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.

```c
#include<stdio.h>
void func(int a);
int main()
{
    int a =10;
    func(a);
    printf("value of a in main function is %d \n",a);
    return 0;
}
void func(int a)
{
   printf("the value of a in user defined function is %d \n",++a);
}
```

Output
The value of a in user defined function is 11
value of a in main function is 10

---

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.

```c
    #include<stdio.h>

void func(int a,int b);
int main()
{
    int a =10,b=20;
    func(a,b);
    printf("value of a and b in main function is %d and %d \n",a,b);
    return 0;
}
void func(int a,int b)
{
    int swap;
    swap=a;
    a=b;
    b=swap;
   printf("the value of a and b in user defined function after swapping is %d and
%d\n",a,b);
}
```

Output
The value of a and b in user defined function after swapping is 20 and 10
value of a and b in main function is 10 and 20

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

```
#include<stdio.h>
void func(int n);
int main()
{
    int a ;
    printf("enter a number \n");
    scanf("%d",&a);
    func(a);
    return 0;
}
void func(int n)
{
    int fact=1;
    for(int i=1;i<=n;i++)
    {
        fact=fact*i;
    }
   printf("the factorial of %d is %d \n",n,fact);
}
```

Output
enter a number
5
the factorial of 5 is 120

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

```
#include<stdio.h>
int func(int n,int m);
int main()
{
    int a ,b,ans;
    printf("enter 2 number \n");
    scanf("%d %d",&a,&b);
    ans=func(a,b);
    printf("the bigger number is %d",ans);
    return 0;
}
int func(int n,int m)
{
    if (n>m)
        return n;
    else
        return m;
}
```

Output

```
enter 2 number
4 10
the bigger number is 10
```

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.

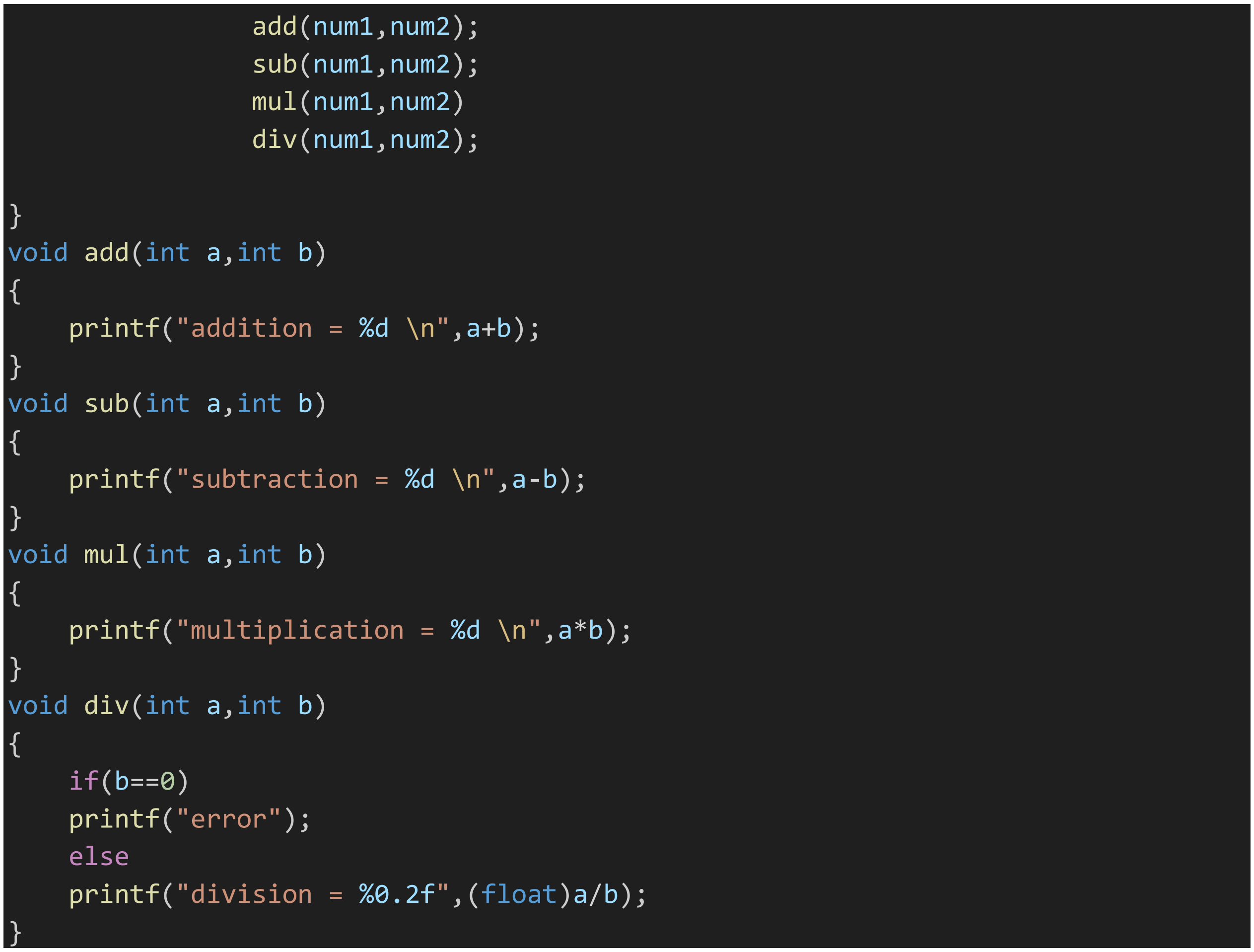
Use appropriate data types for the variables.

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 add(int a,int b);
void sub(int a,int b);
void mul(int a,int b);
void div(int a,int b);
int main()
{
    int num1,num2,i=1;
    printf("enter 2 numbers \n");
    scanf("%d %d",&num1,&num2);
```



```
                add(num1,num2);
                sub(num1,num2);
                mul(num1,num2)
                div(num1,num2);

}
void add(int a,int b)
{
    printf("addition = %d \n",a+b);
}
void sub(int a,int b)
{
    printf("subtraction = %d \n",a-b);
}
void mul(int a,int b)
{
    printf("multiplication = %d \n",a*b);
}
void div(int a,int b)
{
    if(b==0)
    printf("error");
    else
    printf("division = %0.2f",(float)a/b);
}
```

Output

```
enter 2 numbers
5 4
addition = 9
subtraction = 1
multiplication = 20
division = 1.25
```

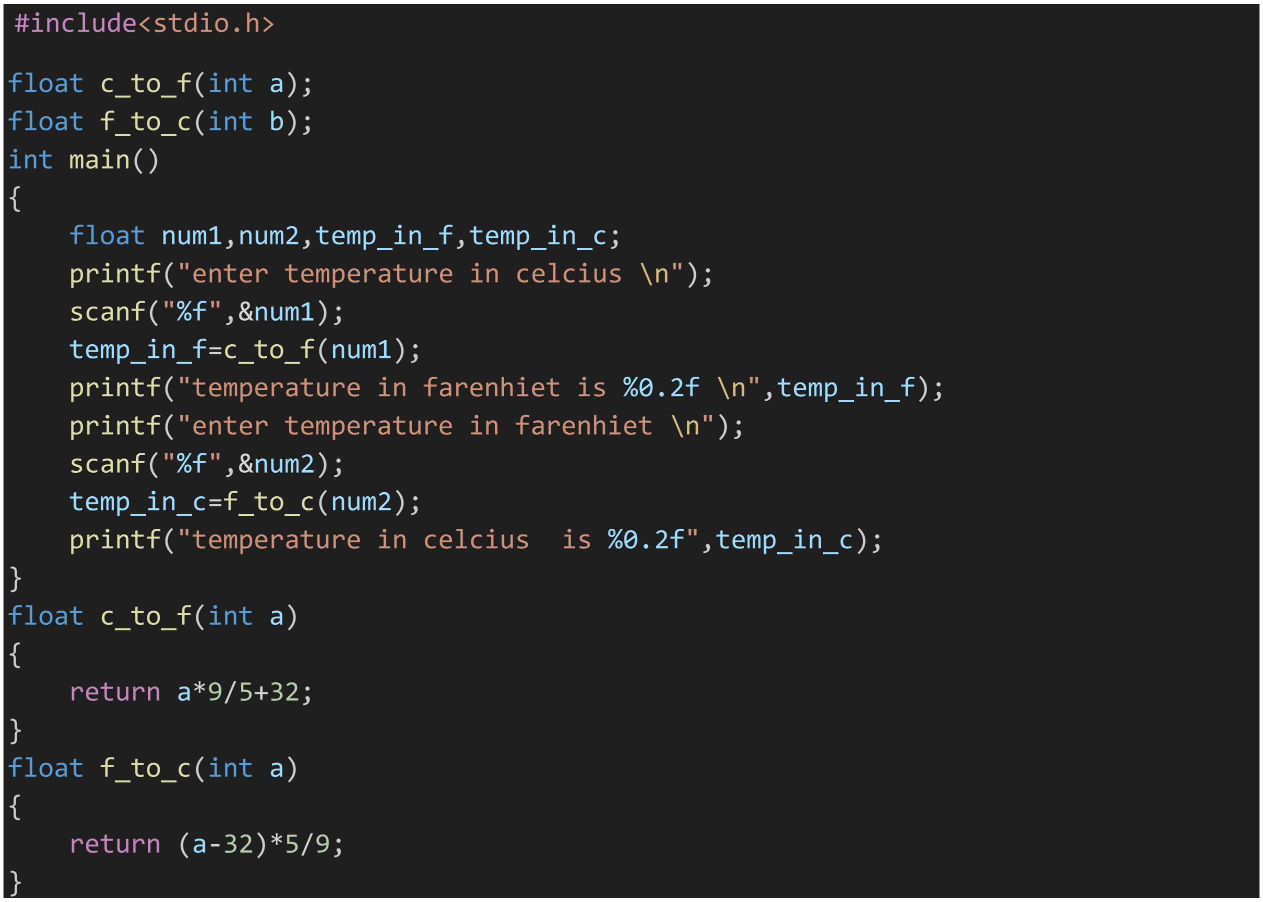
Problem Statement 2: Temperature Conversion

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>

float c_to_f(int a);
float f_to_c(int b);
int main()
{
    float num1,num2,temp_in_f,temp_in_c;
    printf("enter temperature in celcius \n");
    scanf("%f",&num1);
    temp_in_f=c_to_f(num1);
    printf("temperature in farenhiet is %0.2f \n",temp_in_f);
    printf("enter temperature in farenhiet \n");
    scanf("%f",&num2);
    temp_in_c=f_to_c(num2);
    printf("temperature in celcius  is %0.2f",temp_in_c);
}
float c_to_f(int a)
{
    return a*9/5+32;
}
float f_to_c(int a)
{
    return (a-32)*5/9;
}
```

Output

```
enter temperature in celcius
0
temperature in farenhiet is 32.00
enter temperature in farenhiet
32
temperature in celcius  is 0.00
```

==

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).

Utilize 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

```c
#include<stdio.h>
float SI(int p, float r, int t);
int main()
{
    int P,T;
    float R,ans;
    printf("enter principal amount \n");
    scanf("%d",&P);
    printf("enter rate of interest \n");
    scanf("%f",&R);
    printf("enter time period \n");
    scanf("%d",&T);
    ans=SI(P,R,T);
    printf("simple interest is %0.2f",ans);
}
float SI(int p, float r, int t)
{
    float si;
    si=(p*r*t)/100;
    return si;
}
```

Output

```
enter principal amount
1000
enter rate of interest
5
enter time period
3
simple interest is 150.00
```

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

```
#include<stdio.h>
int main()
{
    int num1=10,num2=20,temp;
    int *ptr1=&num1,*ptr2=&num2;
    temp=*ptr1;
    *ptr1=*ptr2;
    *ptr2=temp;
    printf("value of num1 and num2 after swapping= %d and %d \n",num1,num2);

    return 0;
}
```

Output
value of num1 and num2 after swapping= 20 and 10

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

```
#include<stdio.h>
int swap(int *,int*);
int main()
{
    int num1=10,num2=20;
    swap(&num1,&num2);
    printf("value of num1 and num2 after swapping= %d and %d \n",num1,num2);
    return 0;
}
int swap(int*a,int*b)
{
    int temp;
    temp=*a;
    *a=*b;
    *b=temp;
}
```

Output
value of num1 and num2 after swapping= 20 and 10

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

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

```
void cube(int *);
int main()
{
    int num1=10;
    cube(&num1);
    return 0;
}
void cube(int*a)
{
    printf("cube of the number= %d  \n",*a* *a* *a);
}
```

Output
cube of the number= 1000

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

```
#include<stdio.h>
float SI(int *p, float *r, int *t);
int main()
{
    int P,T;
    float R,ans;
    printf("enter principal amount \n");
    scanf("%d",&P);
    printf("enter rate of interest \n");
    scanf("%f",&R);
    printf("enter time period \n");
    scanf("%d",&T);
    ans=SI(&P,&R,&T);
    printf("simple interest is %0.2f",ans);
}
float SI(int *p, float *r, int *t)
{
    float si;
    si=(*p**r**t)/100;
    return si;
}
```

Output

```
enter principal amount
1000
enter rate of interest
5
enter time period
```

```
3
simple interest is 150.00
```