COMPUTERS AND PROGRAMMING

(EE-163)

ASSIGNMENT

NAME: Muhammad Khuzaima Hassan
ROLL NO: EE-22104
SECTION: C
DEPARTMENT: Electrical Engineering
SUBMITTED TO: Sir Iqbal Azeem
SUBMISSION DATE: 14-JULY-2023

Q1: How C++ program is compiled into an executable? Answer in 4 lines only. Make a flow chart.

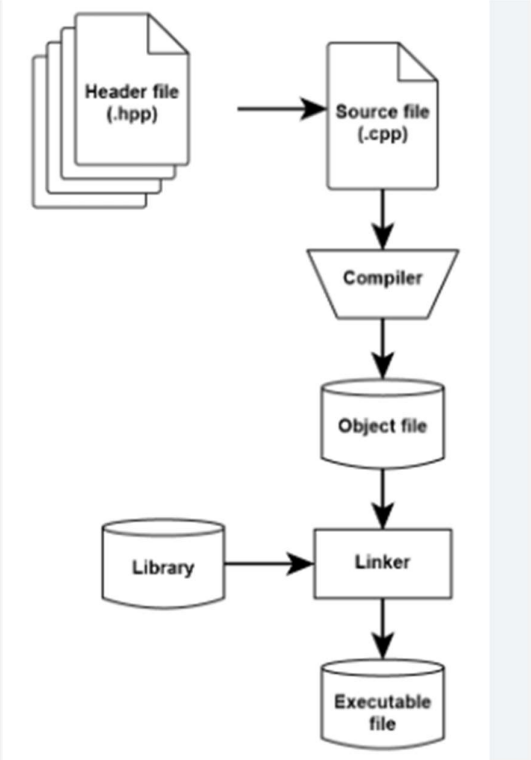
ANS: C++ program compilation into an executable involves the following steps:

Preprocessing: The preprocessor resolves directives and includes header files.

Compilation: The compiler translates the source code into object code (.obj) files.

Linking: The linker combines the object code files and resolves external dependencies, generating an executable file.

Execution: The operating system loads the executable into memory and executes it.



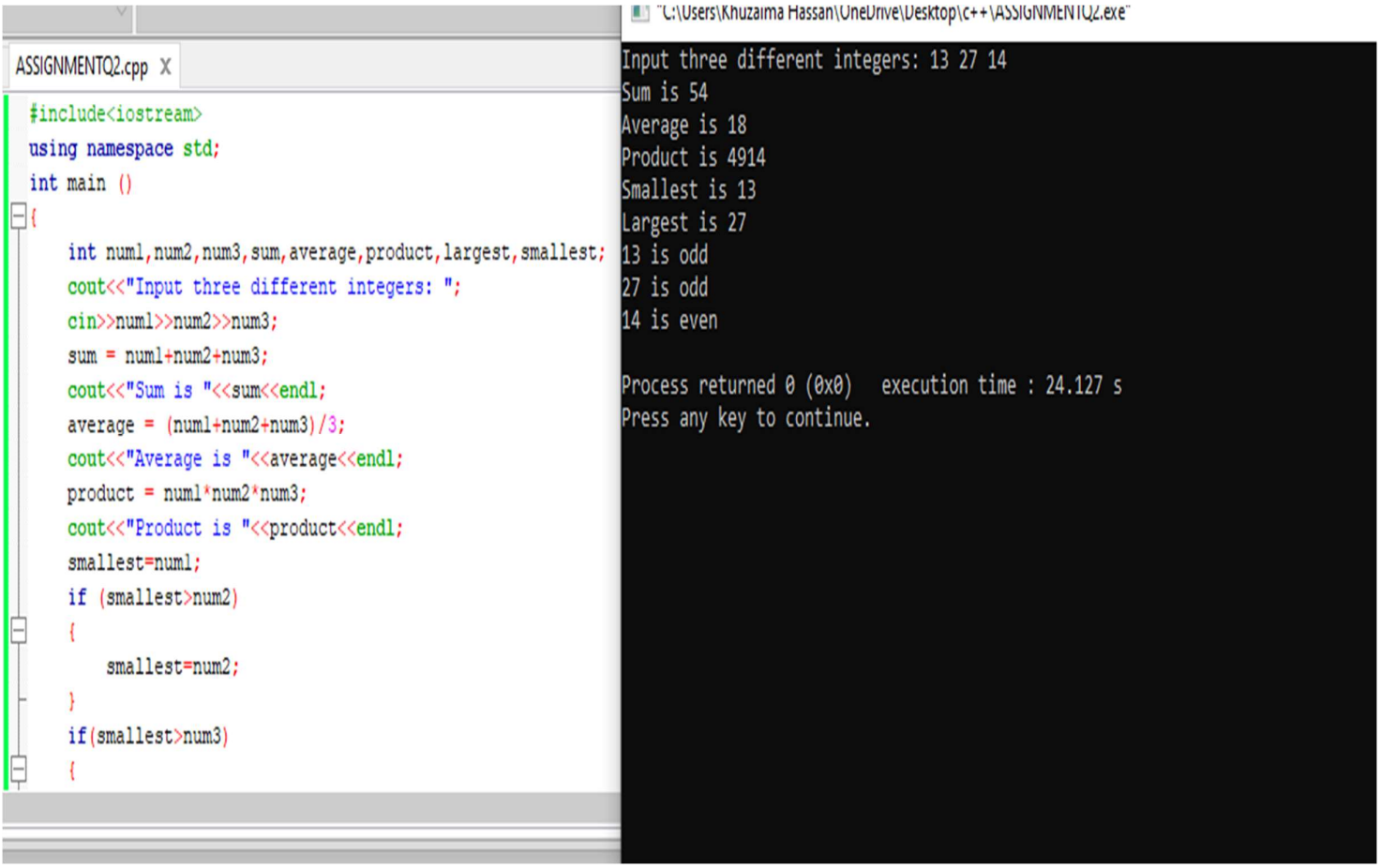
Q2: Write a program that inputs three integers from the keyboard and prints the sum, average, product, smallest and largest of these numbers. The screen dialog should appear as follows:
Input three different integers: 13 27 14
Sum is 54
Average is 18
Product is 4914
Smallest is 13
Largest is 27
13 is odd
27 is odd
14 is even

CODE:

```
#include<iostream>
using namespace std;
int main ()
{
   int num1,num2,num3,sum,average,product,largest,smallest;
   cout<<"Input three different integers: ";
   cin>>num1>>num2>>num3;
   sum = num1+num2+num3;
   cout<<"Sum is "<<sum<<endl;
   average = (num1+num2+num3)/3;
   cout<<"Average is "<<average<<endl;
   product = num1*num2*num3;
   cout<<"Product is "<<product<<endl;
   smallest=num1;
   if (smallest>num2)
   {
      smallest=num2;
   }
   if(smallest>num3)
   {
      smallest=num3;
   }
   cout<<"Smallest is "<<smallest<<endl;
   largest=num1;
   if (largest<num2)
   {
```

```
   largest=num2;
}
if(largest<num3)
{
   largest=num3;
}
cout<<"Largest is "<<largest<<endl;
if (num1%2==0)
{
   cout<<num1<<" is even"<<endl;
}
else
{
   cout<<num1<<" is odd"<<endl;
}
if (num2%2==0)
{
   cout<<num2<<" is even"<<endl;
}
else
{
   cout<<num2<<" is odd"<<endl;
}
if (num3%2==0)
{
   cout<<num3<<" is even"<<endl;
}
else
{
   cout<<num3<<" is odd"<<endl;
}
return 0;}
```

OUTPUT SCREEN:



ASSIGNMENTQ2.cpp

```
#include<iostream>
using namespace std;
int main ()
{
    int num1,num2,num3,sum,average,product,largest,smallest;
    cout<<"Input three different integers: ";
    cin>>num1>>num2>>num3;
    sum = num1+num2+num3;
    cout<<"Sum is "<<sum<<endl;
    average = (num1+num2+num3)/3;
    cout<<"Average is "<<average<<endl;
    product = num1*num2*num3;
    cout<<"Product is "<<product<<endl;
    smallest=num1;
    if (smallest>num2)
    {
        smallest=num2;
    }
    if(smallest>num3)
    {
```

"C:\Users\Khuzaima Hassan\OneDrive\Desktop\c++\ASSIGNMENTQ2.exe"

```
Input three different integers: 13 27 14
Sum is 54
Average is 18
Product is 4914
Smallest is 13
Largest is 27
13 is odd
27 is odd
14 is even

Process returned 0 (0x0)   execution time : 24.127 s
Press any key to continue.
```

Q3: Write a program that inputs a five digit integer, separates the integer into its digit and prints them separated by three spaces each. For example, if the user types in 42339, the program should print:
4 2 3 3 9

CODE:

```
#include<iostream>

using namespace std;

int main (){    int number;    cout<<"Enter a five digit integer: ";    cin>>number;

   int digit1,digit2,digit3,digit4,digit5;

   digit1 = (number/10000)%10;

   digit2 = (number/1000)%10;

   digit3 = (number/100)%10;

   digit4 = (number/10)%10;

   digit5 = number%10;

   cout<<digit1<<"   ";

   cout<<digit2<<"   ";

   cout<<digit3<<"   ";

   cout<<digit4<<"   ";

   cout<<digit5<<"   ";

   return 0;

}
```

OUTPUT SCREEN:

```
using namespace std;
int main ()
{
    int number;
    cout<<"Enter a five digit integer: ";
    cin>>number;
    int digit1,digit2,digit3,digit4,digit5;
    digit1 = (number/10000)%10;
    digit2 = (number/1000)%10;
    digit3 = (number/100)%10;
    digit4 = (number/10)%10;
    digit5 = number%10;
    cout<<digit1<<"   ";
    cout<<digit2<<"   ";
    cout<<digit3<<"   ";
    cout<<digit4<<"   ";
    cout<<digit5<<"   ";
    return 0;
}
```

```
"C:\Users\Khuzaima Hassan\OneDrive\Desktop\c++\ASSIGNMENTQ3.exe"
Enter a five digit integer: 12345
1   2   3   4   5
Process returned 0 (0x0)   execution time : 8.427 s
Press any key to continue.
```

Q4: Develop a C++ program that uses a `while` statement to determine the gross pay for each of several employees. The company pays "straight time" for the first 40 hours worked by each employee and pays "time-and-a-half" for all hours worked in excess of 40 hours. You are given a list of the employees of the company, the number of hours each employee worked last week and the hourly rate of each employee. Your program should input this information for each employee and should determine and display the employee's gross pay.

Sample Output:

```
Enter hours worked (-1 to end): 39
Enter hourly rate of the employee: 10.00
Salary is 390.00 Rs.
Enter hours worked (-1 to end): 40
Enter hourly rate of the employee: 10.00
Salary is 400.00 Rs.
Enter hours worked (-1 to end): 41
Enter hourly rate of the employee: 10.00
Salary is 415.00 Rs.
Enter hours worked (-1 to end): -1
```

CODE:

```
#include<iostream>;
using namespace std;
int main ()
{
   int hoursworked;
   double hourlyrate,grosspay;
   cout<<"Enter hours worked (-1 to end): ";
   cin>>hoursworked;
   while(hoursworked!=-1)
   {
      cout<<"Enter hourly rate of the employee: ";
      cin>>hourlyrate;
      if(hoursworked<=40)
      {        grosspay = hoursworked*hourlyrate;      }      else
      {
         grosspay = 40*hourlyrate+((hoursworked-40)*(hourlyrate*1.5));
      }
      cout<<"Salary is "<<grosspay<<" Rs."<<"\n\n";
      cout<<"Enter hours worked (-1 to end): ";
      cin>>hoursworked;

   }
   return 0;}
```

OUTPUT SCREEN:

```cpp
#include<iostream>;
using namespace std;
int main ()
{
    int hoursworked;
    double hourlyrate,grosspay;
    cout<<"Enter hours worked (-1 to end): ";
    cin>>hoursworked;
    while(hoursworked!=-1)
    {
        cout<<"Enter hourly rate of the employee: ";
        cin>>hourlyrate;
        if(hoursworked<=40)
        {
            grosspay = hoursworked*hourlyrate;
        }
        else
        {
            grosspay = 40*hourlyrate+((hoursworked-40)*(hourlyrate*1.5));
        }
        cout<<"Salary is "<<grosspay<<" Rs."<<"\n\n";
        cout<<"Enter hours worked (-1 to end): ";
        cin>>hoursworked;

    }
    return 0;

```

```
"C:\Users\Khuzaima Hassan\OneDrive\Desktop\c++\ASSIGNMENTQ4.exe"
Enter hours worked (-1 to end): 39
Enter hourly rate of the employee: 10.00
Salary is 390 Rs.

Enter hours worked (-1 to end): 40
Enter hourly rate of the employee: 10.00
Salary is 400 Rs.

Enter hours worked (-1 to end): 41
Enter hourly rate of the employee: 10.00
Salary is 415 Rs.

Enter hours worked (-1 to end): -1

Process returned 0 (0x0)   execution time : 40.567 s
Press any key to continue.
```

Q5: Write a program that ask user to enter an integer number and evaluates its factorial. Your program should print the output as below,

```
Enter an integer : 5
5 x 4 x 3 x 2 x 1 = 120
```

CODE:

```
#include<iostream>;

using namespace std;

int main ()

{

   int number,factorial=1;

   cout<<"Enter an integer : ";

   cin>>number;

   for(int i=number;i>=1;--i)

   {

      factorial*=i;

      if(i!=1)

      {

       cout<<i<<" x ";

      }

      else

      {

         cout<<"1";

      }

 }

   cout<<" = "<<factorial<<endl;

   return 0;}
```

OUTPUT SCREEN:

```
1  #include<iostream>;
2  using namespace std;
3  int main ()
4  {
5      int number,factorial=1;
6      cout<<"Enter an integer : ";
7      cin>>number;
8      for(int i=number;i>=1;--i)
9      {
10         factorial*=i;
11         if(i!=1)
12         {
13          cout<<i<<" x ";
14         }
15         else
16         {
17             cout<<"1";
18         }
19
20     }
21     cout<<" = "<<factorial<<endl;
22     return 0;
23 }
24
```

```
"C:\Users\Khuzaima Hassan\OneDrive\Desktop\c++\ASSIGNM...
Enter an integer : 5
5 x 4 x 3 x 2 x 1 = 120

Process returned 0 (0x0)   execution time : 2.680 s
Press any key to continue.
```

Q6: Write a program that ask user to input the number of elements in a Fibonacci sequence and then generates a Fibonacci sequence up-to the given number of elements.
(Hint: In Fibonacci sequence, the next element is
the sum of two previous values)
Sample Output:

```
Enter number of elements:10
0 1 1 2 3 5 8 13 21 34
```

CODE:

```
#include<iostream>
 using namespace std;
 int main (void)
 {
 int counter, n_terms;
 cout<<"Enter number of elements: ";
 cin>>n_terms;
 int newterm=0,prevterm=1,sum;

 for(counter=0;counter<n_terms;counter++)
 {
 if(counter%10==0);
 cout<<newterm<<" ";
 sum=prevterm+newterm;
 prevterm=newterm;
 newterm=sum;
 }
 return 0;
 }
```

OUTPUT SCREEN:

```
#include<iostream>
 using namespace std;
 int main (void)
 {
 // variable definition
 int counter, n_terms;
 cout<<"Enter number of elements: ";
 cin>>n_terms;
 int newterm=0,prevterm=1,sum;

 for(counter=0;counter<n_terms;counter++)
 {
 if(counter%10==0);
 cout<<newterm<<" ";
 sum=prevterm+newterm;
 prevterm=newterm;
 newterm=sum;
 }
 return 0;
 }
```

```
"C:\Users\Khuzaima Hassan\OneDrive\Desktop\c++\lesson_...
Enter number of elements: 10
0 1 1 2 3 5 8 13 21 34
Process returned 0 (0x0)   execution time : 1.814 s
Press any key to continue.
```

Q7: Write a program that reads three non-zero `double` values and determines and prints whether they could represents sides of triangle.
[Hint: *a*,*b* and *c* represents sides of triangle if following criteria is met,
$a + b > c$
$a + c > b$
$b + c > a$]

Sample Output:

```
Enter length of three sides: 3 4 5
They are sides of triangle.
Enter length of three sides: 2 2 5
They are not sides of triangle.
Enter length of three sides: 2.4 3.8 5.5
They are sides of triangle.
```

CODE:

```
#include<iostream>

using namespace std;

int main ()

{

   double a, b, c;

   cout<<"Enter length of three sides: ";

   cin>>a>>b>>c;

   if(a+b>c&&a+c>b&&b+c>a)

   {

      cout<<"They are sides of triangle";

   }

   else

   {

      cout<<"They are not sides of triangle";

   }

  return 0;}
```

OUTPUT SCREEN:

```
#include<iostream>
using namespace std;
int main ()
{
    double a, b, c;
    cout<<"Enter length of three sides: ";
    cin>>a>>b>>c;
    if(a+b>c&&a+c>b&&b+c>a)
    {
        cout<<"They are sides of triangle";
    }
    else
    {
        cout<<"They are not sides of triangle";
    }
   return 0;
}
```

```
"C:\Users\Khuzaima Hassan\OneDrive\Desktop\c++\ASSIGNMENTQ7.exe"
Enter length of three sides: 2 3 4
They are sides of triangle
Process returned 0 (0x0)   execution time : 5.251 s
Press any key to continue.
```

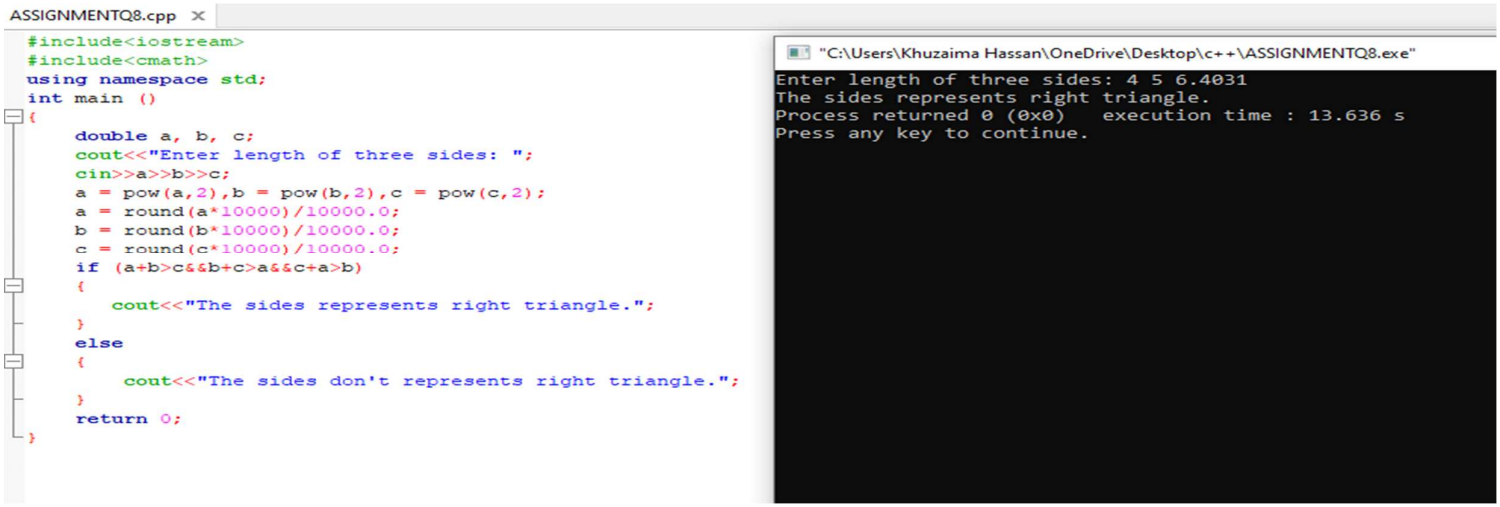
Q8: Write a program that reads three non-zero `double` values and determines and prints whether they are sides of right triangle. The program should verify the results up to 4 decimal places.
[Hint: Use Pythagoras theorem to determine whether the three sides form right triangle.] Sample Output:

```
Enter length of three sides: 3 4 5
The sides represents right triangle.
Enter length of three sides: 4 5 6.403
The sides don't represents right triangle.
Enter length of three sides: 4 5 6.4031
The sides represents right triangle.
```

CODE:

```
#include<iostream>

#include<cmath>

using namespace std;

int main ()

{

   double a, b, c;

   cout<<"Enter length of three sides: ";

   cin>>a>>b>>c;

   a = pow(a,2),b = pow(b,2),c = pow(c,2);

   a = round(a*10000)/10000.0;

   b = round(b*10000)/10000.0;

   c = round(c*10000)/10000.0;

   if (a+b>c&&b+c>a&&c+a>b)

   {

     cout<<"The sides represents right triangle.";

   }

   else

   {

      cout<<"The sides don't represents right triangle.";

   }

   return 0;}
```

OUTPUT SCREEN:



Q9: Write a program that ask user to input a floating point number and computes exponential of that number using Taylor series as below, $e^x = 1 + x1! + x^2 2! + x^3 3! + \cdots$

Also, prompt the user for desired accuracy of e (i.e., the number of terms in summation).

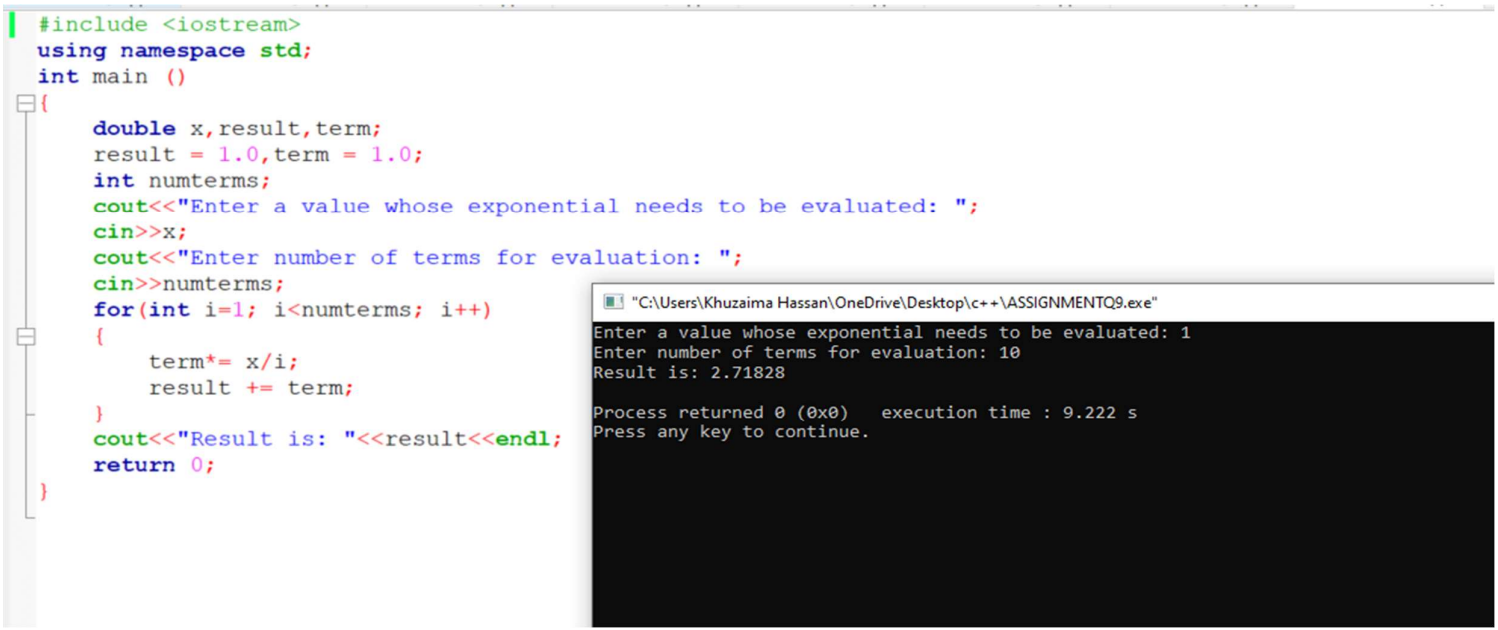
Sample Output:

```
Enter a value whose exponential needs to be evaluated: 1
Enter number of terms for evaluation: 20
Result is: 2.71828
```

CODE:

```
#include <iostream>

using namespace std;

int main ()

{

   double x,result,term;

   result = 1.0,term = 1.0;

   int numterms;

   cout<<"Enter a value whose exponential needs to be evaluated: ";

   cin>>x;

   cout<<"Enter number of terms for evaluation: ";

   cin>>numterms;

   for(int i=1; i<numterms; i++)

   {

      term*= x/i;

      result += term;

   }

   cout<<"Result is: "<<result<<endl;

   return 0;
```

OUTPUT SCREEN:



Q10: Write a program that ask user to input angle in radians and computes its sine using Taylor series as below,
$\sin(x) = \Sigma(-1)_n(2n+1)!x_{2n+1\infty n=0}$
Also, prompt the user for desired accuracy of *sine.* (i.e., the number of terms in summation). Sample Output:

```
Enter a value for sin evaluation: 2
Enter number of terms in the summation:10
Result is: 0.909297
```

CODE:

```
#include <iostream>

using namespace std;

int main ()

{

   int terms;

   double angle,result = 0.0;

   cout << "Enter a value for sin evaluation: ";

   cin >> angle;

   cout << "Enter number of terms in the summation: ";

   cin >> terms;

   int sign = 1;

   double power = angle,factorial = 1.0;

   for (int n = 0; n < terms; n++)

   {

      result += sign * power / factorial;

      power *= angle * angle;

      factorial *= (2 * n + 2) * (2 * n + 3);

      sign *= -1;

   }

   cout << "Result is: " << result <<endl;

   return 0;}
```

OUTPUT SCREEN:

```
#include <iostream>
using namespace std;
int main ()
{
    int terms;
    double angle,result = 0.0;
    cout << "Enter a value for sin evaluation: ";
    cin >> angle;
    cout << "Enter number of terms in the summation: ";
    cin >> terms;
    int sign = 1;
    double power = angle,factorial = 1.0;
    for (int n = 0; n < terms; n++)
    {
        result += sign * power / factorial;
        power *= angle * angle;
        factorial *= (2 * n + 2) * (2 * n + 3);
        sign *= -1;
    }
    cout << "Result is: " << result <<endl;
    return 0;

}
```

```
"C:\Users\Khuzaima Hassan\OneDrive\Desktop\c++\ASSIGNMENTQ10.exe"
Enter a value for sin evaluation: 2
Enter number of terms in the summation: 10
Result is: 0.909297

Process returned 0 (0x0)   execution time : 4.128 s
Press any key to continue.
```

Q11: Write a program that prints any one of the following pattern as shown below,

```
*                  **********          **********                  *
**                  *********          *********                  **
***                  ********          ********                  ***
****                  *******          *******                  ****
*****                  ******          ******                  *****
******                  *****          *****                  ******
*******                  ****          ****                  *******
********                  ***          ***                  ********
*********                  **          **                  *********
**********                  *          *                  **********
```

CODE:

```
#include<iostream>

using namespace std;

int main()

{

   for (int i = 10; i>0; --i)

   {

   for (int j = i; j > 0; --j)

   {

    cout<< "*";

   }

    cout<<endl;

   }

   return 0;

}
```

OUTPUT SCREEN:

```
#include<iostream>
using namespace std;
int main()
{

    for (int i = 10; i>0; --i)
    {
    for (int j = i; j > 0; --j)
    {
      cout<< "*";
    }
      cout<<endl;
    }
    return 0;
}
```

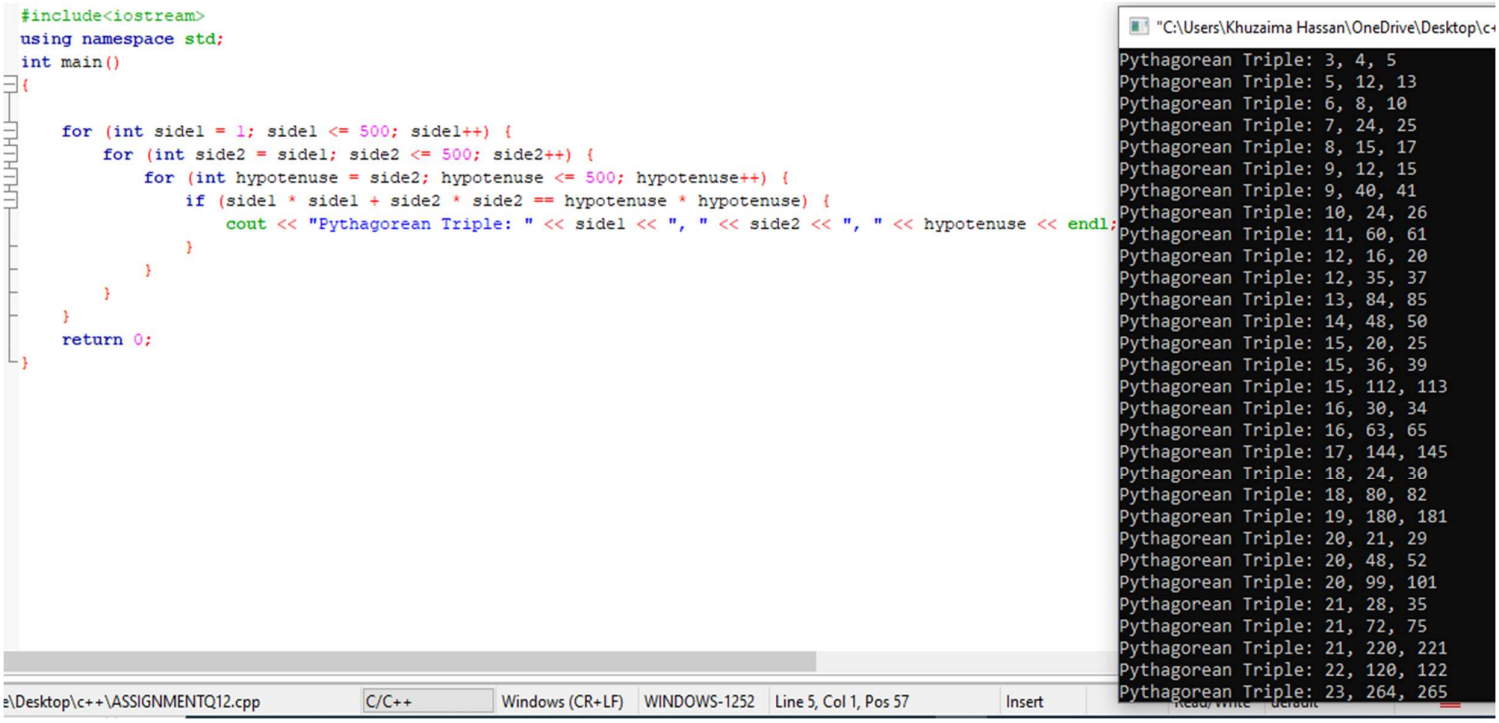
```
"C:\Users\Khuzaima Hassan\OneDrive\Desktop\c++\ASSIGNMENTQ11.exe"
**********
*********
********
*******
******
*****
****
***
**
*

Process returned 0 (0x0)   execution time : 0.061 s
Press any key to continue.
```

Q12: A right triangle can have sides that are all integers. A set of three integer values for the sides of a right triangle is called a Pythagorean triple. These three sides must satisfy the relationship that the sum of the squares of two of the sides is equal to the square of the hypotenuse. Find all Pythagorean triples for `side1`, `side2` and `hypotenuse` all no larger than 500. Use a triple-nested for loop that tries all possibilities.

CODE:

```
#include<iostream>

using namespace std;

int main()

{


   for (int side1 = 1; side1 <= 500; side1++) {

      for (int side2 = side1; side2 <= 500; side2++) {

         for (int hypotenuse = side2; hypotenuse <= 500; hypotenuse++) {

            if (side1 * side1 + side2 * side2 == hypotenuse * hypotenuse) {

               cout << "Pythagorean Triple: " << side1 << ", " << side2 << ", " << hypotenuse <<
endl;

            }

         }

      }

   }

   return 0;

}
```

OUTPUT SCREEN:



Q13: Consider the following code: Explain the output and any errors if any

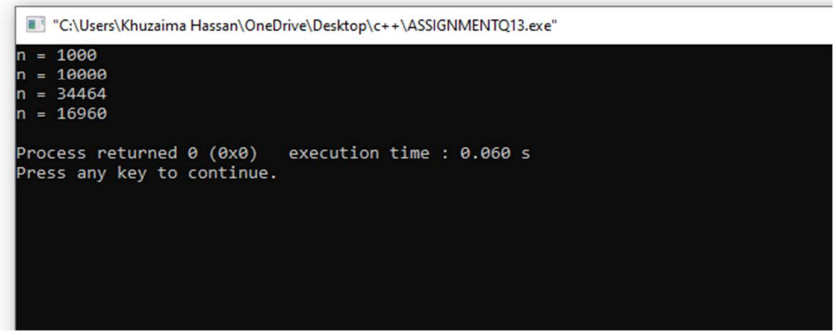
```
#include <iostream>
using namespace std;
int main()
{
unsigned short n=1000;
cout << "n = " << n << endl;
n *= 10;
cout << "n = " << n << endl;
n *= 10;
cout << "n = " << n << endl;
n *= 10;
cout << "n = " << n << endl;
return 0;
}
```

The output is

```
n = 1000
n = 10000
n = 34464
n = 16960
```

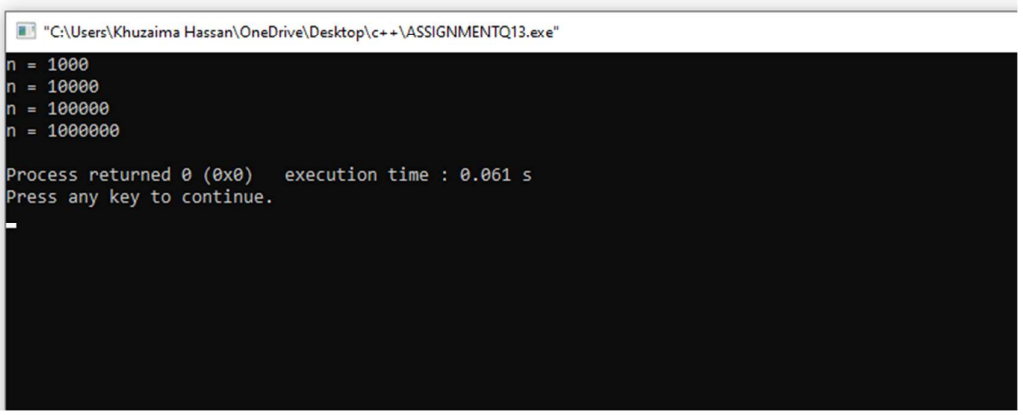
ERROR ONE:

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

int main()
{
    unsigned  short int n = 1000;//short int cannot used bcz it has 16 bits which is not enough for this program so it will show garbage value
    cout << "n = " << n << endl;

    n *= 10;
    cout << "n = " << n << endl;

    n *= 10;
    cout << "n = " << n << endl;

    n *= 10;
    cout << "n = " << n << endl;

    return 0;
}
```



CORRECT ONE:

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

int main()
{
    unsigned  int n = 1000;//short int cannot used bcz it has 16 bits which is not enough for this program so it will show garbage value
    cout << "n = " << n << endl;

    n *= 10;
    cout << "n = " << n << endl;

    n *= 10;
    cout << "n = " << n << endl;

    n *= 10;
    cout << "n = " << n << endl;

    return 0;
}
```



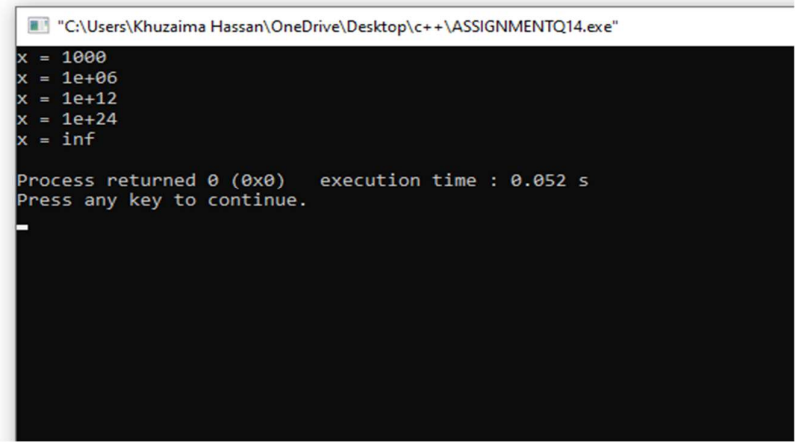
Q.14 Consider the following code and its output. Explain any errors.

```
#include <iostream>
using namespace std;
int main()
{
float x=1000.0;
cout << "x = " << x << endl;
x *= x; // multiplies n by itself; i.e.,it squares x
cout << "x = " << x << endl;
x *= x; // multiplies n by itself; i.e.,it squares x
cout << "x = " << x << endl;
x *= x; // multiplies n by itself; i.e.,it squares x
cout << "x = " << x << endl;
x *= x; // multiplies n by itself; i.e.,it squares x
cout << "x = " << x << endl;

return 0;
}
```

```
x = 1000
x = 1e+06
x = 1e+12
x = 1e+24
x = inf
```

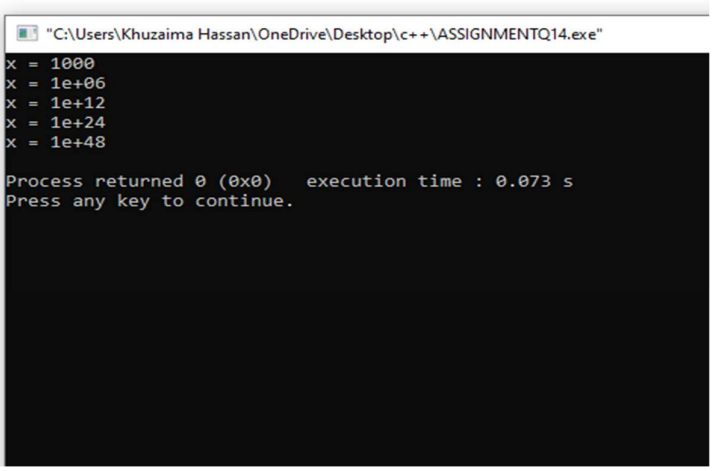
ERROR ONE:

```
#include <iostream>
using namespace std;
int main()
{
    float x=1000.0;
    cout << "x = " << x << endl;
    x *= x; // multiplies n by itself; i.e.,it squares x
    cout << "x = " << x << endl;
    x *= x; // multiplies n by itself; i.e.,it squares x
    cout << "x = " << x << endl;
    x *= x; // multiplies n by itself; i.e.,it squares x
    cout << "x = " << x << endl;
    x *= x; // multiplies n by itself; i.e.,it squares x
    cout << "x = " << x << endl;
    return 0;
}
```



CORRECT ONE:

```
#include <iostream>
using namespace std;
int main()
{
    double x=1000.0;
    cout << "x = " << x << endl;
    x *= x; // multiplies n by itself; i.e.,it squares x
    cout << "x = " << x << endl;
    x *= x; // multiplies n by itself; i.e.,it squares x
    cout << "x = " << x << endl;
    x *= x; // multiplies n by itself; i.e.,it squares x
    cout << "x = " << x << endl;
    x *= x; // multiplies n by itself; i.e.,it squares x
    cout << "x = " << x << endl;
    return 0;
}
```



Q.16 (Body Mass Index Calculator)
Create a BMI calculator application that reads the user's weightin pounds and height in inches (or, if you prefer, the user's weight in kilograms and height in meters), then calculates and displays the user's body mass index. Also, the application should display the following information from the Department of Health and Human Services/National Institutes of Health so the user can evaluate his/her BMI

BMI VALUES
Underweight: less than 18.5
Normal: between 18.5 and 24.9
Overweight: between 25 and 29.9
Obese: 30 or greater

CODE:

```
#include<iostream>

using namespace std;

int main()

{

   float weight,height,bmi;

   cout<<"  \tBMI CALCULATOR AND BMI CATEGORY"<<endl;

   cout<<"Enter your weight in kilograms: ";

   cin>>weight;

   cout<<"Enter your height in meters: ";

   cin>>height;

   bmi=weight/(height*height);

   cout<<"\t\t Your BMI is: "<<bmi<<endl;

   if (bmi<18.5)

   {

      cout<<"\t\t BMI CATEGORY: UNDERWEIGHT";

   }

   if (bmi>=18.5&&bmi<=24.9)

   {

      cout<<"\t\t BMI CATEGORY: NORMAL";

   }

   if (bmi>=25&&bmi<=29.9)

   {

      cout<<"\t\t BMI CATEGORY: OVERWEIGHT";

   }
```

```
    if (bmi>=30)
    {
       cout<<"\t\t BMI CATEGORY: OBESITY";
    }
    // Display BMI values
    cout<<endl<<"The following information from the Department of Health and Human
Services/National Institutes of Health"<<endl;
    cout << "\tBMI VALUES" << endl;
    cout << "Underweight: less than 18.5" << endl;
    cout << "Normal: between 18.5 and 24.9" << endl;
    cout << "Overweight: between 25 and 29.9" << endl;
    cout << "Obesity: 30 or greater" << endl;

    return 0;
}
```

OUTPUT SCREEN:

```
#include<iostream>
using namespace std;
int main()
{
    float weight,height,bmi;
    cout<<"  \tBMI CALCULATOR AND BMI CATEGORY"<<endl;
    cout<<"Enter your weight in kilograms: ";
    cin>>weight;
    cout<<"Enter your height in meters: ";
    cin>>height;
    bmi=weight/(height*height);
    cout<<"\t\t Your BMI is: "<<bmi<<endl;
    if (bmi<18.5)
    {
        cout<<"\t\t BMI CATEGORY: UNDERWEIGHT";
    }
    if (bmi>=18.5&&bmi<=24.9)
    {
        cout<<"\t\t BMI CATEGORY: NORMAL";
    }
    if (bmi>=25&&bmi<=29.9)
    {
        cout<<"\t\t BMI CATEGORY: OVERWEIGHT";
    }
    if (bmi>=30)
    {
        cout<<"\t\t BMI CATEGORY: OBESITY";
    }
    // Display BMI values
    cout<<endl<<"The following information from the Department of Health and Human Services/National Institutes of Health"<<endl;
    cout << "\tBMI VALUES" << endl;
    cout << "Underweight: less than 18.5" << endl;
    cout << "Normal: between 18.5 and 24.9" << endl;
    cout << "Overweight: between 25 and 29.9" << endl;
    cout << "Obesity: 30 or greater" << endl;

    return 0;
}
```

```
"C:\Users\Khuzaima Hassan\OneDrive\Desktop\c++\ASSIGNMENTQ16.exe"
       BMI CALCULATOR AND BMI CATEGORY
Enter your weight in kilograms: 60
Enter your height in meters: 1.7272
                Your BMI is: 20.1125
                BMI CATEGORY: NORMAL
The following information from the Department of Health and Human Services/National Institutes of Healt
       BMI VALUES
Underweight: less than 18.5
Normal: between 18.5 and 24.9
Overweight: between 25 and 29.9
Obesity: 30 or greater

Process returned 0 (0x0)   execution time : 9.786 s
Press any key to continue.
```

Q.17 (Printing the Decimal Equivalent of a Binary Number) Input an integer containing only 0s and 1s (i.e., a "binary" integer) and print its decimal equivalent. Use the remainder and division operators to pick off the "binary" number's digits one at a time from right to left.

CODE:

```
#include <iostream>
using namespace std;
int main()
{
   int binary,decimal = 0,base = 1;
   cout<<"Decimal Equivalent of a Binary Number"<<endl;
   cout << "Enter a binary number: ";
   cin >> binary;
   while (binary > 0)
      {
      int digit = binary % 10;
      decimal += digit * base;
      binary /= 10;
      base *= 2;
   }
   cout << "Decimal equivalent: " << decimal << endl;
   return 0;
}
```

OUTPUT SCREEN:

```
#include <iostream>
using namespace std;
int main()
{
    int binary,decimal = 0,base = 1;
    cout<<"Decimal Equivalent of a Binary Number"<<endl;
    cout << "Enter a binary number: ";
    cin >> binary;
    while (binary > 0)
        {
        int digit = binary % 10;
        decimal += digit * base;
        binary /= 10;
        base *= 2;
    }
    cout << "Decimal equivalent: " << decimal << endl;

    return 0;
}
```

```
"C:\Users\Khuzaima Hassan\OneDrive\Desktop\c++\ASSIGNMENTQ17.exe"
Decimal Equivalent of a Binary Number
Enter a binary number: 10101101
Decimal equivalent: 173

Process returned 0 (0x0)   execution time : 82.976 s
Press any key to continue.
```