in celcius is %.2f",cel);
    }

    return 0;
}

float tofahren(float c)
{
    float fa=(9.0/5)*c + 32;
    return fa;
}

float tocelcius(float f)
{
```

```
   float cel=(5.0/9)*(f -32) ;
   return cel;
}
```

7- Problem Statement 3: Simple Interest Calculator

Description: Develop a C program that calculates simple interest based on user input for principal amount, rate of interest, and time period.

The program should use a function to compute interest and demonstrate call by value. Requirements: Implement a function that takes three parameters (principal, rate, time) and returns the calculated simple interest.

Use appropriate data types for financial calculations (e.g., float or double).

Utilise arithmetic operators to compute simple interest using the formula SI = P×R×T/100

Example Input/Output: Enter principal amount: 1000

Enter rate of interest: 5

Enter time period (in years): 3

Simple Interest is: 150.0

```
#include<stdio.h>
void simpleintrest(int, int,int);
void simpleintrest(int p, int r,int t)
{
   float si=(float)(p*r*t)/100;
   printf("simple interest is %f",si);
}

int main()
{
   int p,r,t;
```

```
   printf("\nEnter Principle Amount:");
   scanf("%d",&p);
   printf("\nEnter intrest rate:");
   scanf("%d",&r);
   printf("\nEnter time period in years:");
   scanf("%d",&t);
   simpleintrest(p,r,t);
   return 0;
}
```

8- Write a C program that swaps the values of two integers using pointers.

```
#include <stdio.h>
int main()
{
    int num1=10,num2=20;
    int *p1 = &num1;
    int *p2 = &num2;
   printf("Before swapping: num1 = %d, num2 = %d\n", num1, num2);
    int temp = *p1;
    *p1 = *p2;
    *p = temp;
    printf("After swapping: num1 = %d, num2 = %d\n", num1, num2);
    return 0;
}
```

9-.WAP to swap the number using swap function and follow the pass by reference method.

```
#include<stdio.h>
void swap(int *,int *);
int main()
{
   int num1=10,num2=20;
   printf("\nBefore Swapping num1=%d num2=%d\n",num1,num2);
   swap(&num1,&num2);
   printf("\nAfter Swapping num1=%d num2=%d",num1,num2);
   return 0;
}
void swap(int *p,int *q )
{
   int temp=*p;
   *p=*q;
   *q=temp;
}
```

10-WAP for Finding the Cube of a Number Using Pass by Reference

```
#include <stdio.h>
int cube(int *n)
{
   int c = (*n) * (*n) * (*n);
   return c;
}

int main()
{
   int num;
```

```
    printf("Enter a number: ");
    scanf("%d", &num);

    int c=cube(&num);
    printf("The cube of the number is: %d\n", c);
    return 0;
}
```

11- WAP to calculate the simple interest with the help of a function and pass call by reference method.

```
#include<stdio.h>
int simpleinterest(int *a,int *b,int *c)
{
    int si=((*a)*(*b)*(*c))/100;
    return si;
}

int main()
{
    int p,r,t,si;
    printf("\nEnter principle amount:");
    scanf("%d",&p);
    printf("\nEnter rate of interest:");
    scanf("%d",&r);
    printf("\nEnter time period:");
    scanf("%d",&t);
    si=simpleinterest(&p,&r,&t);
```

```
    printf("\nSimple Interest is %d",si);
}
```