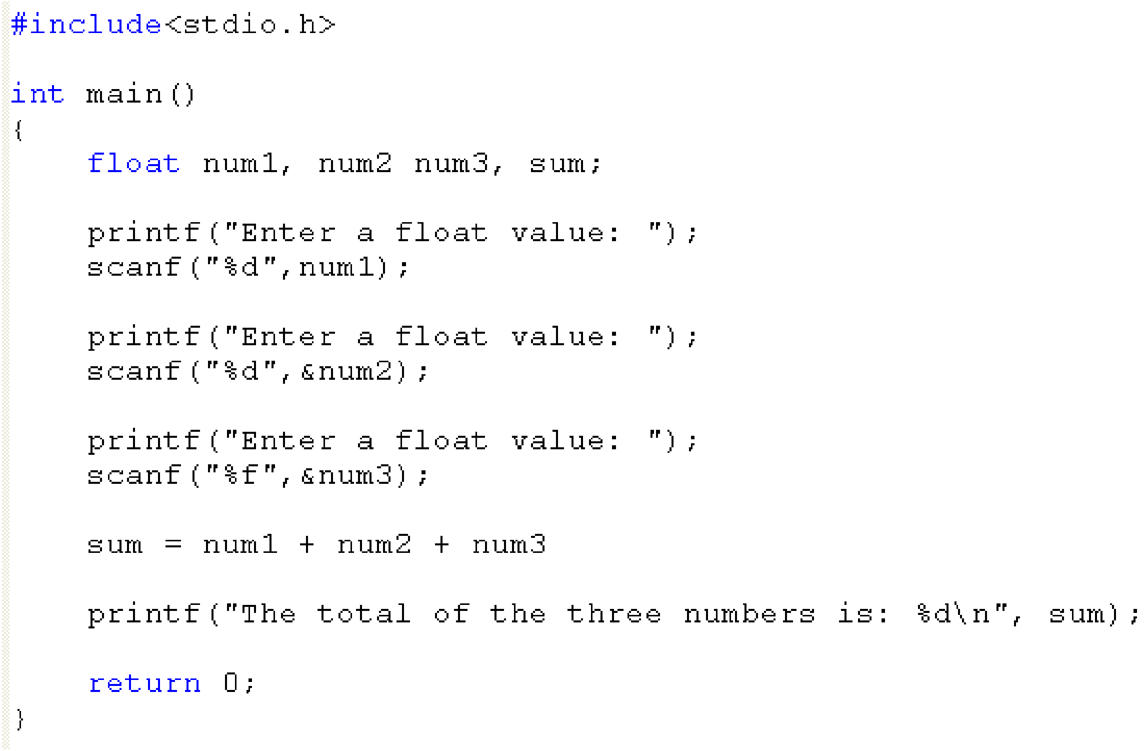
Ceng198

Introduction to Computer Programming

Lab#2

Part 1. Error Correction

Compile and run the following program. Correct the errors if there is any.

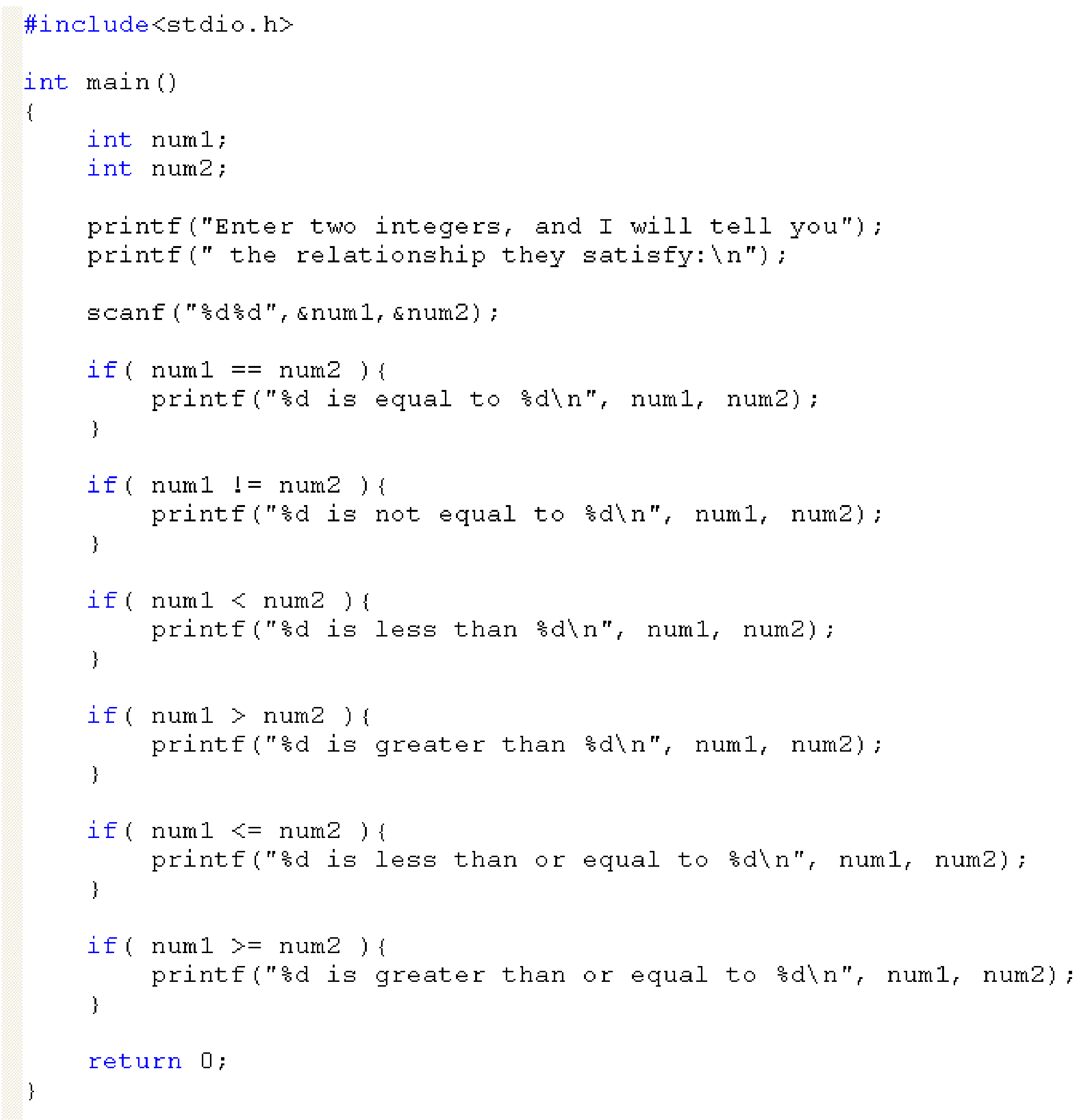


```c
#include<stdio.h>

int main()
{
    float num1, num2 num3, sum;

    printf("Enter a float value: ");
    scanf("%d",num1);

    printf("Enter a float value: ");
    scanf("%d",&num2);

    printf("Enter a float value: ");
    scanf("%f",&num3);

    sum = num1 + num2 + num3

    printf("The total of the three numbers is: %d\n", sum);

    return 0;
}
```

Part 2.

a.) Equality and Relational Operators

Compile and run the following program. Analyze its output. Run the program using several numbers as inputs.



```c
#include<stdio.h>

int main()
{
    int num1;
    int num2;

    printf("Enter two integers, and I will tell you");
    printf(" the relationship they satisfy:\n");

    scanf("%d%d",&num1,&num2);

    if( num1 == num2 ){
        printf("%d is equal to %d\n", num1, num2);
    }

    if( num1 != num2 ){
        printf("%d is not equal to %d\n", num1, num2);
    }

    if( num1 < num2 ){
        printf("%d is less than %d\n", num1, num2);
    }

    if( num1 > num2 ){
        printf("%d is greater than %d\n", num1, num2);
    }

    if( num1 <= num2 ){
        printf("%d is less than or equal to %d\n", num1, num2);
    }

    if( num1 >= num2 ){
        printf("%d is greater than or equal to %d\n", num1, num2);
    }

    return 0;
}
```

b.) Modification

Modify the program above so that

- when they are exactly equal your program should multiply them,
- if they are not equal your program should divide the numbers,
- if the first number is greater than the second number your program should subtract the two numbers,
- if the first number is less than the second number your program should add the two numbers,
- if the first number is less than or equal to the second number your program should calculate the average of the two numbers,
- and if the first number is greater than or equal to the second number then your program should calculate the remainder of the division of the two numbers.

Part 3. Do It Yourself.

Write a complete C program to do following tasks:

- 4 numbers will be entered
- They will be added 2 by 2.
- The modulus of the obtained 2 number will be printed out.

Part 4. Do It Yourself.

In the previous example, there is a danger that the obtained second number after addition operation could be gretater than the first one. So, the modulus operator will produce a wrong result. To avoid from this danger modify the previous program as;

- Check if the second number is greater than than the first one.
- if TRUE, start over (you will need to insert a *while* loop control structure)
- if FALSE, apply the modulus operator

Part 5. Do It Yourself.

Write a complete C program to produce (exactly) the following output. This output is just a sample output and subject to change depending to your entered X and Y values. The relational operators are ==, !=, >, <, >=, <=.

Program Output:

```
Enter two integers, and I will tell you the relationship thery satisfy: X Y

X is not equal to Y

X is less than or equal to Y

X is less than Y
```