**WEEK-7**

**#1 Write a C++ Program for Count vowels String Using Pointer.**

This is the required code:

#include <iostream>

using namespace std;

int countvowels(char \*str)

{

    int cnt =0;

    while (\*str != '\0')

    {

        if (\*str == 'a' || \*str == 'e' || \*str == 'i' || \*str == 'o' || \*str == 'u' || \*str == 'A' || \*str == 'E' || \*str == 'I' || \*str == 'O' || \*str == 'U')

        {

            cnt++;

        }

        \*str++;

    }

    return cnt;

}

int main()

{

    char strng[100];

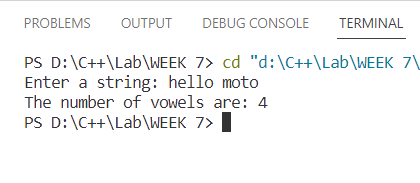
    int count;

    cout << "Enter a string: ";

    cin.getline(strng, 100);

    count = countvowels(strng);

    cout <<"The number of vowels are: "<< count;

Output:

**#2 Write a C++ Program for Length of String Using Pointer.**

This is the required program:

#include <iostream>

using namespace std;

int lnthStr(char \*str)

{

    int lnth = 0;

    while (\*str != '\0')

    {

        lnth++;

        \*str++;

    }

    return lnth;

}

int main()

{

    char strng[100];

    int length;

    cout << "Enter a string: ";

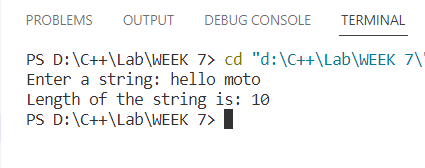
    cin.getline(strng, 100);

    length = lnthStr(strng);

    cout << "Length of the string is: " << length;

}

Output:



**#3 Write a C++ program using pointers to compute the sum, mean and standard deviation of all elements stored in an array of n real numbers.**

This the required code:

#include <iostream>

#include <cmath>

using namespace std;

double arrSum(double \*ar, int n)

{

    double sum = 0;

    for (int i = 0; i < n; i++)

        sum += ar[i];

    return sum;

}

double arrMean(double \*ar, int n)

{

    double sum = 0;

    for (int i = 0; i < n; i++)

            sum += ar[i];

    double mn = sum / n;

    return mn;

}

double sd(double \*ar, int n)

{

    double mean, smsqr = 0, mnsq;

    for (int i = 0; i < n; i++)

            smsqr += ar[i] \* ar[i];

    smsqr = smsqr / n;

    mean = arrMean(ar, n);

    mnsq = mean \* mean;

    return smsqr - mnsq;

}

int main()

{

    int n;

    cout << "Enter the size of the array: ";

    cin >> n;

    cout << "Enter the elements\n";

    double arr[n], mean = 0, sum = 0, std = 0;

    for (int i = 0; i < n; i++)

            cin >> arr[i];

    sum = arrSum(arr, n);

    mean = arrMean(arr, n);

    std = sqrt(sd(arr, n));

    cout << "The sum is " << sum << endl

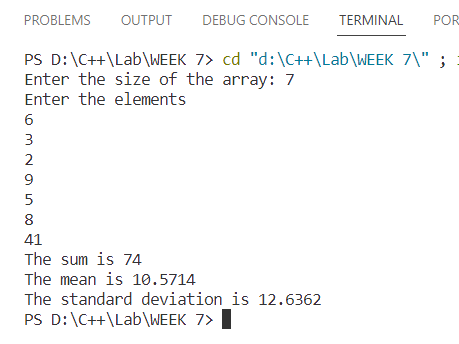
         << "The mean is "

         << mean << endl

         << "The standard deviation is " << std;

}

Output:



**#4 Write a C++ program to create three objects for a class named pntr\_obj with data members such as roll\_no & name. Create a member function set\_data() for setting the data values and print() member function to print which object has invoked it using the ‘this’ pointer.**

This is the required code:

#include <iostream>

using namespace std;

class pntr\_obj

{

    string name;

    int roll\_no;

public:

    void set\_data(int roll, const string &student\_name)

    {

        roll\_no = roll;

        name = student\_name;

    }

    void print()

    {

        cout << "Object " << this << " - Roll No: " << roll\_no << ", Name: " << name << endl;

    }

};

int main()

{

    pntr\_obj obj1, obj2, obj3;

    obj1.set\_data(11, "Moto");

    obj2.set\_data(12, "carlie");

    obj3.set\_data(13, "Hello");

    obj1.print();

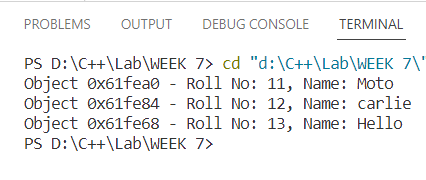
    obj2.print();

    obj3.print();

    return 0;

}

Output:



**#5 Develop a C++ program to find the greatest of two numbers using this pointer which returns the member variable.**

This is the required program:

#include <bits/stdc++.h>

using namespace std;

class GreatestNumFinder

{

private:

    int num1;

    int num2;

public:

    GreatestNumFinder(int a, int b) : num1(a), num2(b) {}

    int ReturnGreatest()

    {

        return this->num1 > this->num2 ? this->num1 : this->num2;

    }

};

int main()

{

    int a, b;

    cout << "Enter the first number: ";

    cin >> a;

    cout << "Enter the second number: ";

    cin >> b;

    GreatestNumFinder finder(a, b);

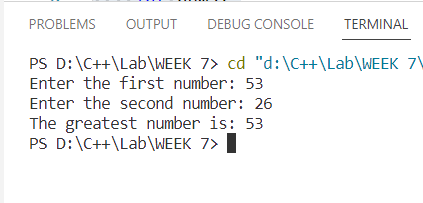
    int greatest = finder.ReturnGreatest();

    cout << "The greatest number is: " << greatest << endl;

    return 0;

}

Output:



**#6 Write a C++ program to implement flight class with data member as flightno., source destination and fare. Write a member function to display the flight information using this pointer.**

This is the required code:

#include <iostream>

using namespace std;

class flight

{

    int flight\_no;

    string source;

    string destination;

    double fare;

public:

    flight(int fl\_no, string src, string des, double fr)

    {

        flight\_no = fl\_no;

        source = src;

        destination = des;

        fare = fr;

    }

    void information()

    {

        cout << "Flight Information\n";

        cout << "Flight number is:- " << this->flight\_no<<endl<< "Source is:- " << this->source<<endl << "Destination is:- " << this->destination<<endl << "Fare is:- " << this->fare;

    }

};

int main()

{

    flight f1 = { 222,

                  "Russia",

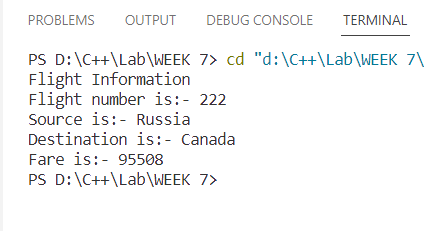
                  "Canada",

                  95508 };

    f1.information();

}

Output:



**#7 Write a C++ program to use this pointer and return the pointer reference.**

This is the required code:

#include <iostream>

using namespace std;

class Sample {

private:

    int data;

public:

    Sample(int value) : data(value) {}

    // Member function to return a reference to the current object

    Sample& returnThis() {

        return \*this;

    }

    // Member function to get a modifiable reference to data

    int& getData() {

        return data;

    }

};

int main() {

    Sample obj(42);

    // Using the returnThis() member function to get a reference to the current object

    Sample& ref = obj.returnThis();

    cout << "Original object's data: " << obj.getData() << endl;

    cout << "Referenced object's data: " << ref.getData() << endl;

    // Modify the data through the reference

    ref.getData() = 99;

    // Verify that the data is modified in the original object as well

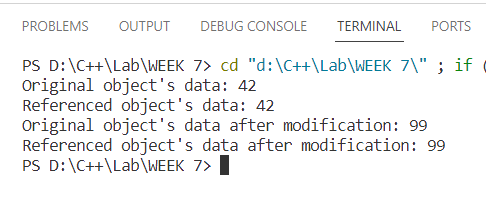
    cout << "Original object's data after modification: " << obj.getData() << endl;

    cout << "Referenced object's data after modification: " << ref.getData() << endl;

    return 0;

}

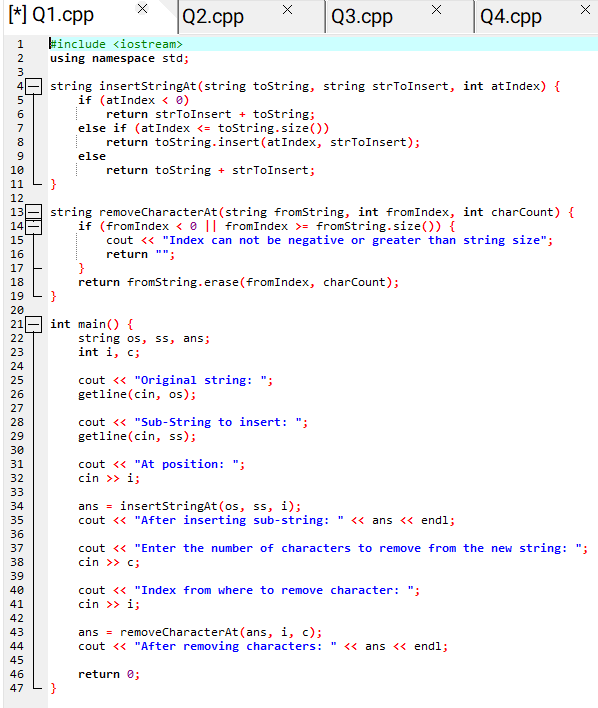
Output:



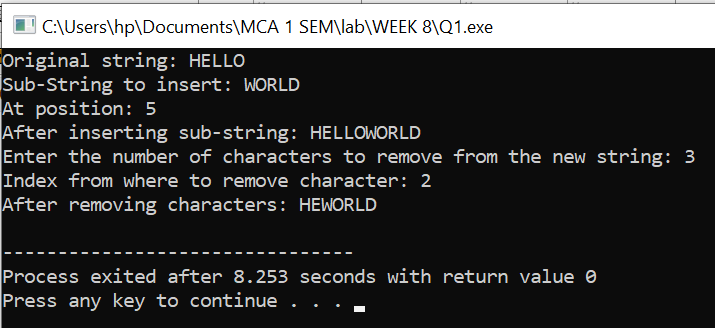
**Wekk-8**

**#1 Write a C++ program that uses functions to perform the following operations: i) To insert a sub string into a given main string from a given position. ii) To delete n characters from a given position in a given string.**

This is the required program:

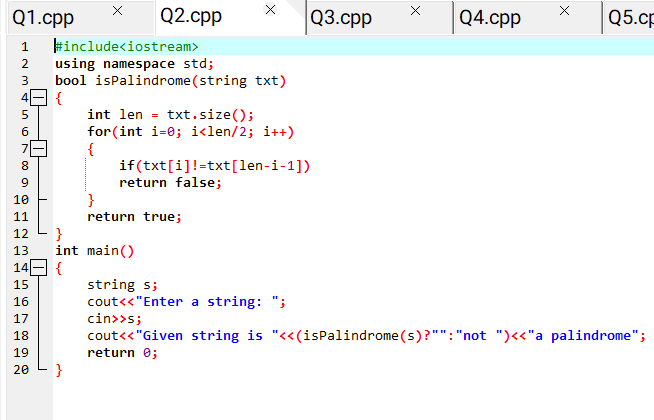


Output:

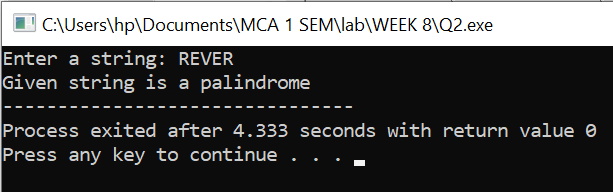


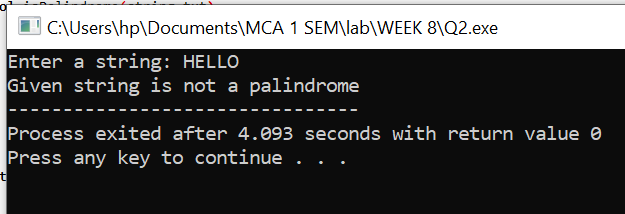
**#2 Write a C++ program to determine if the given string is a palindrome or not.**

This is the required code:



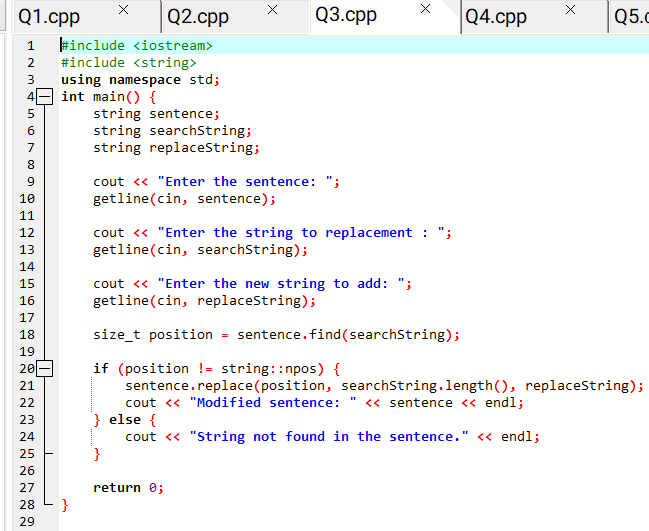
Output:



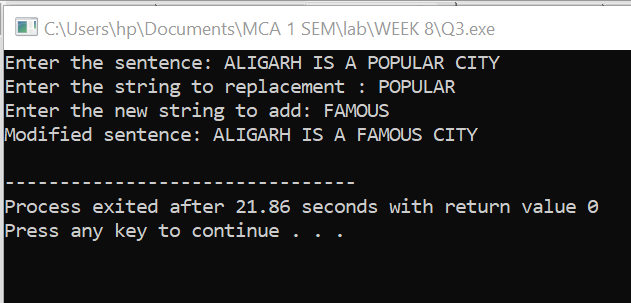


**#3 Write a C++ program to find a string within a sentence and replace it with another string.**

This is the required code:

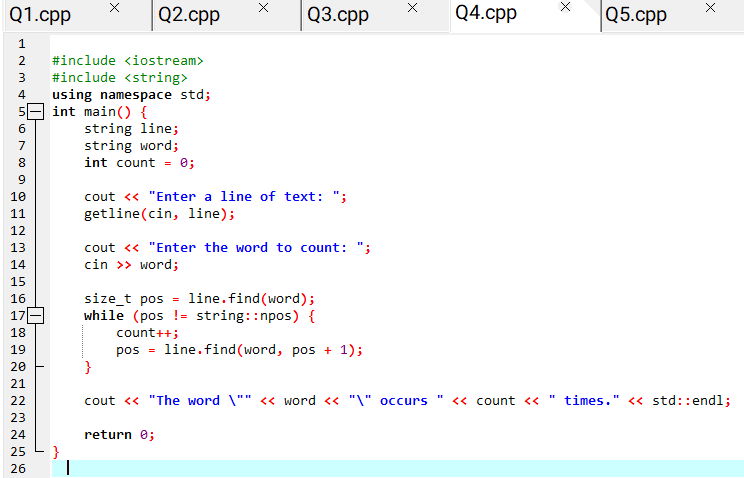


Output:

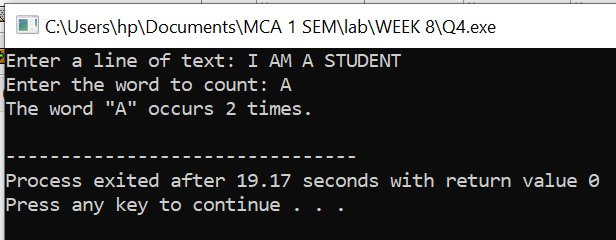


**#4 Write a C++ program that reads a line of text and counts all occurrence of a particular word.**

This is the required code:

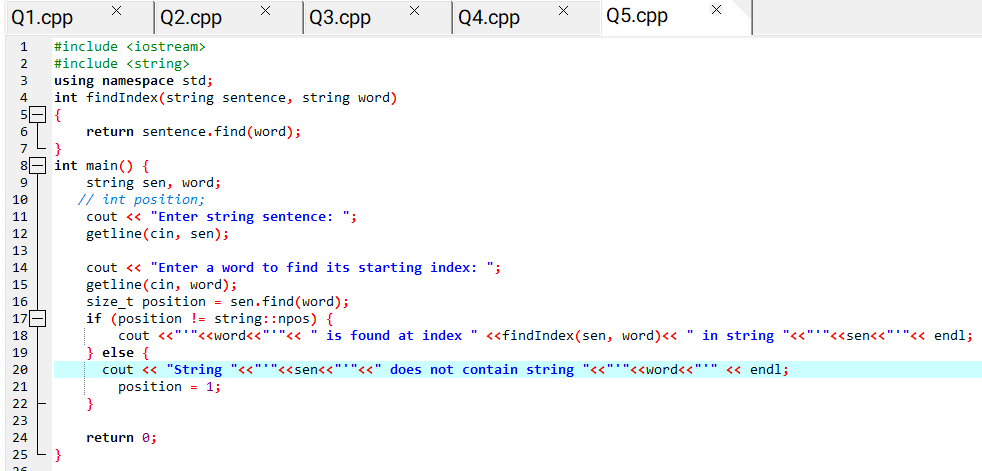


Output:



**#5 Write a C++ program that displays the position or index in the string S where the string T begins, or 1 if S doesn’t contain T.**

This is the required program:



Output:

