

COMPUTER AND PROGRAMMING (EE-163)

ASSIGNMENT

Q1) Write a program to print the following text on the console screen.

Welcome to C++

Ans) #include <iostream>

using namespace std;

int main ()

{

cout << "Welcome to C++" << endl;

return 0;

}

Q2) Using escape sequences to print the text in following fashion.

Welcome

to

C++

Ans) #include <iostream>

using namespace std;

int main ()

{

cout << "Welcome \n to \n C++" << endl;

return 0;

}

Q3) Write a program that takes two integer numbers from the user and print their sum on console screen.

Ans)

```
#include <iostream>
using namespace std;
int main ()
{
    int a, b;
    cout<<"Enter First Integer: ";
    cin>>a;
    cout<<"Enter Second Integer: ";
    cin>>b;
    cout<<"The Sum of "<<a<<" And "<<b<<" Is "<<a+b<<endl;
    return 0;
}
```

Q4) What are Arithmetic operators? Write the rule of precedence for the Arithmetic operators.

Ans) Arithmetic Operators are the operators used to perform arithmetical/mathematical operations on operands. Examples: (+, -, *, /, %).

C++ applies the operators in arithmetic expressions in a precise order determined by these rules of operator precedence.

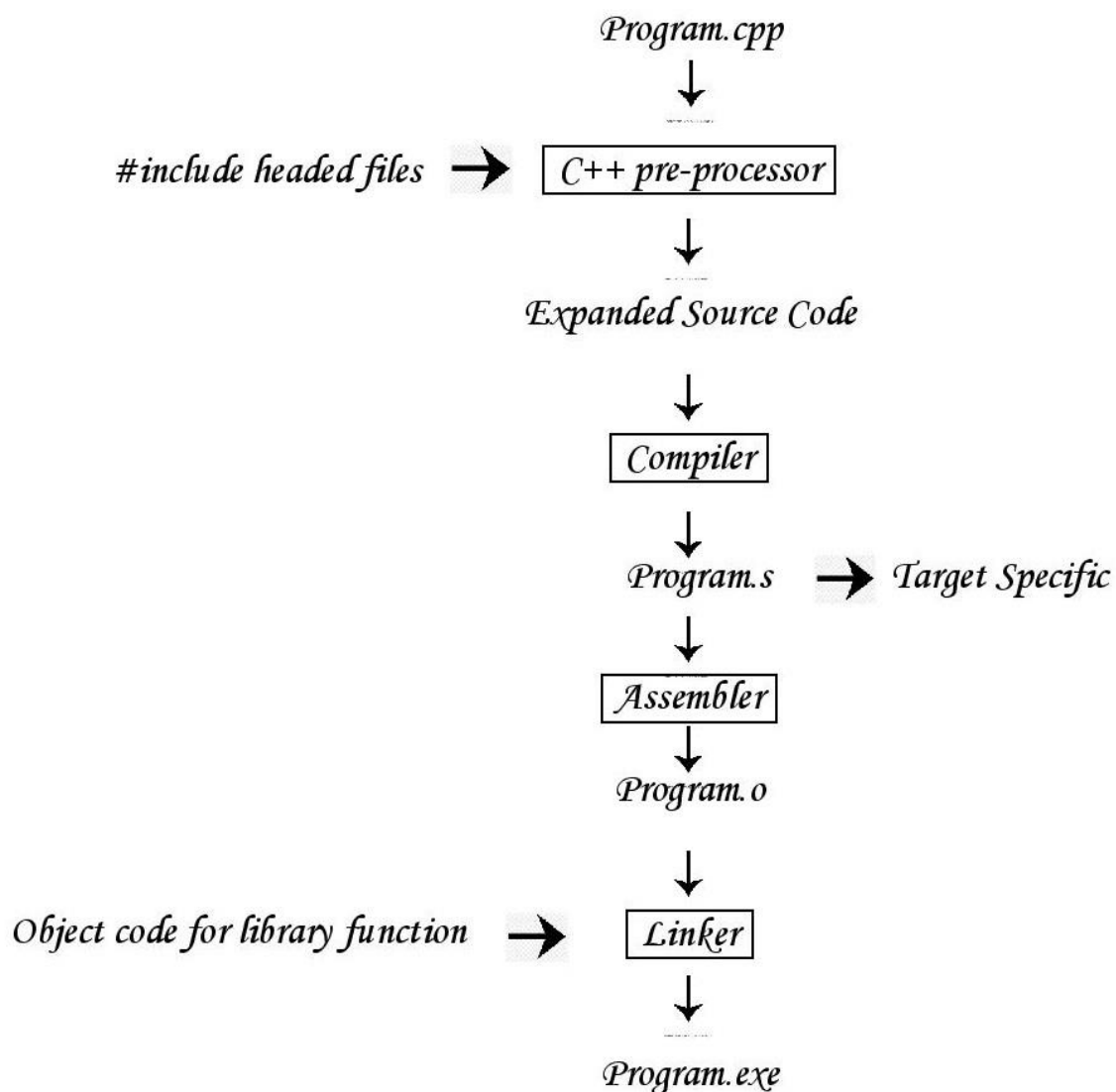
- 1) Operators in expression contained within pair of parentheses are evaluated first. Parentheses are said to be at the highest level of precedence. In case of nested parentheses such as, $(a^*(b+c))$, the operators in the innermost pair of parentheses are applied first.
- 2) Multiplication, division and modulus operations are applied next. If an expression contains several multiplication, division and modulus operations, operators are applied left to right. Multiplication, division and modulus are said to be on the same level of precedence.

3) Addition and subtraction operations are applied last. If an expression contains several addition and subtraction operations, operators are applied from left to right. Addition, and subtraction also have same level of precedence.

Q5) How C++ program is compiled into an executable?

Ans) The compilation process consist of four steps: -

- 1) The C++ processor copies the content of the included header files into the source code file, generates macro code, and replaces symbolic constant with defined using “# define” with their values.
- 2) The expanded source code file produced by pre-processor directive is compiled into the assembly language for the platform.
- 3) The assembles code generated by the compiler is assembled into the object code for the platform.
- 4) The object code file generated by the assembles is linked together with the object code file for any other library functions used to produce an executable file.



Q6) Write a program that take two numbers and operation to perform on those numbers as input from the user and print the result of operation on console screen. (Hint: Use if-statements to identify the required operation).

Ans)

```
#include <iostream>
using namespace std;
int main (void)
{
    int a, b;
    char c;
    cout<<"Enter First Integer: ";
    cin>>a;
    cout<<"Enter Second Integer: ";
    cin>>b;
    cout<<"Enter Operation to Perform: ";
    cin>>c;
    if (c=='/')
    {
        cout<<"Result is: "<<a/b<<endl;
    }
    if (c=='*')
    {
        cout<<"Result is: "<<a*b<<endl;
    }
    if (c=='%')
    {
        cout<<"Result is: "<<a%b<<endl;
    }
}
```

```

    if (c=='+')
    {
        cout<<"Result is: "<<a+b<<endl;
    }
    if (c=='-')
    {
        cout<<"Result is: "<<a-b<<endl;
    }
    else
    {
        cout<<"Invalid Operation!"<<endl;
    }
}

```

Q7) Write a program that asks user to enter two integers, obtains the numbers from the user, and then prints the larger number followed by the words "is larger." If the numbers are equal, print the message "These numbers are equal."

Ans)

```

#include <iostream>
using namespace std;
int main (void)
{
    int a, b;
    cout<<"Enter First Integer: ";
    cin>>a;
    cout<<"Enter Second Integer: ";
    cin>>b;
    if (a==b)
    {

```

```

    cout<<"These numbers are equal."<<endl;
}
else if (a>b)
{
    cout<<a<<" is larger"<<endl;
}
else
{
    cout<<b<<" is larger"<<endl;
}
}

```

Q8) 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

Ans)

```

#include <iostream>
using namespace std;
int main (void)
{
    int a, b, c;
    cout<<"Input three different integers: ";
    cin>>a>>b>>c;
    cout<<"Sum is "<<a+b+c<<endl;
    cout<<"Average is "<<(a+b+c)/3<<endl;
}

```

```

cout<<"Product is "<<a*b*c<<endl;
if ((a<b) && (a<c))
{
    cout<<"Smallest is "<<a<<endl;
}
else if ((b<a) && (b<c))
{
    cout<<"Smallest is "<<b<<endl;
}
else if ((c<a) && (c<b))
{
    cout<<"Smallest is "<<c<<endl;
}
if ((a>b) && (a>c))
{
    cout<<"Largest is "<<a<<endl;
}
else if ((b>a) && (b>c))
{
    cout<<"Largest is "<<b<<endl;
}
else if ((c>a) && (c>b))
{
    cout<<"Largest is "<<c<<endl;
}
}

```

Q9: Write a program that reads an integer and determines and prints whether it is odd or even.

Ans)

```
#include <iostream>
using namespace std;
int main (void)
{
    int a;
    cout<<"Input Any Integer: ";
    cin>>a;
    if (a%2==0)
    {
        cout<<"The Number Is Even"<<endl;
    }
    else
    {
        cout<<"The Number Is Odd"<<endl;
    }
}
```

Q10: 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

Ans)

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



```

{
    int a, b, c, d, e, f;
    cout<<"Input Any Five Digit Integer: ";
    cin>>a;
    b=a%10;
    c=(a/10) %10;
    d=(a/100) %10;
    e=(a/1000) %10;
    f=(a/10000);
    cout<<f<<" "<<e<<" "<<d<<" "<<c<<" "<<b;
}

```

Q11) 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

Ans)

```
#include <iostream>
```

```
using namespace std;
```

```
int main (void)
{
    int hours;
    float salary, rate;
    cout<<"Enter hours worked (-1 to end): ";
    cin>>hours;
    while (hours>0)
    {
        cout<<"Enter hourly rate of the employee: ";
        cin>>rate;
        if (hours<=40)
        {
            salary=hours*rate;
        }
        else
        {
            salary=hours*rate+(hours-40)*rate*0.5;
        }
        cout<<"Salary is "<<salary<<" Rs."<<endl;
        cout<<"Enter hours worked (-1 to end): ";
        cin>>hours;
    }
}
```

Q12) 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

Ans)

```
#include <iostream>
using namespace std;
int main ()
{
    int a, fact=1;
    cout<<"Enter an integer: ";
    cin>>a;
    while (a>0)
    {
        fact=fact*a;
        cout<<a<<" x ";
        a--;
    }
    cout<<"\b\b= "<<fact;
}
```

Q13) 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

Ans) #include <iostream>

using namespace std;

```
int main ()
{
    int a, t1=0, t2=1, b;
    cout<<"Enter number of elements: ";
    cin>>a;
    for (int i=1; i<=a; i++)
    {
        if (i==1)
        {
            cout<<t1;
            continue;
        }
        if (i==2)
        {
            cout<<" "<<t2<<" ";
            continue;
        }
        b=t1+t2;
        t1=t2;
        t2=b;
        cout<<b<<" ";
    }
}
```

Q14) Write a program that reads three non-zero double values and determines and prints whether they could represent 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.

Ans)

```
#include <iostream>
```

```
using namespace std;
```

```
int main (void)
```

```
{
```

```
    while (1)
```

```
    {
```

```
        double a, b, c;
```

```
        cout<<"Enter lengths of three sides: ";
```

```
        cin>>a>>b>>c;
```

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

```
        {
```

```
            cout<<"They are sides of triangle. \n\n";
```

```
        }
```

```
        else
```

```
        {
```

```
            cout<<"They are not sides of triangle. \n\n";
```

```

    }
}
}

```

Q15) 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 represent right triangle.

Enter length of three sides: 4 5 6.403

The sides don't represent right triangle.

Enter length of three sides: 4 5 6.4031

The sides represent right triangle.

Ans)

```
#include <iostream>
```

```
using namespace std;
```

```
int main (void)
```

```
{
```

```
    while (1)
```

```
    {
```

```
        double a, b, c;
```

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

```
        cin>>a>>b>>c;
```

```
        if ((a*a+b*b-c*c<=0.001) || (a*a+c*c-b*b<=0.001) || (b*b+c*c-a*a<=0.001))
```

```
        {
```

```
            cout<<"The sides represents right triangle. \n\n";
```

```
        }
```

```
    else
```

```

{
    cout<<"The sides don't represents right triangle. \n\n";
}
}
}

```

Q16) Write a program that ask user to input a floating point number and computes exponential of that number using Taylor series as below: -

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots$$

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

Ans)

```

#include <iostream>
#include <cmath>
using namespace std;
int fact (int x);
int main (void)
{
    double x, a=1.0;
    int n;
    cout<<"Enter a value whose exponential needs to be evaluated: ";
    cin>>x;
    cout<<"Enter number of terms for evaluation: ";
    cin>>n;
    for (int i=1; i<=n; i++)

```

```

{
    a=a+pow (x i)/fact(i);
}

cout<<"Result is: "<<a;
}

int fact (int x)
{
    int fact=1;
    while (x>0)
    {
        fact=fact*x;
        x--;
    }
    return fact;
}

```

Q17) Write a program that ask user to input angle in radians and computes its sine using Taylor series as below: -

$$\sin(x) = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n+1}$$

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

Ans) #include <iostream>

using namespace std;

int main (void)

{


```

double x, sum, t;
int n, i;
cout<<"Enter a value for sin evaluation: ";
cin>>x;
cout<<"Enter number of terms in the summation: ";
cin>>n;
t=x;
sum=x;
for (i=1; i<=n; i++)
{
    t=(t*(-1)*x*x)/(2*i*(2*i+1));
    sum=sum+t;
}
cout<<"Result is: "<<sum<<endl;
}

```

Q18) Write a program that prints the following pattern as shown below: -

```

*               ****               ****               *
**             ****               ****               **
***           ****               ****               ***
****         ****               ****               ****
*****       ****               ****               *****
*****      ****               ****               *****
*****     ****               ****               *****
*****    ****               ****               *****
*****   ****               ****               *****
*****  ****               ****               *****
***** ****               ****               *****
***** *               ****               *****

```

You can make separate programs for all these four patterns. Extra Credit: If all the four patterns are printed in a single program. [Hint: Use nested for loop].

Ans) DIAGRAM 1: -

```
#include <iostream>
using namespace std;
int main (void)
{
    for (int i=1; i<=10; i++)
    {
        for (int j=1; j<=i; j++)
        {
            if (j<=i)
            {
                cout<<"*";
            }
            else
            {
                cout<<" ";
            }
        }
        cout<<endl;
    }
}
```

➤ **DIAGRAM 2: -**

```
#include <iostream>
using namespace std;
int main (void)
{
    for (int i=1; i<=10; i++)
    {
        for (int j=10; j>=i; j--)
        {
            if (j>=i)
            {
                cout<<"*";
            }
            else
            {
                cout<<" ";
            }
        }
    }
}
```

```

    }
}
cout<<endl;
}
}

```

➤ **DIAGRAM 3: -**

```

#include <iostream>
using namespace std;
int main (void)
{
    for (int i=1; i<=10; i++)
    {
        for (int j=1; j<i; j++)
        {
            cout<<" ";
        }
        for (int j=i; j<=10; j++)
        {
            cout<<"★";
        }
        cout<<endl;
    }
}

```

➤ **DIAGRAM 4: -**

```

#include <iostream>
using namespace std;
int main (void)
{
    for (int i=1; i<=10; i++)
    {
        for (int j=10-i; j>0; j--)
        {
            cout<<" ";
        }
        for (int j=1; j<=i; j++)
        {
            cout<<"★";
        }
    }
}

```

```

    }
    cout<<endl;
}
}

```

➤ OVERALL CODE OF DIAGRAM: -

```

#include <iostream>
using namespace std;
int main (void)
{
    for (int i=1; i<=10; i++)
    {
        for (int j=1; j<=i; j++)
        {
            if (j<=i)
            {
                cout<<"* ";
            }
            else
            {
                cout<<" ";
            }
        }
        cout<<endl;
    }
    cout<<endl<<endl;
    for (int i=1; i<=10; i++)
    {
        for (int j=10; j>=i; j--)
        {
            if (j>=i)
            {
                cout<<"* ";
            }
            else
            {
                cout<<" ";
            }
        }
    }
}

```

```

    cout<<endl;
}
cout<<endl<<endl;
for (int i=1; i<=10; i++)
{
    for (int j=1; j<i; j++)
    {
        cout<<" ";
    }
    for (int j=i; j<=10; j++)
    {
        cout<<"★";
    }
    cout<<endl;
}
cout<<endl<<endl;
for (int i=1; i<=10; i++)
{
    for (int j=10-i; j>0; j--)
    {
        cout<<" ";
    }
    for (int j=1; j<=i; j++)
    {
        cout<<"★";
    }
    cout<<endl;
}
}

```

Q19) Write a program that prints the following diamond shape. [Hint: Use nested for loop].

```

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

Ans)

```
#include <iostream>
using namespace std;
int main (void)
{
    for (int i = 1; i <= 5; i++)
    {
        for (int j = 1; j <= 9; j++)
        {
            if ((j >= (6-i)) && (j <= (i+4)))
            {
                cout << " * ";
            }
            else
            {
                cout << " ";
            }
        }
        cout << endl;
    }
    for (int i = 6; i <= 9; i++)
    {
        for (int j = 1; j <= 9; j++)
        {
            if ((j >= i-4) && (j <= 14-i))
            {
                cout << " * ";
            }
        }
    }
}
```

```

        else
        {
            cout << " ";
        }
    }
    cout << endl;
}
}

```

Q20) Modify the program in Q19 to read an odd number in the range of 1 to 19 to specify the number of rows in the diamond, then display a diamond of appropriate size.

Ans)

```

#include <iostream>
using namespace std;
int main(void)
{
    int i,j,r;
    cout<<"Input number of rows: ";
    cin>>r;
    if ((r>=1) && (r<=19) && (r%2!=0))
    {
        for(i=0;i<=r/2+1;i++)
        {
            for(j=1;j<=r/2+1-i;j++)
            {
                cout<<" ";
            }
            for(j=1;j<=2*i-1;j++)
            {
                cout<<"*",
            }
            cout<<endl;
        }
        for(i=r/2;i>=1;i--)
        {
            for(j=1;j<=r/2+1-i;j++)
            {
                cout<<" ";
            }

```

```

    }
    for(j=1;j<=2*i-1;j++)
    {
        cout<<"*";
    }
    cout<<endl;
}
}
else
{
    cout<<"ERROR!";
}
}

```

Q21) *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.*

Ans)

```

#include <iostream>
using namespace std;
int main(void)
{
    for (int a = 1; a <=500; a++)
    {
        for (int b = 1; b <=500; b++)
        {
            for (int c = 1; c <=500; c++)
            {
                if ((a*a+b*b==c*c) || (c*c+b*b==a*a) || (a*a+c*c==b*b))
                {
                    cout << a << " " << b << " " << c <<endl;
                }
            }
        }
    }
}

```