NATIONAL UNIVERSITY OF SCIENCE AND TECHNOLOGY

SCHOOL OF MECHANICAL AND MANUFACTURING ENGINEERING

CS-114 - Fundamental of Programing

LAB MANUAL # 5 (Home Task)

ME -15 (C)

SUBMITTED BY:

MOHAMMAD GULZAIB

SUBMITTED TO:

SIR M. AFFAN

REPETITION STRUCTURE AND LOOPS

Repetition structure or loop is used to repeat the same process again and again until the required conditions are fulfilled. The programmer sets the conditions for the loop and the computer runs the loop under specific conditions. The loop will continue to run until the answer of condition comes out to be yes. Once the condition's answer comes out as false, the loop will terminate automatically, and loop ends.

Types of Repetition Structure:

There are two types of repetition structures:

1. Pretest
2. Post-Test loops

Pretest:

In this type of loops the condition is written at the beginning of loop. The loop first check that condition and then goes to loop. If loop is true, then main body will run and every time the loop run there will be change in the variable. If we did not make any change in value of variable then the loop will become infinite loop. For example, while loop, for loop, etc.

Post Test loop:

The main and only difference between these two types of loops is that posttest loop first runs and then checks the condition. It means that the loop body will run at least one time irrespective of the condition is true or not. For example, do..while loop.

For loop:

This loop will has defined starting point, defined condition, and defined final point. In for loop, it also has the increment in the variable, which is used to check the condition.

Syntax:

```
for (starting point; condition; increment)
  loop body
```

While loop:

While loop has only condition. It does not require increment to run. However, in order to get the required result, we need to define the increment, otherwise it will continue to loop.

Syntax:

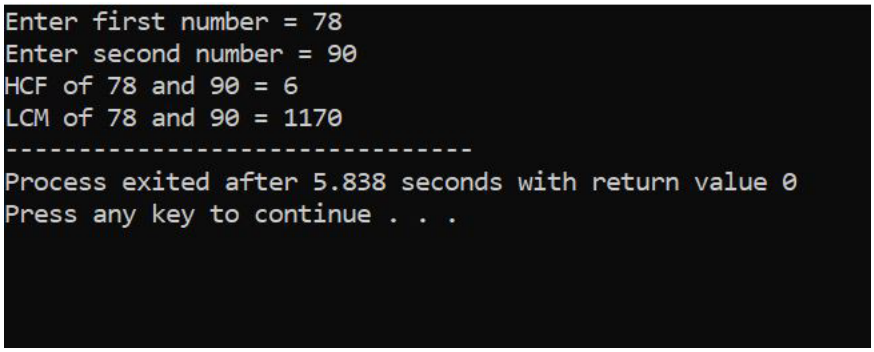
```
while (condition)
    loop body
```

1. **Write a program in C++ to find LCM of any two numbers using HCF.**

Code:

```
#include<iostream>
using namespace std;
int main(){
	int x,y,z;
	cout<<"Enter first number = ";
	cin>>x;
	cout<<"Enter second number = ";
	cin>>y;
	for(int i=1;i<=x||i<=y;i++){    //loop will continue to run until "i"becomes greater than both numbers:
		if(x%i==0&&y%i==0){        //this condition will check if both numbers are divisible by i:
			z=i;                  //everytime the numbers gets divided z becomes equal to i, that will be
its HCF:
		}
	}
	cout<<"HCF of "<<x<<" and "<<y<<" = "<<z<<endl;    //this will show its HCF:
	int LCM = (x*y)/z;                          //LCM will be calculated using formula HCF*LCM = x*y
	cout<<"LCM of "<<x<<" and "<<y<<" = "<<LCM;       //shows the LCM:
	return 0;
}
```

Result:

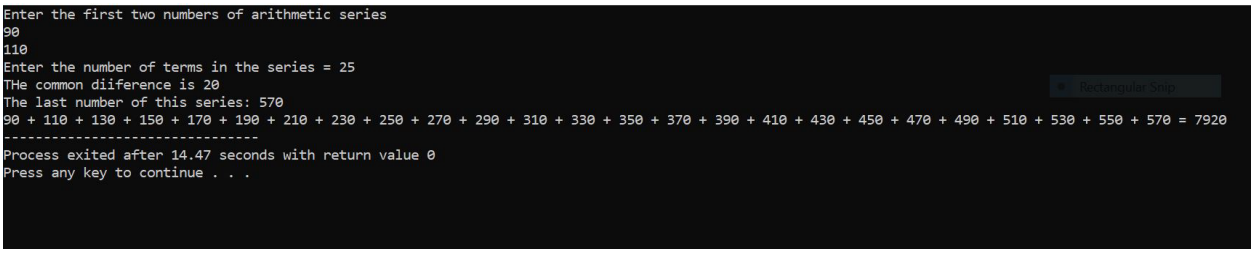


2. **Write a program in C++ to find out the sum of an Arithmetic progression series.**

Code:

```
#include<iostream>
using namespace std;
int main(){
	int x,y,z;
	cout<<"Enter the first two numbers of arithmetic series \n"; //gets input from user
	cin>>x;
	cin>>y;
	cout<<"Enter the number of terms in the series = ";	//the number of terms in series
	cin>>z;
	int s= y-x;	//common difference between two consecutive numbers
	cout<<"THe common diiference is "<<s<<endl;
	int D = (z/2)*(2*x+(z-1)*s) ;	//calculate the sum of AP using formula
	int q=x+(z-1)*s;	//find out last number of the series
	cout<<"The last number of this series: "<<q<<endl;
	int w=x;
	for(w;w<=q;w=w+s){	//to print out whole series, this loop runs until w=x becomes equal to
last number of series that is q:
		if(w!=q){	//if w is not equal to q it will print the following result
			cout<<w<<" + ";
		}
		else
		cout<<w<<" = ";	//if w becomes equal to q it will print following result
	}
	cout<<D;
	return 0;
}
```

Result:



3. Write a program in C++ to create a diamond.

Code:

```
#include<iostream>
using namespace std;


int main(){

	int x,y,z;
	cout<<"Enter the number of rows = ";
	cin>>x;
		for(int i=1;i<=x;i++){  //loop will ceck the number of rows
			for(y=x;y>=i;y--){  //this loop will print spaces until y becomes less than i

			cout<<" ";}
			for(z=1;z<=(2*i)-1;z++){ //this loop will print stars until z becomes less than (2*i)-1

			cout<<"*";}
			cout<<"\n";          //to move to next line
		}
		for(int j=x;j>=1;j--){    //this loop will run in reverse order and runs until j becomes less than 1
			for(y=x;y>=j;y--){    //runs until y remains greater than or equal to j and prints spaces
				cout<<" ";
			}
			for(z=1;z<=(2*j)-1;z++){  //runs and prints star until z is less than or equal to (2*j)-1
			cout<<"*";}
			cout<<"\n";
		}

	return 0;
}
```

Result:

4. Write a program in C++ to convert a decimal number to binary number.

Code:

```
#include<iostream>
using namespace std;
int main(){
	int x,y,z=0;
	cout<<"Enter the decimal number ";   //takes input in decimal number system
	cin>>x;
	cout<<"("<<x<<") =";
	for(x;x>0;x=x/2){		//runs until x>0 and divides x by 2 which we do to convert decimal to binary
	y=x%2;			//remainders which will give number in binary will be stored in y
	cout<<y;}		//shows output of binary which is the number in binary
	return 0;
}
```

Result: