



NUST

NATIONAL UNIVERSITY
OF SCIENCES & TECHNOLOGY

Computer Programming Lab

Manuel5 Home tasks

Submitted by:

Shah Jahan khan

469192 (section C)

Submitted to:

Sir Affan Tariq

Sir Saqib

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

```
using namespace std;
```

```
int main(){
```

```
    int num1,num2,temp;
```

```
    int hcf;
```

```
    cout<<" Enter the first number: ";
```

```
    cin>>num1;
```

```
    cout<<" Enter the second number: ";
```

```
    cin>>num2;
```

```
    if(num2>num1){
```

```
        temp=num2;
```

```
        num2=num1;
```

```
        num1=temp;
```

```
    }
```

```
    for(int i=1; i<=num2 ; i++){
```

```
        if(num1%i==0 && num2%i==0){
```

```
            hcf=i;
```

```
        }
```

```
    }
```

```
    int lcm= (num1*num2)/hcf;
```

```
    cout<<"hcf is: "<<hcf<<endl;
```

```
    cout<<"lcm is: "<<lcm<<endl;
```

```
    return 0;
```

```
}
```

Output

```
^ /tmp/kKgbmI7hTr.o
```

```
Enter the first number: 2
```

```
Enter the second number: 3
```

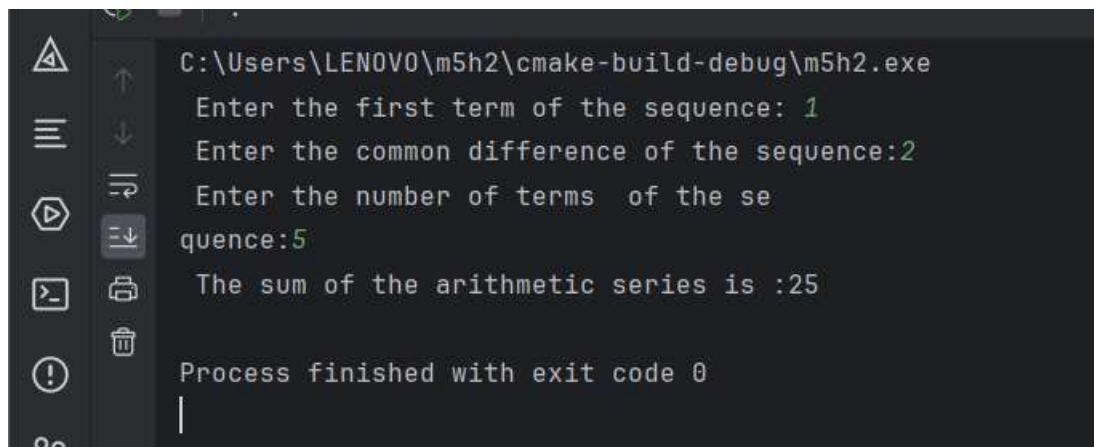
```
hcf is: 1
```

```
lcm is: 6
```

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

```
#include <iostream>
using namespace std;
int main() {
    int a , d , n ;
    cout<<" Enter the first term of the sequence:";
    cin>>a;
    cout<<" Enter the common difference of the sequence:";
    cin>>d;
    cout<<" Enter the number of terms  of the sequence:";
    cin>>n;
    int sum=0;
    for(int i=0; i<n; i++) {
        int term ;
        term = a+i*d;
        sum+= term;
    }
    cout<<" The sum of the arithmetic series is : " <<sum<<endl;

    return 0;
}
```

A screenshot of a terminal window showing the execution of a C++ program. The window title is "C:\Users\LENOVO\m5h2\cmake-build-debug\m5h2.exe". The program prompts the user for three inputs: the first term of the sequence (1), the common difference (2), and the number of terms (5). It then outputs the sum of the arithmetic series as 25. The terminal shows the execution path and the final exit code 0.

```
C:\Users\LENOVO\m5h2\cmake-build-debug\m5h2.exe
Enter the first term of the sequence: 1
Enter the common difference of the sequence: 2
Enter the number of terms  of the se
quence: 5
The sum of the arithmetic series is :25

Process finished with exit code 0
```

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

```
#include<iostream>
using namespace std;
int main(){
    int row , col, spa;

    for(row= 1; row<= 4 ; row++) {
        for (spa = 1; spa <= 4 - row; spa++) {
            cout << " ";
        }
        for (col = 1; col <= row * 2 - 1; col++) {
            cout << " * ";
        }
        cout<<endl;
    }

    for(row=3; row>=1 ; row--){
        for(spa=1 ; spa<=3-row ; spa++) {
            cout << " ";
        }
        for(col=1 ; col<=2*row-1 ; col++){
            cout<< " * ";
        }
        cout<<endl;
    }
    return 0;
}
```



The screenshot shows a terminal window with the command prompt at C:\Users\LENOVO\MSH3\snake-build-debug\MSH3.exe. The output displays a diamond shape composed of asterisks. The top half of the diamond has 4 rows, and the bottom half has 3 rows, totaling 7 rows. The bottom row of the diamond consists of a single asterisk. Below the diamond, the message "Process finished with exit code 0" is displayed.

```
C:\Users\LENOVO\MSH3\snake-build-debug\MSH3.exe
  *
 * * *
* * * * *
* * * * * *
* * * * *
 * * *
  *

Process finished with exit code 0
```

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

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

int main() {
    int decimalNumber;
    cout << "Enter a decimal number: ";
    cin >> decimalNumber;

    int binaryNumber[32];    // Assuming 32-bit binary representation

    if (decimalNumber == 0) {
        cout << "Binary equivalent: 0" << endl;
        return 0;
    }

    int i = 0;
    while (decimalNumber > 0) {
        binaryNumber[i] = decimalNumber % 2;
        decimalNumber /= 2;
        i++;
    }

    cout << "Binary equivalent: ";
    for (int j = i - 1; j >= 0; j--) {
        cout << binaryNumber[j];
    }
    cout << endl;

    return 0;
}
```

