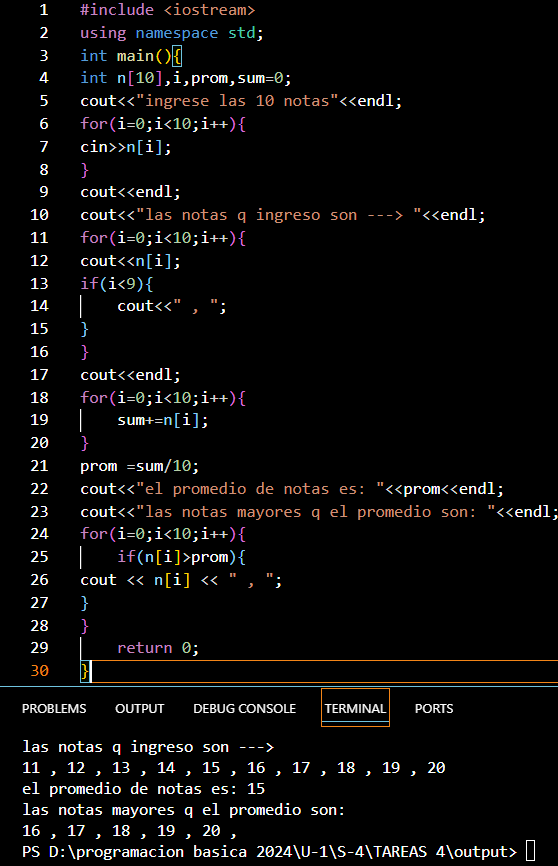
**TAREA 4 DE PROGRAMACION**

**Ejercicios propuestos:**

1. Manejo de Arreglos: Escribe un programa que solicite al usuario ingresar 10 números enteros. Luego, encuentra el promedio de los números ingresados y muestra cuántos números son mayores que el promedio.
2. Palíndromos: Desarrolla un programa que verifique si una palabra ingresada por el usuario es un palíndromo (se lee igual de izquierda a derecha que de derecha a izquierda). Usa un bucle while o for para comparar los caracteres.
3. Matriz Transpuesta: Crea un programa que solicite al usuario ingresar los elementos de una matriz cuadrada. Luego, calcula la matriz transpuesta e imprímela en la pantalla.
4. Sucesión de Collatz: Escribe un programa que tome un número entero positivo n como entrada y genere la secuencia de Collatz para ese número. La secuencia de Collatz se obtiene siguiendo estas reglas:
   1. Si n es par, divídelo por 2.
   2. Si n es impar, multiplícalo por 3 y suma 1.
   3. Repite el proceso hasta llegar al número 1.
5. Cifrado Cesar: Implementa un programa que realice un cifrado César. Solicita al usuario ingresar una cadena y un número (la clave del cifrado). Luego, cifra la cadena desplazando cada letra en la cadena por la cantidad especificada en la clave.
6. Juego de Preguntas y Respuestas: Crea un juego de preguntas y respuestas donde el programa haga preguntas al usuario y luego evalúe las respuestas. Mantén un contador de respuestas correctas e incorrectas.
7. Eliminación de Duplicados: Crea un programa que elimine los elementos duplicados de un vector. Utiliza un bucle for, break y continue según sea necesario.
8. Mayor Elemento en un Vector: Crea un programa que encuentre el mayor elemento en un vector de números enteros. Utiliza un bucle for y break para salir del bucle tan pronto como encuentres el mayor elemento.
9. Números Primos en un Rango: Escribe un programa que solicite al usuario dos números enteros (inicio y fin) y genere un vector con los números primos en ese rango. Utiliza continue para omitir los números no primos.
10. Contador de vocales: Desarrolla un programa que cuente la cantidad de a, e, i, o, u en una oración ingresada por el usuario.

**Ejer 1 (array de 10 notas y verificar si hay notas mayor q el promedio)**

1. Manejo de Arreglos: Escribe un programa que solicite al usuario ingresar 10 números enteros. Luego, encuentra el promedio de los números ingresados y muestra cuántos números son mayores que el promedio.
2. #include <iostream>
3. using namespace std;
4. int main(){
5. int n[10],i,prom,sum=0;
6. cout<<"ingrese las 10 notas"<<endl;
7. for(i=0;i<10;i++){
8. cin>>n[i];
9. }
10. cout<<endl;
11. cout<<"las notas q ingreso son ---> "<<endl;
12. for(i=0;i<10;i++){
13. cout<<n[i];
14. if(i<9){
15. cout<<" , ";
16. }
17. }
18. cout<<endl;
19. for(i=0;i<10;i++){
20. sum+=n[i];
21. }
22. prom =sum/10;
23. cout<<"el promedio de notas es: "<<prom<<endl;
24. cout<<"las notas mayores q el promedio son: "<<endl;
25. for(i=0;i<10;i++){
26. if(n[i]>prom){
27. cout << n[i] << " , ";
28. }
29. }
30. return 0;
31. }



**EJER 2 (Palíndromos)**

Palíndromos: Desarrolla un programa que verifique si una palabra ingresada por el usuario es un palíndromo (se lee igual de izquierda a derecha que de derecha a izquierda). Usa un bucle while o for para comparar los caracteres.

#include <iostream>

#include <string>

using namespace std;

int main() {

    string palabra;

    bool esPalindromo = true;

    cout << "Ingrese palabra: ";

    getline(cin,palabra);

    int longitud = palabra.size();

    for (int i = 0; i < longitud / 2; i++) {

        if (palabra[i] != palabra[longitud - 1 - i]) {

            esPalindromo = false;

            break;

        }

    }

    if (esPalindromo) {

        cout << "La palabra es palindroma" << endl;

    } else {

        cout << "La palabra no es palindroma" << endl;

    }

    return 0;

}



**Ejer 3 ( Transpuesta de una matriz )**

Crea un programa que solicite al usuario ingresar los elementos de una matriz cuadrada. Luego, calcula la matriz transpuesta e imprímela en la pantalla.

#include<iostream>

using namespace std;

int main(){

int i,j,f,c;

   cout << "Ingrese el número de filas: ";

    cin >> f;

    cout << "Ingrese el número de columnas: ";

    cin >> c;

int mtrix[f][c];

for(i=0;i<f;i++){

  for(j=0;j<c;j++){

cout<<"ingrese los valores de la matriz["<<i<<"] ["<<j<<"]: ";

cin>>mtrix[i][j];

  }

}

cout<<"la matriz es: \n";

for(i=0;i<f;i++){

  for(j=0;j<c;j++){

cout<<mtrix[i][j]<<" ";

  }

  cout<<"\n";

}

cout<<"la matriz transpuesta es: \n";

for(i=0;i<c;i++){

  for(j=0;j<f;j++){

cout<<mtrix[j][i]<<" ";

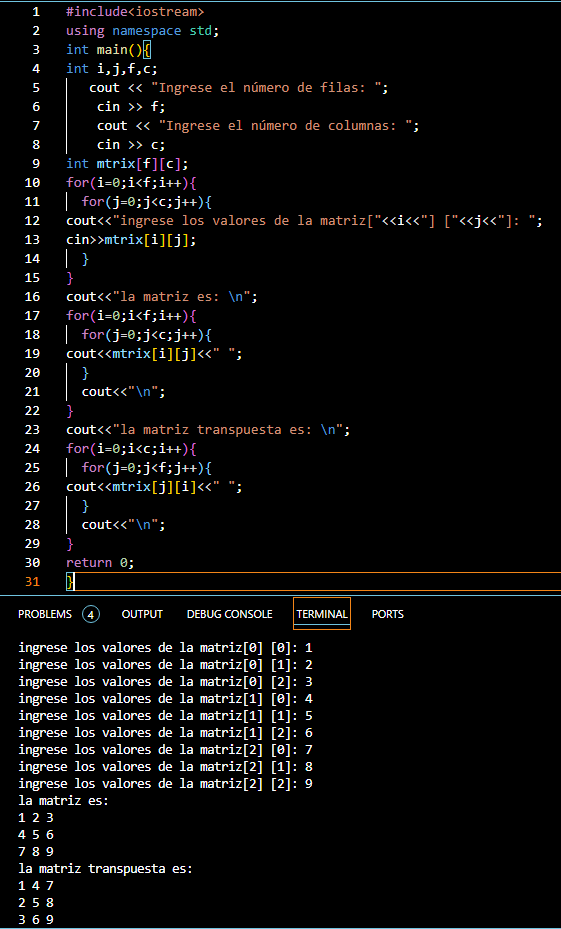
  }

  cout<<"\n";

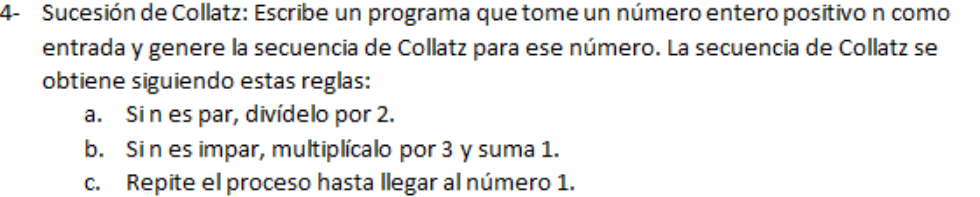
}

return 0;

}



**Ejer 4 (sucesión de collatz)**



#include <iostream>

using namespace std;

int main(){

int n;

cout<<"ingrese # \n"; cin >> n;

while(n!=1){

if (n%2==0){

cout<<"n es par ---> ";

n=(n/2);

cout<<n<<endl;

}

else{

cout<<"n es impar ---> ";

n=(n\*3)+1;

cout<<n<<endl;

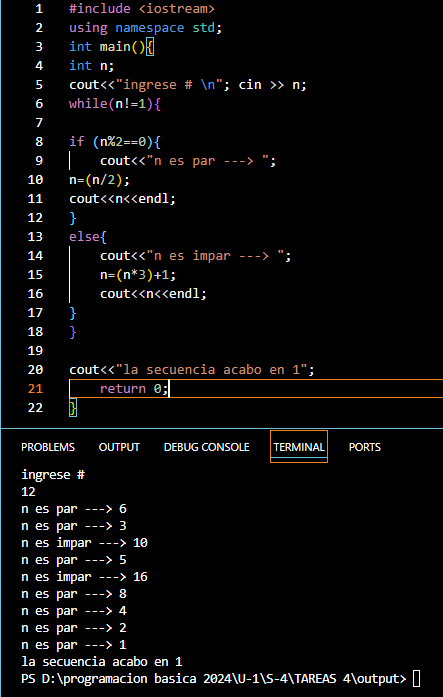
}

}

cout<<"la secuencia acabo en 1";

return 0;

}



**Ejer 5 ( Cifrado cesar )**

Cifrado Cesar: Implementa un programa que realice un cifrado César. Solicita al usuario ingresar una cadena y un número (la clave del cifrado). Luego, cifra la cadena desplazando cada letra en la cadena por la cantidad especificada en la clave.

**#include <iostream>**

**using namespace std;**

**string cifradocesar( string texto , int n ){**

**string cifrado=" ";**

**for(char c : texto ){**

**cifrado=cifrado + char (c + n);**

**}**

**return cifrado;**

**}**

**int main(){**

**string palabras;**

**int x;**

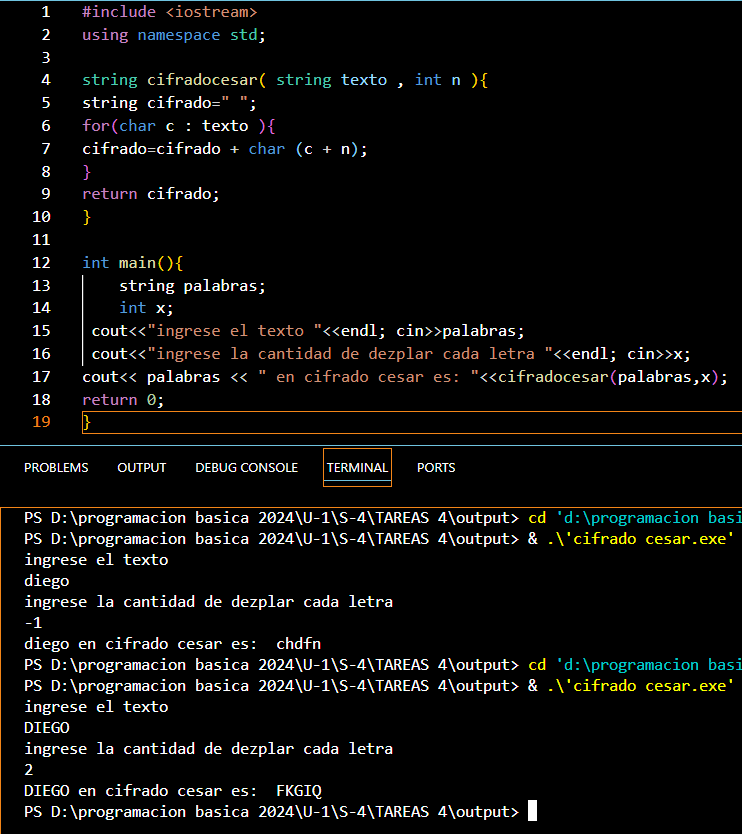
**cout<<"ingrese el texto "<<endl; cin>>palabras;**

**cout<<"ingrese la cantidad de dezplar cada letra "<<endl; cin>>x;**

**cout<< palabras << " en cifrado cesar es: "<<cifradocesar(palabras,x);**

**return 0;**

**}**



**Ejer 7 (Eliminar duplicados de un vector fijo)**

Eliminación de Duplicados: Crea un programa que elimine los elementos duplicados de un vector. Utiliza un bucle for, break y continue según sea necesario.

#include <iostream>

using namespace std;

int main() {

    int numeros[] = {1, 2, 3, 2, 4, 3, 5, 1, 6};

    int sinDuplicados[9];

    int contador = 0;

    for (int i = 0; i < 9; i++) {

        bool esDuplicado = false;

        for (int j = 0; j < i; j++) {

            if (numeros[i] == numeros[j]) {

                esDuplicado = true;

                break;

            }

        }

        int ocurrencias = 0;

        for (int k = 0; k < 9; k++) {

            if (numeros[i] == numeros[k]) {

                ocurrencias++;

            }

        }

        if (!esDuplicado && ocurrencias == 1) {

            sinDuplicados[contador] = numeros[i];

            contador++;

        }

    }

    cout << "nums sin duplicados: ";

    for (int i = 0; i < contador; i++) {

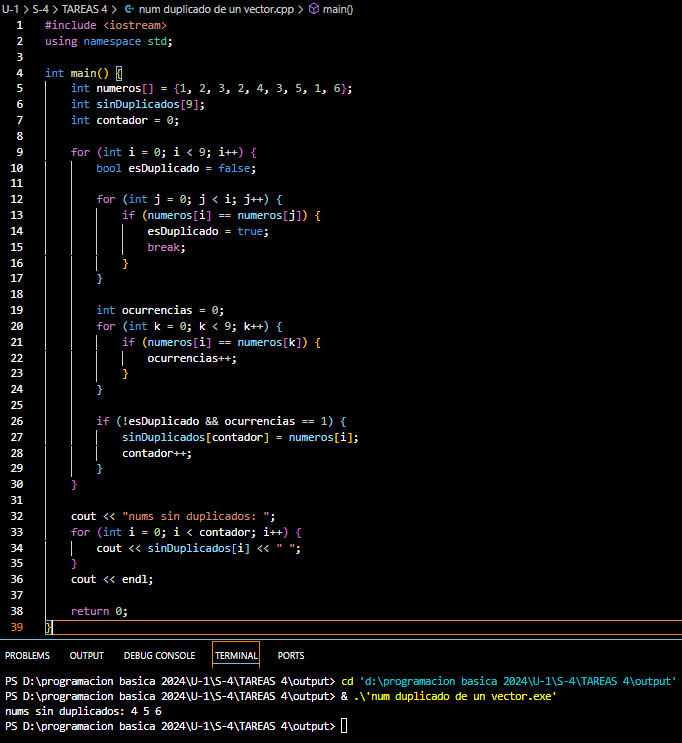
        cout << sinDuplicados[i] << " ";

    }

    cout << endl;

    return 0;

}



**Ejer 8 (Num mayor de un vector )**

Mayor Elemento en un Vector: Crea un programa que encuentre el mayor elemento en un vector de números enteros. Utiliza un bucle for y break para salir del bucle tan pronto como encuentres el mayor elemento.

#include <iostream>

#include <vector>

using namespace std;

int main(){

int n,i=0,nMAYOR;

cout<<"ingrese la cantidad q quiere q almacene el vector"<<endl;

cin>>n;

vector <int> numeros(n);

cout<<"ingrese "<< n <<" numeros en el vector "<<endl;

for(i=0;i<numeros.size();i++){

cin>>numeros[i];

}

cout<<"los num ingresados son"<<endl;

for(i=0;i<numeros.size();i++){

cout<<numeros[i]<<" ";

}

cout<<endl;

nMAYOR=numeros[0];

for(i=1;i<numeros.size();i++){

if (nMAYOR<numeros[i])

{

 nMAYOR = numeros[i];

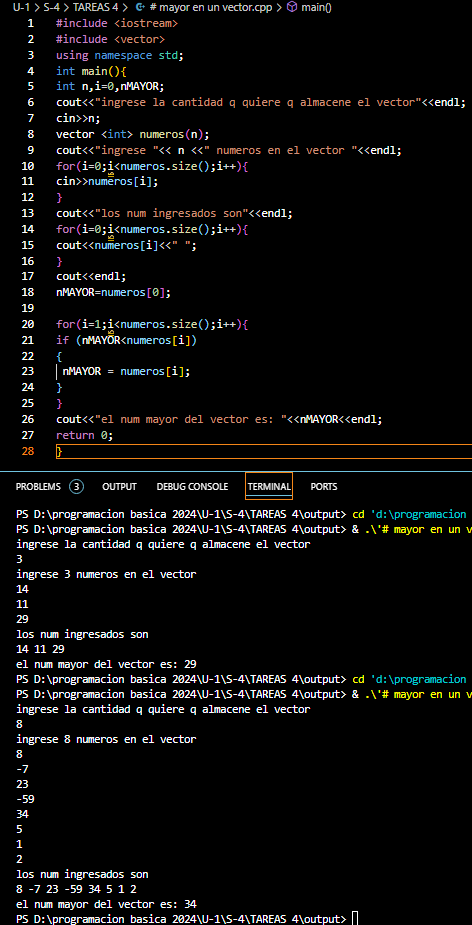
}

}

cout<<"el num mayor del vector es: "<<nMAYOR<<endl;

return 0;

}



**EJER 9 (números primos dentros de un rango ingresado por el usuario)**

Números Primos en un Rango: Escribe un programa que solicite al usuario dos números enteros (inicio y fin) y genere un vector con los números primos en ese rango. Utiliza continue para omitir los números no primos.

#include <iostream>

using namespace std;

bool esPrimo(int num) {

    if (num <= 1) {

        return false;

    }

    for (int i = 2; i \* i <= num; i++) {

        if (num % i == 0) {

            return false;

        }

    }

    return true;

}

int main() {

    int inicio, fin, rango, contador = 0;

    cout << "Introduce el num de inicio: ";

    cin >> inicio;

    cout << "Introduce el num de fin: ";

    cin >> fin;

    rango = fin - inicio + 1;

    int primos[rango];

    for (int num = inicio; num <= fin; num++) {

        if (!esPrimo(num)) {

            continue;

        }

        primos[contador] = num;

        contador++;

    }

    cout << "numeros primos en el rango [" << inicio << ", " << fin << "]: ";

    for (int i = 0; i < contador; i++) {

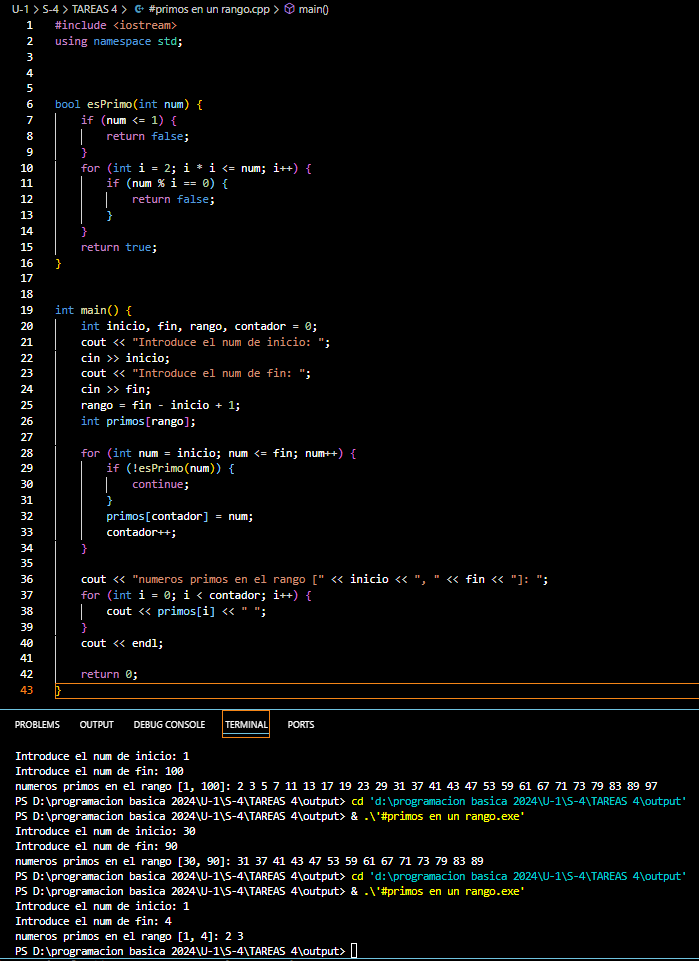
        cout << primos[i] << " ";

    }

    cout << endl;

    return 0;

}



**EJER 10 ( Contador de vocales )**

Contador de vocales: Desarrolla un programa que cuente la cantidad de a, e, i, o, u en una oración ingresada por el usuario.

CON VECTOR

#include <iostream>

#include <vector>

using namespace std;

bool vocales(char x){

vector <char> vocal = {'a','e','i','o','u','A','E','I','O','U'};

for(char v : vocal){

if(x==v){

    return true;

}

}

return false;

}

int main(){

string texto;

int contador=0;

cout<<"ingrese texto "<<endl;

getline(cin,texto);

for(char c: texto){

if (vocales(c)){

contador ++;

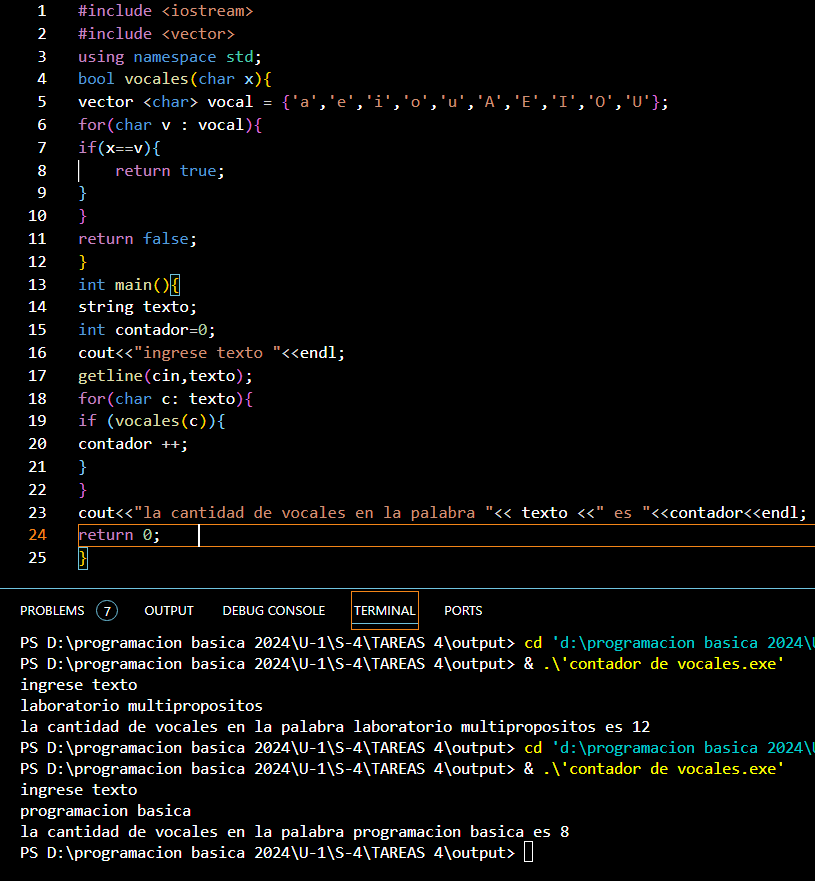
}

}

cout<<"la cantidad de vocales en la palabra "<< texto <<" es "<<contador<<endl;

return 0;

}



CON ARRAY

#include <iostream>

using namespace std;

bool vocales(char x){

char vocal [] = {'a','e','i','o','u','A','E','I','O','U'};

for(char v : vocal){

if(x==v){

    return true;

}

}

return false;

}

int main(){

string texto;

int contador=0;

cout<<"ingrese texto "<<endl;

getline(cin,texto);

for(char c: texto){

if (vocales(c)){

contador ++;

}

}

cout<<"la cantidad de vocales en la palabra "<< texto <<" es "<<contador<<endl;

return 0;

}

