



ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN DE TECNÓLOGOS



PROGRAMACION (TDSD214)

ASIGNATURA:

PROGRAMACION

PROFESOR:

Ing. Yadira Franco MSc.

PERÍODO ACADÉMICO:

2025B

TALLER

TÍTULO

PRÁCTICAS BASICAS

ESTUDIANTE:

Emely Maite Bone Zambrano

Introducción

En este taller tuve como objetivo reforzar y aplicar los conocimientos básicos de programación en C++, mediante el desarrollo de distintos ejercicios prácticos. A través de estas actividades busqué mejorar mi lógica de programación, el uso correcto de estructuras de control, arreglos, vectores, funciones y manejo de archivos, fortaleciendo así mis bases para resolver problemas computacionales.

Durante el desarrollo del taller realicé cada uno de los ejercicios propuestos, iniciando con estructuras condicionales y ciclos, para luego trabajar con arreglos, vectores, funciones y estructuras. Implementé programas que permiten ingresar datos, realizar cálculos, validar condiciones y mostrar resultados en pantalla. Además, en los ejercicios más avanzados utilicé archivos de texto para almacenar y recuperar información.

En esta sección se incluyen capturas de pantalla del código fuente y de la ejecución de los programas, las cuales evidencian el correcto funcionamiento de cada ejercicio desarrollado.

REFORZAR CONFUSIÓN ENTRE Y (&&) y O (||)**Ejercicio 1**

Elabore un programa en C++ que solicite un número entero e indique si el número pertenece al rango de 10 a 50, mostrando un mensaje cuando el valor sea válido y otro cuando no lo sea.

```
REFUERZO.cpp X
1  #include<iostream>
2
3  using namespace std;
4
5  int main(){
6
7      int num;
8
9      cout<<"Ingrese un numero del 10 al 50: ";
10     cin>>num;
11
12     if(num >= 10 && num <= 50){
13
14         cout<<"Valor valido.";
15     }else{
16         cout<<"Valor invalido, ingrese un valor del ranfo.";
17     }
18
19     return 0;
20 }
```

```
C:\Users\USER\Desktop\REFU X + v
Ingrese un numero del 10 al 50: 9
Valor invalido, ingrese un valor del ranfo.
-----
Process exited after 5.972 seconds with return value 0
Presione una tecla para continuar . . .
```

```
C:\Users\USER\Desktop\REFU X + v
Ingrese un numero del 10 al 50: 30
Valor valido.
-----
Process exited after 2.94 seconds with return value 0
Presione una tecla para continuar . . .
```

Ejercicio 2

Desarrolle un programa que solicite un número entero y determine si el número NO se encuentra en el rango de 10 a 50, mostrando un mensaje de advertencia cuando esté fuera del rango permitido.

```
REFUERZO.cpp X
20
21  #include<iostream>
22
23  using namespace std;
24
25  int main(){
26
27      int numero;
28
29      cout<<"Ingrese un numero de un rango de 10 a 50: ";
30      cin>>numero;
31
32      if(numero < 10 || numero > 50){
33          cout<<"ADVERTENCIA, NUMERO FUERA DEL RANGO.";
34      }else{
35          cout<<"Numero dentro del rango.";
36      }
37
38      return 0;
39  }
```

```
C:\Users\USER\Desktop\REFU x + v
Ingrese un numero de un rango de 10 a 50: 9
ADVERTENCIA, NUMERO FUERA DEL RANGO.
-----
Process exited after 29.7 seconds with return value 0
Presione una tecla para continuar . . .
```

□ B. EJERCICIOS SOLO CON FOR

Ejercicio 3

Desarrolle un programa que muestre en pantalla los números del 1 al 10, pero indicando al final cuántos números fueron mostrados, utilizando únicamente la estructura for.

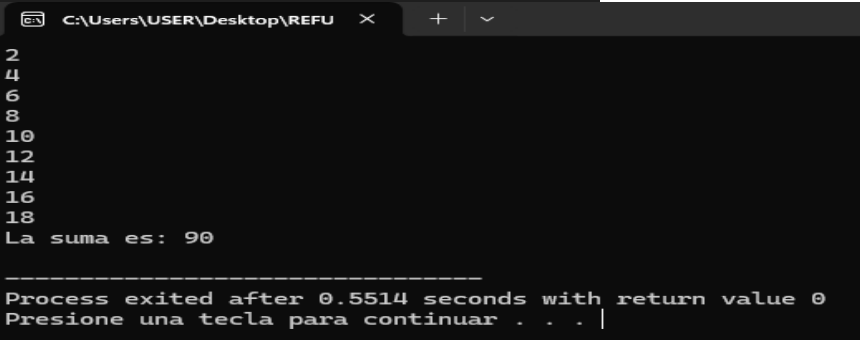
```
REFUERZO.cpp x EJERCICIO FOR 1.cpp x
1  #include<iostream>
2
3  using namespace std;
4
5  int main(){
6
7      int contador = 0;
8
9      for(int i=1;i<=10;i++){
10         cout<<i<<endl;
11         contador ++;
12     }
13
14     cout<<"Total de numeros: "<<contador<<endl;
15     return 0;
16 }
17 }
```

```
C:\Users\USER\Desktop\REFU x + v
1
2
3
4
5
6
7
8
9
10
Total de numeros: 10
-----
Process exited after 0.5389 seconds with return value 0
Presione una tecla para continuar . . .
```

Ejercicio 4

Realice un programa que muestre los números pares del 2 al 20 utilizando un ciclo for y muestre también la suma de dichos números.

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6
7      int suma = 0;
8
9      for(int i=2; i<20;i +=2){
10         cout<<i<<endl;
11
12         suma += i;
13     }
14
15     cout<<"La suma es: "<<suma<<endl;
16
17     return 0;
18 }
19
20
```



```
2
4
6
8
10
12
14
16
18
La suma es: 90

-----
Process exited after 0.5514 seconds with return value 0
Presione una tecla para continuar . . . |
```

Ejercicio 5

Elabore un programa que solicite un número entero positivo y muestre su tabla de multiplicar del 1 al 10, además de calcular la suma de los resultados obtenidos.

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6
7      int num;
8      int suma=0;
9
10     cout<<"Ingrese un numero positivo: ";
11     cin>>num;
12
13     for(int i=0; i<=10; i++){
14         cout<<i<<" x "<<num<<" = "<<i*num<<endl;
15
16         suma += i*num;
17     }
18
19     cout<<"La suma de los resultados es: "<<suma<<endl;
20
21 }
```

```
C:\Users\USER\Desktop\REFU X + v
Ingrese un numero positivo: 6
0 x 6 = 0
1 x 6 = 6
2 x 6 = 12
3 x 6 = 18
4 x 6 = 24
5 x 6 = 30
6 x 6 = 36
7 x 6 = 42
8 x 6 = 48
9 x 6 = 54
10 x 6 = 60
La suma de los resultados es: 330

-----
Process exited after 3.435 seconds with return value 0
Presione una tecla para continuar . . .
```

□ EJERCICIOS FOR + IF

Ejercicio 6

Desarrolle un programa que muestre los números del 1 al 20 e indique para cada número si es par o impar, utilizando estructuras for e if.

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6
7      for(int i=1;i<=20;i++){
8          if(i % 2 == 0){
9              cout<<i<<" : Numero Par"<<endl;
10             }else{
11                 cout<<i<<" : Numero Impar."<<endl;
12             }
13         }
14
15
16         return 0;
17     }
```

```
C:\Users\USER\Desktop\REFU X + v
1: Numero Impar.
2: Numero Par
3: Numero Impar.
4: Numero Par
5: Numero Impar.
6: Numero Par
7: Numero Impar.
8: Numero Par
9: Numero Impar.
10: Numero Par
11: Numero Impar.
12: Numero Par
13: Numero Impar.
14: Numero Par
15: Numero Impar.
16: Numero Par
17: Numero Impar.
18: Numero Par
19: Numero Impar.
20: Numero Par

-----
Process exited after 0.4653 seconds with return value 0
Presione una tecla para continuar . . .
```

Ejercicio 7

Elabore un programa que recorra los números del 1 al 50 y cuente cuántos son múltiplos de 3 y cuántos no lo

son, mostrando ambos resultados.

```

1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6
7      int multiplo = 0, nmultiplo = 0;
8
9      for(int i=1;i<=50;i++){
10         if (i%3 == 0){
11             multiplo ++;
12         }else{
13             nmultiplo ++;
14         }
15     }
16
17     cout<<"Multiplos de 3: "<<multiplo<<endl;
18     cout<<"No multiplos: "<<nmultiplo<<endl;
19
20     return 0;
21 }

```

```

C:\Users\USER\Desktop\REFU X + v
Multiplos de 3: 16
No multiplos: 34

-----
Process exited after 0.5552 seconds with return value 0
Presione una tecla para continuar . . .

```

Ejercicio 8

Realice un programa que permita ingresar 10 números y determine cuántos son positivos, cuántos son negativos y cuántos son iguales a cero.

```

5  int main(){
6
7      int positivos = 0;
8      int negativos = 0;
9      int cero = 0;
10     int num;
11
12     cout<<"== Ingrese 10 numeros ==<<endl;
13     for(int i=1;i<=10;i++){
14         cout<<"Numero "<<i<<": ";
15         cin>>num;
16
17         if(num > 0){
18             positivos ++;
19         }else if(num < 0){
20             negativos ++;
21         }else{
22             cero ++;
23         }
24     }
25
26     cout<<"Numeros Positivos: "<<positivos<<endl;
27     cout<<"Numeros Negativos: "<<negativos<<endl;
28     cout<<"Igual a ceros: "<<cero<<endl;
29
30     return 0;
31 }

```

```

C:\Users\USER\Desktop\REFU X + v
== Ingrese 10 numeros ==
Numero 1: 6
Numero 2: 9
Numero 3: -4
Numero 4: -3
Numero 5: 9
Numero 6: 1
Numero 7: 0
Numero 8: 0
Numero 9: 0
Numero 10: 4
Numeros Positivos: 5
Numeros Negativos: 2
Igual a ceros: 3

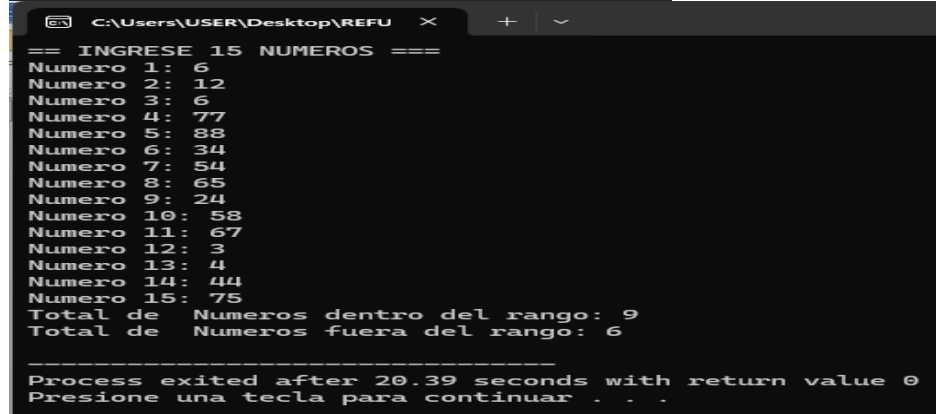
-----
Process exited after 12.02 seconds with return value 0
Presione una tecla para continuar . . .

```

Ejercicio 9

Desarrolle un programa que permita ingresar 15 números y determine cuántos valores se encuentran dentro del rango de 20 a 80 y cuántos están fuera de ese rango.

```
2
3 using namespace std;
4
5 int main(){
6
7     int num;
8     int rango = 0;
9     int fuera = 0;
10
11
12     cout<<"== INGRESE 15 NUMEROS ==<<endl;
13     for(int i=1;i<=15;i++){
14         cout<<"Numero "<<i<<": ";
15         cin>>num;
16
17         if(num >=20 && num <= 80){
18             rango ++;
19         }else{
20             fuera ++;
21         }
22     }
23
24     cout<<"Total de  Numeros dentro del rango: "<<rango<<endl;
25     cout<<"Total de  Numeros fuera del rango: "<<fuera<<endl;
26
27     return 0;
28 }
```



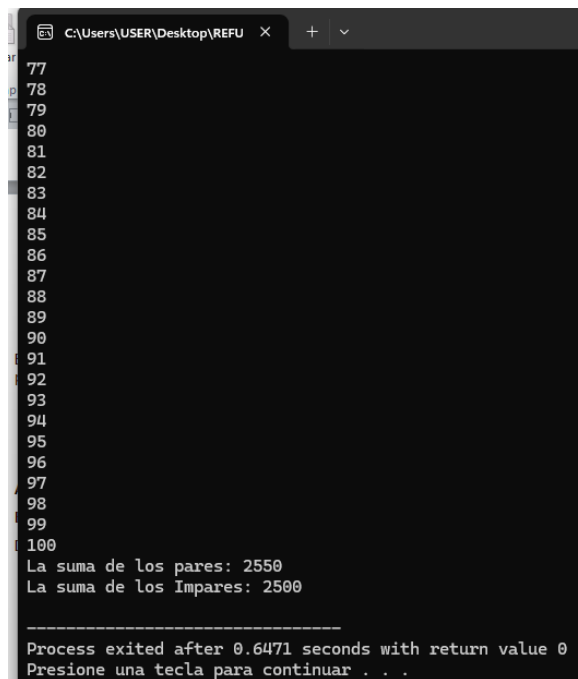
```
== INGRESE 15 NUMEROS ==
Numero 1: 6
Numero 2: 12
Numero 3: 6
Numero 4: 77
Numero 5: 88
Numero 6: 34
Numero 7: 54
Numero 8: 65
Numero 9: 24
Numero 10: 58
Numero 11: 67
Numero 12: 3
Numero 13: 4
Numero 14: 44
Numero 15: 75
Total de  Numeros dentro del rango: 9
Total de  Numeros fuera del rango: 6

Process exited after 20.39 seconds with return value 0
Presione una tecla para continuar . . .
```

Ejercicio 10

Elabore un programa que muestre los números del 1 al 100 y calcule la suma de los números pares y la suma de los números impares, indicando al final cuál suma es mayor.

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main(){
6
7     int suma = 0;
8     int suma1 = 0;
9
10
11     for(int i=1;i<=100;i++){
12         cout<<i<<endl;
13
14         if(i % 2 == 0){
15             suma +=i;
16         }else{
17             suma1 +=i;
18         }
19     }
20
21     cout<<"La suma de los pares: "<<suma<<endl;
22     cout<<"La suma de los Impares: "<<suma1<<endl;
23
24     return 0;
25 }
26
```



```
C:\Users\USER\Desktop\REFU X + v
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
La suma de los pares: 2550
La suma de los Impares: 2500

-----
Process exited after 0.6471 seconds with return value 0
Presione una tecla para continuar . . .
```

ARREGLOS (ARRAYS) – CON CÁLCULOS

Ejercicio 11

Desarrolle un programa que permita ingresar **10 números en un arreglo** y calcule:

- La suma total de los valores
- El promedio
- El porcentaje de números positivos
- El porcentaje de números negativos

```
1  #include<iostream>
2
3  using namespace std;
4
5  int main(){
6      int numeros [10];
7      int suma = 0;
8      int posit = 0;
9      int negat = 0;
10
11      cout<<"== INGRESO DE 10 NUMEROS =="<<endl;
12      for(int i=0;i<10;i++){
13          cout<<"Numero "<<i<<": ";
14          cin>>numeros[i];
15
16          suma+=numeros[i];
17
18          if(numeros [i] > 0){
19              posit ++;
20          }else{
21              negat ++;
22          }
23      }
24      int porcentajeP = posit * 10;
25      int porcentajeN = negat * 10;
26      int promedio = suma / 10;
```



```
27
28     cout<<"La Suma: "<<suma<<endl;
29     cout<<"Promedio: "<<promedio<<endl;
30     cout<<"Porcentaje Negativos: "<<porcentajeN<<endl;
31     cout<<"Porcentaje Positivos: "<<porcentajeP<<endl;
32
33     return 0;
34 }
35
```

```
C:\Users\USER\Desktop\REFU x + v
== INGRESO DE 10 NUMEROS ==
Numero 0: 2
Numero 1: 7
Numero 2: 8
Numero 3: 33
Numero 4: 5
Numero 5: 1
Numero 6: 3
Numero 7: 9
Numero 8: 4
Numero 9: 6
La Suma: 78
Promedio: 7
Porcentaje Negativos: 0
Porcentaje Positivos: 100

-----
Process exited after 16.78 seconds with return value 0
Presione una tecla para continuar . . .
```

Ejercicio 12

Elabore un programa que permita ingresar **8 números en un arreglo** y determine:

- El número mayor
- El número menor
- La diferencia entre el mayor y el menor

```
C:\Users\USER\Desktop\REFU x + v
== INGRESO DE 8 NUMEROS ==
Numero 1: 7
Numero 2: 8
Numero 3: 3
Numero 4: 11
Numero 5: 7
Numero 6: 9
Numero 7: 8
Numero 8: 3
Numero mayor: 11
Numero menor: 3
Diferencia: 8

-----
Process exited after 9.283 seconds with return value 0
Presione una tecla para continuar . . .
```

```
1  #include <iostream>
2  using namespace std;
3
4  int main(){
5
6      int numeros[8];
7      int mayor;
8      int menor;
9
10     cout << "== INGRESO DE 8 NUMEROS ==" << endl;
11     for(int i = 0; i < 8; i++){
12         cout << "Numero " << i + 1 << ": ";
13         cin >> numeros[i];
14     }
15
16     mayor = numeros[0];
17     menor = numeros[0];
18
19     for(int i = 1; i < 8; i++){
20         if(numeros[i] > mayor){
21             mayor = numeros[i];
22         }
23         if(numeros[i] < menor){
24             menor = numeros[i];
25         }
26     }
27
28     int diferencia = mayor - menor;
29
30     cout << "Numero mayor: " << mayor << endl;
31     cout << "Numero menor: " << menor << endl;
32     cout << "Diferencia: " << diferencia << endl;
33
34     return 0;
35 }
```

Ejercicio 13

Desarrolle un programa que permita ingresar **12 números en un arreglo** y calcule:

- La suma de los números pares
- La suma de los números impares
- Cuál suma es mayor (pares o impares)

```
1  #include<iostream>
2
3  using namespace std;
4
5  int main(){
6
7      int numeros[12];
8      int suma = 0;
9      int suma1 =0;
10
11     cout<<"== INGRESO DE 12 NUMEROS =="<<endl;
12     for(int i=0;i<12;i++){
13         cout<<"Numero "<<i<<": ";
14         cin>>numeros[i];
15         if(numeros[i] % 2 == 0){
16             suma+=numeros[i];
17         }else{
18             suma1+=numeros[i];
19         }
20     }
21     cout<<"Suma de Pares: "<<suma<<endl;
22     cout<<"Suma de Impares: "<<suma1<<endl;
```

```
24     if(suma > suma1){  
25         cout<<"La mayor suma es de los pares"<<endl;  
26     }else{  
27         cout<<"La mayor suma es de los impares"<<endl;  
28     }  
29     }  
30  
31     return 0;  
32 }
```

```
C:\Users\USER\Desktop\REFU X + v  
== INGRESO DE 12 NUMEROS ==  
Numero 0: 5  
Numero 1: 8  
Numero 2: 1  
Numero 3: 35  
Numero 4: 7  
Numero 5: 89  
Numero 6: 2  
Numero 7: 1  
Numero 8: 9  
Numero 9: 1  
Numero 10: 6  
Numero 11: 4  
Suma de Pares: 20  
Suma de Impares: 148  
La mayor suma es de los impares  
  
-----  
Process exited after 24.6 seconds with return value 0  
Presione una tecla para continuar . . .
```

Ejercicio 14

Realice un programa que permita ingresar **10 calificaciones (0–10)** en un arreglo y determine:

- El promedio general
- Cuántos estudiantes aprobaron (≥ 7)
- Cuántos reprobaron (< 7)

```
1  #include<iostream>
2
3  using namespace std;
4
5  int main(){
6
7      int calificaciones[10];
8      int aprobados = 0;
9      int reprobados = 0;
10     int suma = 0;
11
12     for(int i=1;i<11;i++){
13         cout<<"Nota "<<i<<" : ";
14         cin>>calificaciones[i];
15
16         if (calificaciones[i]>=7){
17             aprobados ++;
18         }else if(calificaciones[i]<=7){
19             reprobados ++;
20         }
21
22         suma+=i;
23     }
24
25     int promedio = suma/10;
26     cout<<"Promedio Gnereal: "<<promedio<<endl;
27     cout<<"Estudiantes Aprobados: "<<aprobados<<endl;
28     cout<<"Estudiantes Reprobados: "<<reprobados<<endl;
29
30     return 0;
31 }
```

```
C:\Users\USER\Desktop\REFU x + v
Nota 1: 5
Nota 2: 9
Nota 3: 10
Nota 4: 8
Nota 5: 7
Nota 6: 4
Nota 7: 3
Nota 8: 2
Nota 9: 9
Nota 10: 10
Promedio Gnereal: 5
Estudiantes Aprobados: 6
Estudiantes Reprobados: 4

-----
Process exited after 28.18 seconds with return value 0
Presione una tecla para continuar . . .
```

Ejercicio 15

Elabore un programa que permita ingresar **15 números en un arreglo** y calcule:

- El promedio de los números positivos
- El promedio de los números negativos

```

C:\Users\USER\Desktop\REFU x + v
== 15 NUMEROS ==
Numero 1: -9
Numero 2: 8
Numero 3: 5
Numero 4: -13
Numero 5: 6
Numero 6: 7
Numero 7: -5
Numero 8: -1
Numero 9: -3
Numero 10: 8
Numero 11: 11
Numero 12: 4
Numero 13: 2
Numero 14: -9
Numero 15: -6
Promedio positivos: 6
Promedio negativos: -6

-----
Process exited after 37.89 seconds with return value 0
Presione una tecla para continuar . . .

```

```

1  #include<iostream>
2  using namespace std;
3
4  int main(){
5
6      int arreglo[15];
7      int positivos = 0;
8      int negativos = 0;
9      int suma = 0;
10     int suma1 = 0;
11
12     cout<<"== 15 NUMEROS == "<<endl;
13     for(int i = 0; i < 15; i++){
14         cout<<"Numero "<<i+1<<": ";
15         cin>>arreglo[i];
16
17         if(arreglo[i] > 0){
18             suma += arreglo[i];
19             positivos++;
20         }else if(arreglo[i] < 0){
21             suma1 += arreglo[i];
22             negativos++;
23         }
24     }
25
26     if(positivos > 0){
27         int promedio = suma / positivos;
28         cout<<"Promedio positivos: "<<promedio<<endl;
29     }else{
30         cout<<"No hay numeros positivos"<<endl;
31     }
32
33     if(negativos > 0){
34         int promedio1 = suma1 / negativos;
35         cout<<"Promedio negativos: "<<promedio1<<endl;
36     }else{
37         cout<<"No hay numeros negativos"<<endl;
38     }
39
40     return 0;
41 }

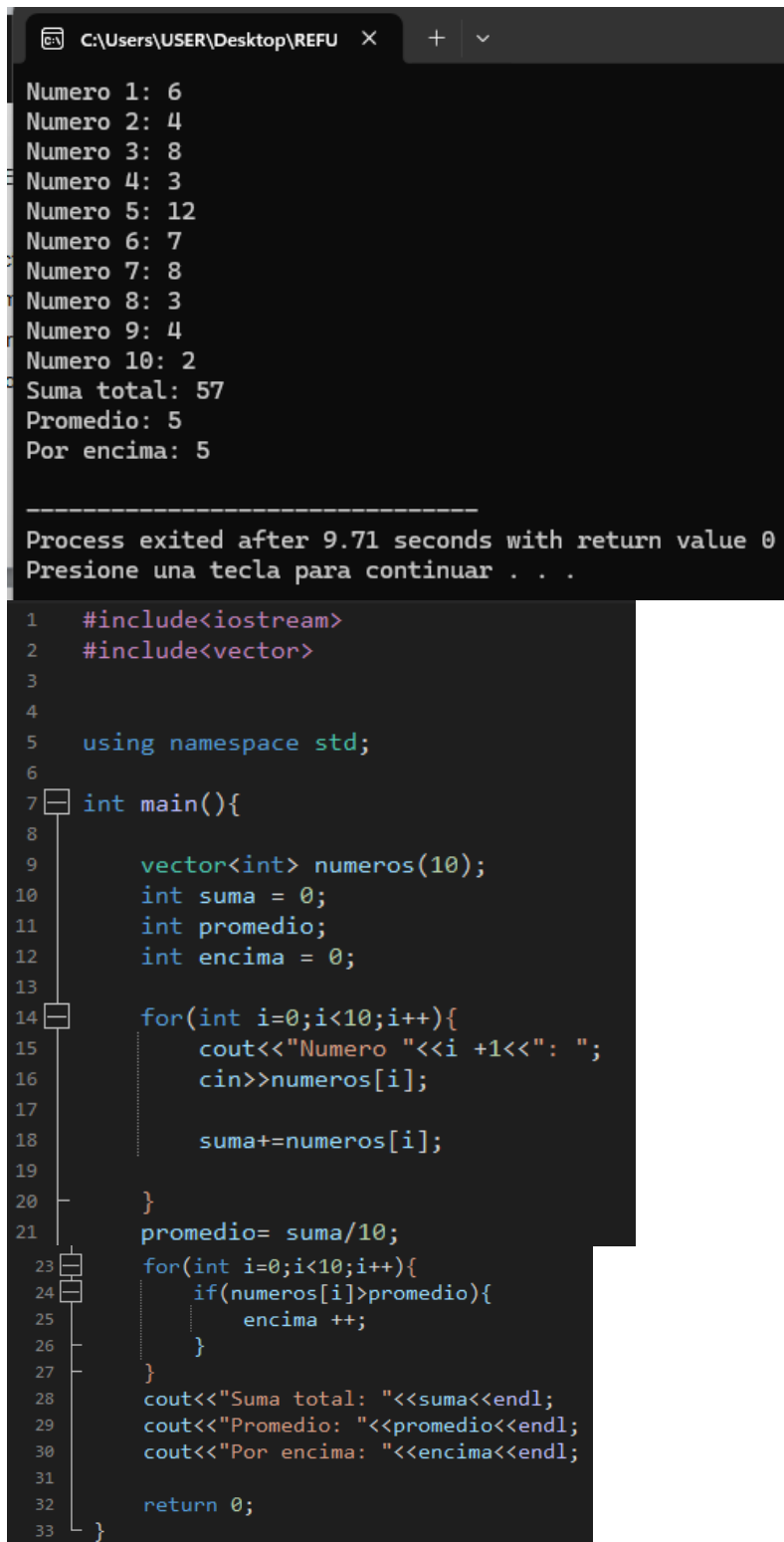
```

□ VECTORES – CON CÁLCULOS

Ejercicio 16

Utilizando vectores, desarrolle un programa que permita ingresar **10 números** y determine:

- La suma total
- El promedio
- Cuántos números están por encima del promedio



```

C:\Users\USER\Desktop\REFU x + v
Numero 1: 6
Numero 2: 4
Numero 3: 8
Numero 4: 3
Numero 5: 12
Numero 6: 7
Numero 7: 8
Numero 8: 3
Numero 9: 4
Numero 10: 2
Suma total: 57
Promedio: 5
Por encima: 5

-----
Process exited after 9.71 seconds with return value 0
Presione una tecla para continuar . . .

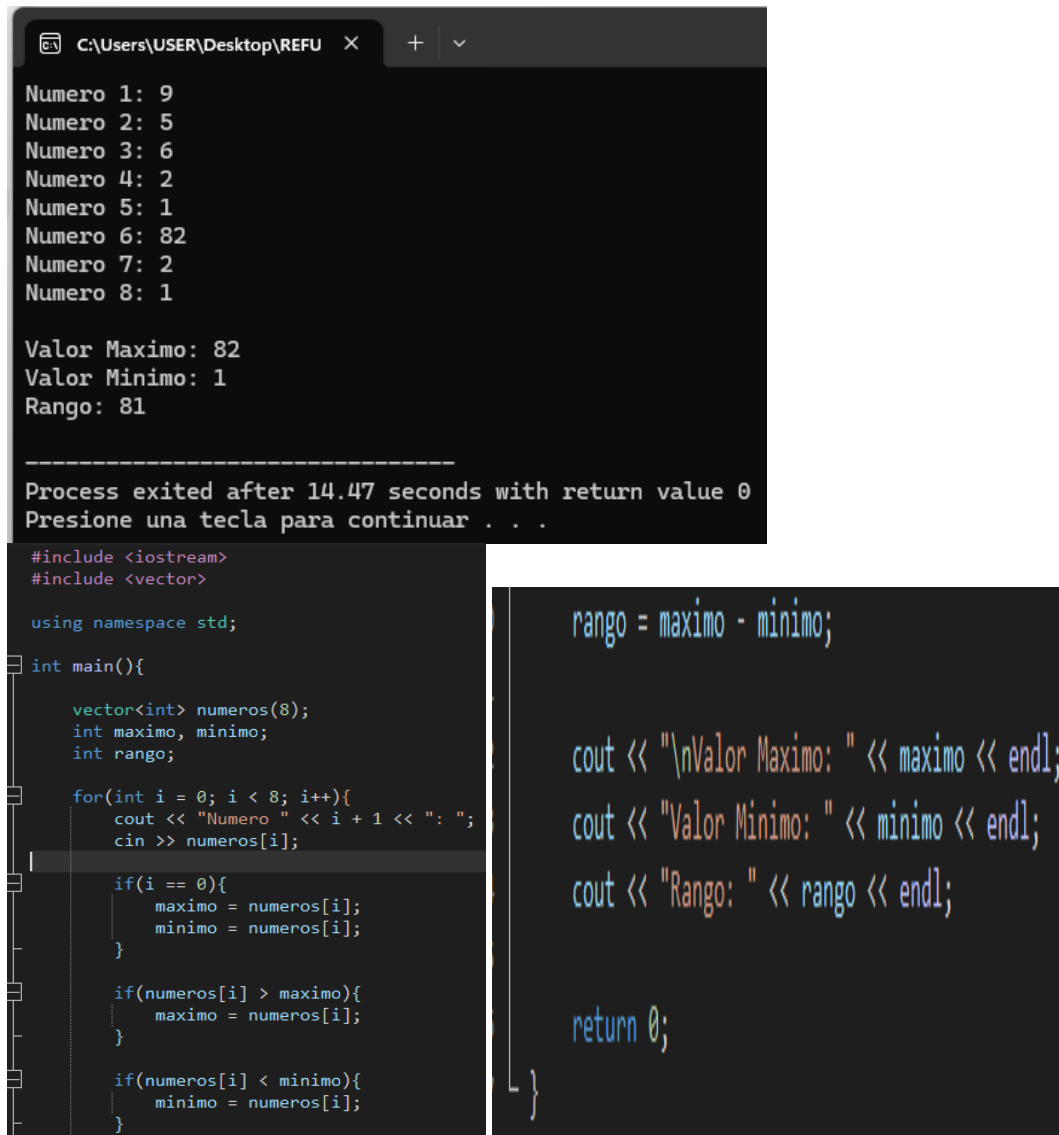
1  #include<iostream>
2  #include<vector>
3
4
5  using namespace std;
6
7  int main(){
8
9      vector<int> numeros(10);
10     int suma = 0;
11     int promedio;
12     int encima = 0;
13
14     for(int i=0;i<10;i++){
15         cout<<"Numero "<<i +1<<" : ";
16         cin>>numeros[i];
17
18         suma+=numeros[i];
19     }
20
21     promedio= suma/10;
22
23     for(int i=0;i<10;i++){
24         if(numeros[i]>promedio){
25             encima ++;
26         }
27     }
28     cout<<"Suma total: "<<suma<<endl;
29     cout<<"Promedio: "<<promedio<<endl;
30     cout<<"Por encima: "<<encima<<endl;
31
32     return 0;
33 }
```

Ejercicio 17

Elabore un programa con vectores que permita ingresar **8 valores** y calcule:

- El valor máximo
- El valor mínimo

- El rango (máximo – mínimo)



```
Numero 1: 9
Numero 2: 5
Numero 3: 6
Numero 4: 2
Numero 5: 1
Numero 6: 82
Numero 7: 2
Numero 8: 1

Valor Maximo: 82
Valor Minimo: 1
Rango: 81

-----
Process exited after 14.47 seconds with return value 0
Presione una tecla para continuar . . .

#include <iostream>
#include <vector>

using namespace std;

int main(){

    vector<int> numeros(8);
    int maximo, minimo;
    int rango;

    for(int i = 0; i < 8; i++){
        cout << "Numero " << i + 1 << ": ";
        cin >> numeros[i];

        if(i == 0){
            maximo = numeros[i];
            minimo = numeros[i];
        }

        if(numeros[i] > maximo){
            maximo = numeros[i];
        }

        if(numeros[i] < minimo){
            minimo = numeros[i];
        }
    }

    rango = maximo - minimo;

    cout << "\nValor Maximo: " << maximo << endl;
    cout << "Valor Minimo: " << minimo << endl;
    cout << "Rango: " << rango << endl;

    return 0;
}
```

Ejercicio 18

Desarrolle un programa que permita ingresar **12 valores en un vector** y determine:

- La suma de los valores positivos
- La suma de los valores negativos
- El valor absoluto total acumulado

```
1  #include <iostream>
2  #include <vector>
3
4  using namespace std;
5
6  int main(){
7
8      vector<int> numeros(12);
9      int sumaPositivos = 0;
10     int sumaNegativos = 0;
11     int absolutoTotal = 0;
12
13     for(int i = 0; i < 12; i++){
14         cout << "Valor " << i + 1 << ": ";
15         cin >> numeros[i];
16
17         if(numeros[i] > 0){
18             sumaPositivos += numeros[i];
19             absolutoTotal += numeros[i];
20         }
21
22         if(numeros[i] < 0){
23             sumaNegativos += numeros[i];
24             absolutoTotal += -numeros[i];
25         }
26     }
27
28     cout << "\nSuma de positivos: " << sumaPositivos << endl;
29     cout << "Suma de negativos: " << sumaNegativos << endl;
30     cout << "Valor absoluto total acumulado: " << absolutoTotal << endl;
31
32     return 0;
33 }
34 }
```

C:\Users\USER\Desktop\REFU X + v

Valor 1: 7
Valor 2: 3
Valor 3: -4
Valor 4: -3
Valor 5:
9
Valor 6: -7
Valor 7: 4
Valor 8: 1
Valor 9: 1
Valor 10: 3
Valor 11: 6
Valor 12: 11

Suma de positivos: 45
Suma de negativos: -14
Valor absoluto total acumulado: 59

Process exited after 16.95 seconds with return value 0
Presione una tecla para continuar . . .

Ejercicio 19

Utilizando vectores, elabore un programa que permita ingresar **10 edades** y calcule:

- El promedio de edades
- Cuántas edades son mayores al promedio
- Cuántas edades son menores o iguales al promedio


```
C:\Users\USER\Desktop\REFU X + v

Edad 1: 8
Edad 2: 9
Edad 3: 11
Edad 4: 21
Edad 5: 22
Edad 6: 3
Edad 7: 4
Edad 8: 7
Edad 9: 5
Edad 10: 3
Promedio de edades: 9
Mayores al promedio: 5
Menores al promedio: 5

-----
Process exited after 17.88 seconds with return value 0
Presione una tecla para continuar . . .

6 int main(){
7
8
9     vector<int> edades(10);
10
11     int promedio;
12     int suma = 0;
13     int mayores = 0;
14     int menores = 0;
15
16     for(int i=0;i<10;i++){
17         cout<<"Edad "<<i+1<<" ";
18         cin>>edades[i];
19         suma+=edades[i];
20
21         promedio=suma/10;
22
23         if(edades[i]>promedio){
24             mayores ++;
25         }
26         if(edades[i]<promedio){
27             menores ++;
28         }
29     }
30
31     cout<<"Promedio de edades: "<<promedio<<endl;
32     cout<<"Mayores al promedio: "<<mayores<<endl;
33     cout<<"Menores al promedio: "<<menores<<endl;
34
35
36     return 0;
37 }
```

Ejercicio 20

Desarrolle un programa con vectores que permita ingresar **10 números** y calcule:

- El cuadrado de cada número
- La suma de los cuadrados
- El promedio de los cuadrados



```

C:\Users\USER\Desktop\REFU x + v
Numero 1: 6
Numero 2: 7
Numero 3: 8
Numero 4: 3
Numero 5: 2
Numero 6: 4
Numero 7: 5
Numero 8: 7
Numero 9: 5
Numero 10: 2
Cuadrados de los numeros:
Numero 6 al cuadrado = 36
Numero 7 al cuadrado = 49
Numero 8 al cuadrado = 64
Numero 3 al cuadrado = 9
Numero 2 al cuadrado = 4
Numero 4 al cuadrado = 16
Numero 5 al cuadrado = 25
Numero 7 al cuadrado = 49
Numero 5 al cuadrado = 25
Numero 2 al cuadrado = 4
Suma de los cuadrados: 281
Promedio de los cuadrados: 28.1

-----
Process exited after 8.78 seconds with return value 0
Presione una tecla para continuar . . . |

1  #include <iostream>
2  #include <vector>
3
4  using namespace std;
5
6  int main() {
7
8      vector<int> numeros(10);
9      int sumaCuadrados = 0;
10     float promedioCuadrados;
11
12     for(int i = 0; i < 10; i++){
13         cout << "Numero " << i + 1 << ": ";
14         cin >> numeros[i];
15         sumaCuadrados += numeros[i] * numeros[i];
16     }
17     promedioCuadrados = sumaCuadrados / 10.0;
18
19     cout << "Cuadrados de los numeros:" << endl;
20     for(int i = 0; i < 10; i++){
21         cout << "Numero " << numeros[i] << " al cuadrado = " << numeros[i] * numeros[i] << endl;
22     }
23     cout << "Suma de los cuadrados: " << sumaCuadrados << endl;
24     cout << "Promedio de los cuadrados: " << promedioCuadrados << endl;
25
26     return 0;
27 }

```

□ ARREGLOS / VECTORES + CONDICIONES

Ejercicio 21

Elabore un programa que permita ingresar **10 números en un arreglo** y determine:

- Cuántos están en el rango de **20 a 80**
- Cuántos están fuera de ese rango
- El porcentaje de cada grupo

```
C:\Users\USER\Desktop\REFU x + v

Ingrese 10 numeros:
Numero 1: 22
Numero 2: 88
Numero 3: 56
Numero 4: 3
Numero 5: 1
Numero 6: 7
Numero 7: 54
Numero 8: 3
Numero 9: 2
Numero 10: 9
Resultados:
Cantidad en el rango (20-80): 3
Cantidad fuera del rango: 7
Porcentaje en el rango: 30%
Porcentaje fuera del rango: 70%

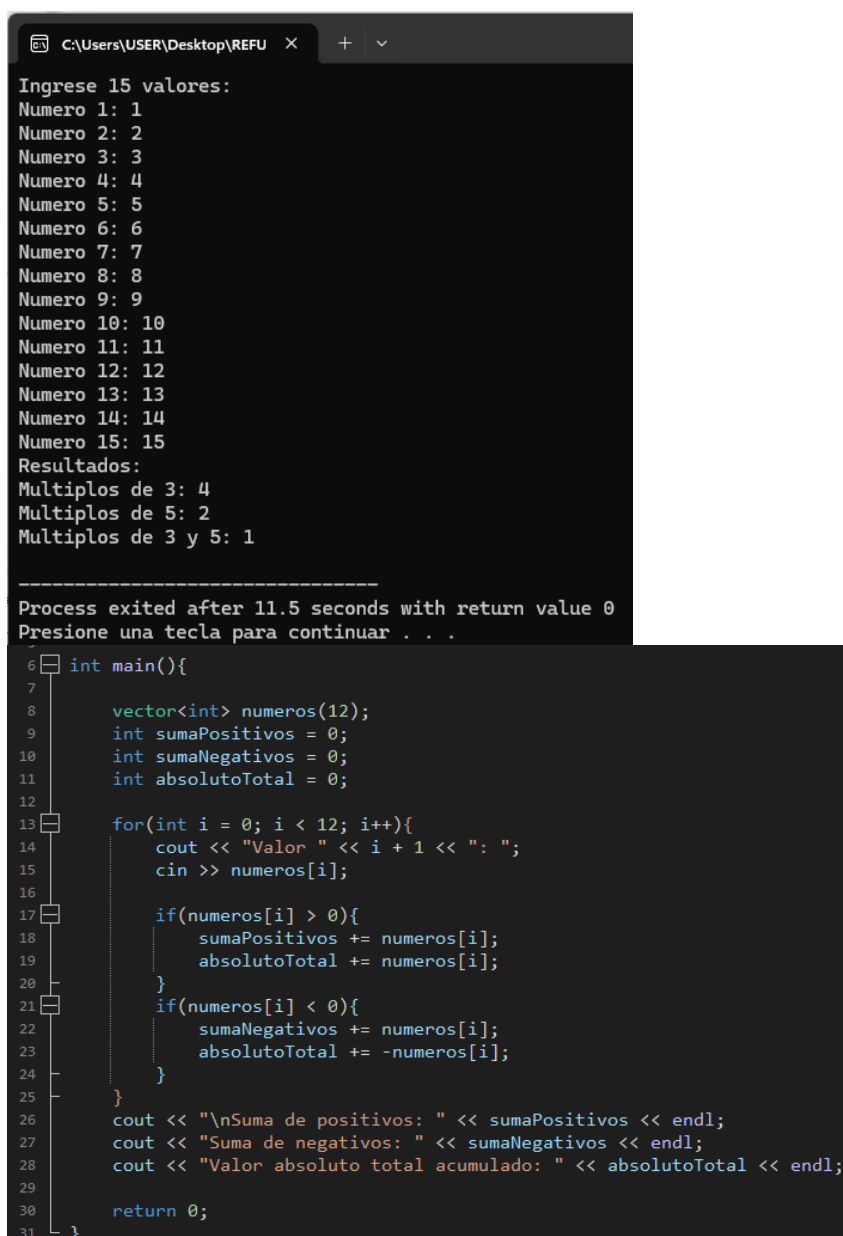
-----
Process exited after 10.55 seconds with return value 0
Presione una tecla para continuar . . .

1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int numeros[10];
6      int enRango = 0;
7      int fueraRango = 0;
8
9      cout << "Ingrese 10 numeros:" << endl;
10
11     for(int i = 0; i < 10; i++) {
12         cout << "Numero " << i+1 << ": ";
13         cin >> numeros[i];
14
15         if(numeros[i] >= 20 && numeros[i] <= 80) {
16             enRango++;
17         } else {
18             fueraRango++;
19         }
20     }
21
22     double porcentajeRango = (enRango * 100.0) / 10;
23     double porcentajeFuera = (fueraRango * 100.0) / 10;
24
25     cout << "Resultados:" << endl;
26     cout << "Cantidad en el rango (20-80): " << enRango << endl;
27     cout << "Cantidad fuera del rango: " << fueraRango << endl;
28     cout << "Porcentaje en el rango: " << porcentajeRango << "%" << endl;
29     cout << "Porcentaje fuera del rango: " << porcentajeFuera << "%" << endl;
30
31     return 0;
32 }
```

Ejercicio 22

Desarrolle un programa que permita ingresar **15 valores en un vector** y calcule:

- Cuántos valores son múltiplos de 3
- Cuántos valores son múltiplos de 5
- Cuántos valores son múltiplos de ambos



```
Process exited after 11.5 seconds with return value 0
Presione una tecla para continuar . . .

6 int main(){
7
8     vector<int> numeros(12);
9     int sumaPositivos = 0;
10    int sumaNegativos = 0;
11    int absolutoTotal = 0;
12
13    for(int i = 0; i < 12; i++){
14        cout << "Valor " << i + 1 << ": ";
15        cin >> numeros[i];
16
17        if(numeros[i] > 0){
18            sumaPositivos += numeros[i];
19            absolutoTotal += numeros[i];
20        }
21        if(numeros[i] < 0){
22            sumaNegativos += numeros[i];
23            absolutoTotal += -numeros[i];
24        }
25    }
26    cout << "\nSuma de positivos: " << sumaPositivos << endl;
27    cout << "Suma de negativos: " << sumaNegativos << endl;
28    cout << "Valor absoluto total acumulado: " << absolutoTotal << endl;
29
30    return 0;
31 }
```

Ejercicio 23

Elabore un programa que permita ingresar **10 precios** y calcule:

- El subtotal
- El IVA (12 %)
- El total a pagar

```
C:\Users\USER\Desktop\REFU X + v
Precio 1: 6.60
Precio 2: 5.30
Precio 3: 2
Precio 4: 6
Precio 5: 8
Precio 6: 9.99
Precio 7: 2
Precio 8: 1.50
Precio 9: 1
Precio 10: 3
Subtotal: 45.39
Iva (12%): 5.4468
Total: 50.8368

-----
Process exited after 13.98 seconds with return value 0
Presione una tecla para continuar . . .
```

```
1  #include<iostream>
2  #include<vector>
3
4  using namespace std;
5
6  int main(){
7
8      vector<double> precios(10);
9      double iva = 0;
10     double subtotal = 0;
11     double total = 0;
12
13     for(int i=0;i<10;i++){
14         cout<<"Precio "<<i+1<<" ";
15         cin>>precios[i];
16
17         subtotal += precios[i];
18     }
19     iva = subtotal * 0.12;
20     total = subtotal + iva;
21
22     cout<<"Subtotal: "<<subtotal<<endl;
23     cout<<"Iva (12%): "<<iva<<endl;
24     cout<<"Total: "<<total<<endl;
25     return 0;
26 }
```

Ejercicio 24

Desarrolle un programa que permita ingresar **10 sueldos** y calcule:

- El sueldo promedio
- Cuántos sueldos están por encima del promedio
- Cuántos están por debajo del promedio

```

C:\Users\USER\Desktop\REFU x + v
Sueldo 1: 5.50
Sueldo 2: 9.99
Sueldo 3: 4
Sueldo 4: 5
Sueldo 5: 7
Sueldo 6: 8.9
Sueldo 7: 2
Sueldo 8: 1
Sueldo 9: 5
Sueldo 10: 7.50
Sueldo Promedio: 5.589
Suelos por encima: 4
Suelos por debajo: 6

-----
Process exited after 20.62 seconds with return value 0
Presione una tecla para continuar . . .

```

```

3  using namespace std;
4
5  int main(){
6
7      float arreglo[10];
8      float promedio;
9      float suma = 0;
10     float encima = 0;
11     float debajo = 0;
12
13     for(int i=0;i<10;i++){
14         cout<<"Sueldo "<<i+1<<" ";
15         cin>>arreglo[i];
16
17         suma += arreglo[i];
18     }
19
20     promedio = suma/10;
21
22     for(int i=0;i<10;i++){
23         if(arreglo[i]>promedio){
24             encima ++;
25         }
26         if(arreglo[i]<promedio){
27             debajo ++;
28         }
29     }
30
31     cout<<"Sueldo Promedio: "<<promedio<<endl;
32     cout<<"Suelos por encima: "<<encima<<endl;
33     cout<<"Suelos por debajo: "<<debajo<<endl;
34
35     return 0;
36 }

```

□ RETO (INTEGRADOR)

Ejercicio 25

Desarrolle un programa que permita ingresar **10 notas finales** y calcule:

- El promedio general
- La nota más alta y la más baja
- El porcentaje de aprobados y reprobados
- Muestre un resumen final de resultados

```

C:\Users\USER\Desktop\REFU X + v
Ingrese 10 notas finales:
Nota 1: 3.30
Nota 2: 9
Nota 3: 8.90
Nota 4: 9
Nota 5: 9
Nota 6: 9
Nota 7: 3
Nota 8: 1
Nota 9: 5
Nota 10: 7
==== RESUMEN DE RESULTADOS ====
Promedio general: 6.42
Nota más alta: 9
Nota más baja: 1
Cantidad de aprobados: 6 (60%)
Cantidad de reprobados: 4 (40%)

-----
Process exited after 20.89 seconds with return value 0
Presione una tecla para continuar . . .

1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      float notas[10];
6      float suma = 0;
7      float promedio;
8      float notaMax = 0;
9      float notaMin = 100;
10     int aprobados = 0;
11     int reprobados = 0;
12
13     cout << "Ingrese 10 notas finales:" << endl;
14
15     for(int i = 0; i < 10; i++) {
16         cout << "Nota " << i+1 << ": ";
17         cin >> notas[i];
18
19         suma += notas[i];
20
21         if(notas[i] > notaMax){
22             notaMax = notas[i];
23         }
24         if(notas[i] < notaMin){
25             notaMin = notas[i];
26         }
27     }
28
29     if(notas[i] >= 7)
30         aprobados++;
31     else
32         reprobados++;
33 }
34
35 promedio = suma/10;
36
37 float porcAprobados = (aprobados * 100.0) / 10;
38 float porcReprobados = (reprobados * 100.0) / 10;
39
40 cout << "==== RESUMEN DE RESULTADOS =====>

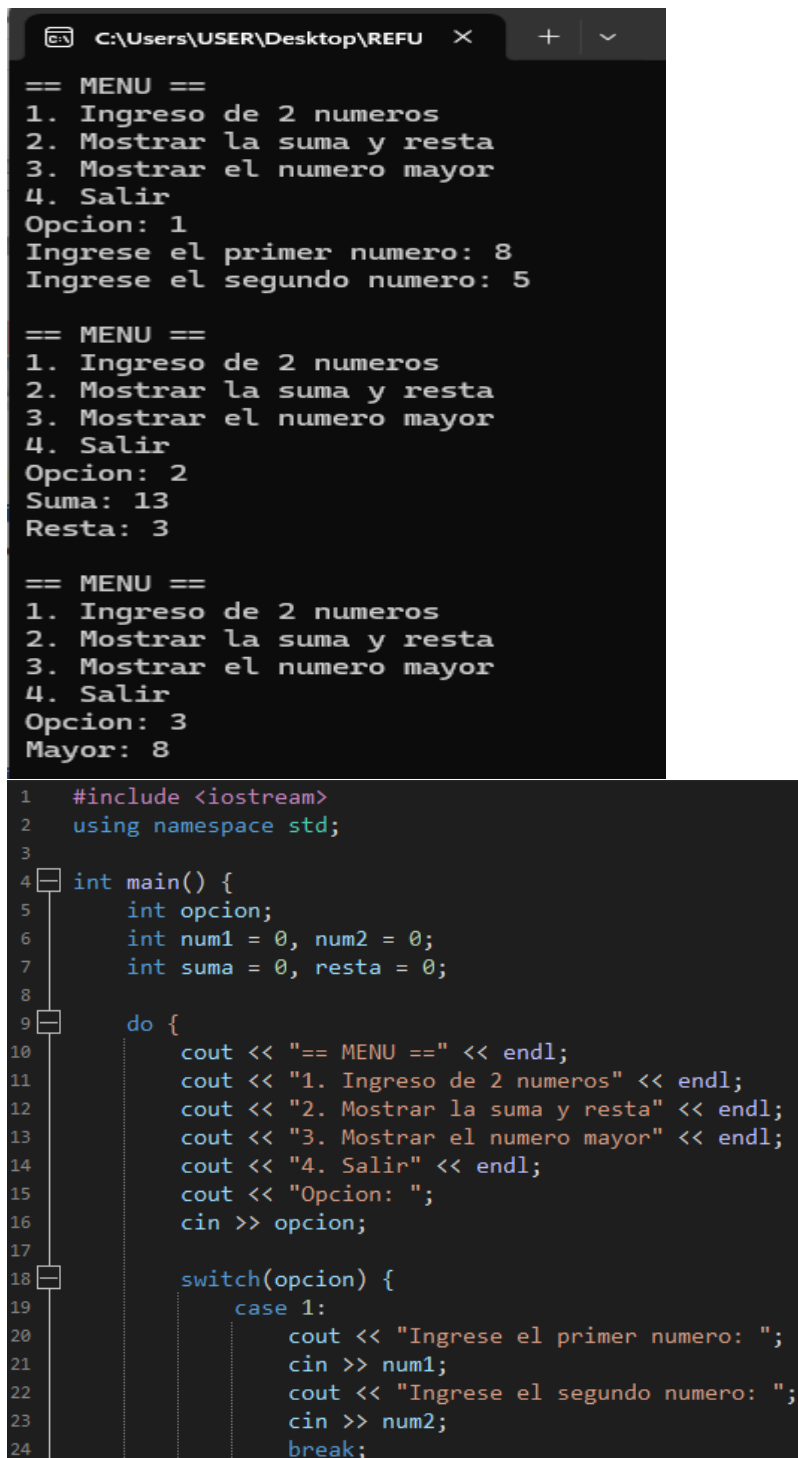
```

MENÚ

Ejercicio 26

Desarrolle un programa que muestre un menú con las siguientes opciones:

1. Ingresar dos números
2. Mostrar la suma y la resta
3. Mostrar el mayor de los dos números
4. Salir



```
== MENU ==
1. Ingreso de 2 numeros
2. Mostrar la suma y resta
3. Mostrar el numero mayor
4. Salir
Opcion: 1
Ingrese el primer numero: 8
Ingrese el segundo numero: 5

== MENU ==
1. Ingreso de 2 numeros
2. Mostrar la suma y resta
3. Mostrar el numero mayor
4. Salir
Opcion: 2
Suma: 13
Resta: 3

== MENU ==
1. Ingreso de 2 numeros
2. Mostrar la suma y resta
3. Mostrar el numero mayor
4. Salir
Opcion: 3
Mayor: 8

1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int opcion;
6      int num1 = 0, num2 = 0;
7      int suma = 0, resta = 0;
8
9      do {
10         cout << "== MENU ==" << endl;
11         cout << "1. Ingreso de 2 numeros" << endl;
12         cout << "2. Mostrar la suma y resta" << endl;
13         cout << "3. Mostrar el numero mayor" << endl;
14         cout << "4. Salir" << endl;
15         cout << "Opcion: ";
16         cin >> opcion;
17
18         switch(opcion) {
19             case 1:
20                 cout << "Ingrese el primer numero: ";
21                 cin >> num1;
22                 cout << "Ingrese el segundo numero: ";
23                 cin >> num2;
24                 break;
```

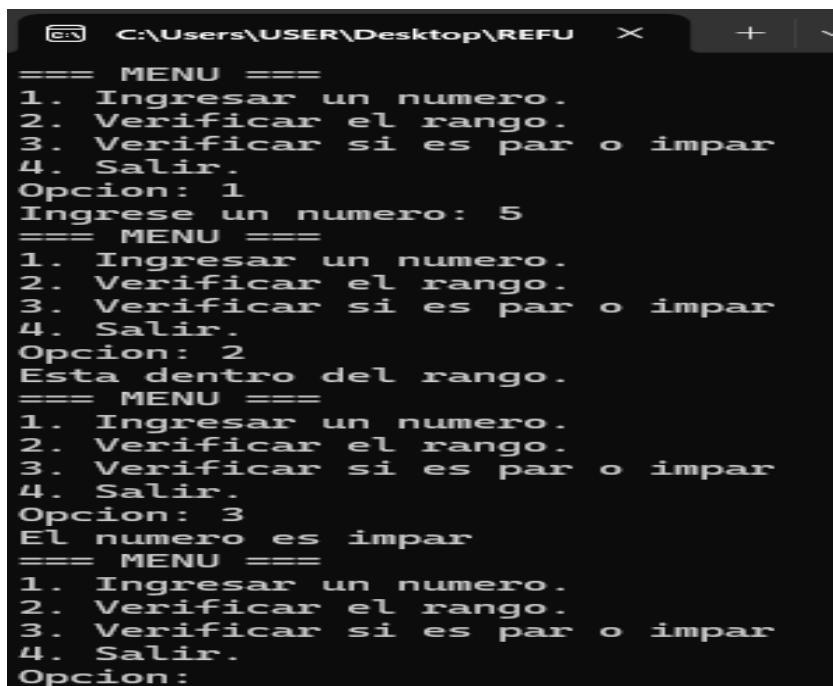


```
25         case 2:
26             suma = num1 + num2;
27             resta = num1 - num2;
28             cout << "Suma: " << suma << endl;
29             cout << "Resta: " << resta << endl;
30             break;
31         case 3:
32             if(num1 == num2) {
33                 cout << "Ambos numeros son iguales: " << num1 << endl;
34             } else if(num1 > num2) {
35                 cout << "Mayor: " << num1 << endl;
36             } else {
37                 cout << "Mayor: " << num2 << endl;
38             }
39             break;
40         case 4:
41             cout << "Saliendo del programa..." << endl;
42             break;
43         default:
44             cout << "Opcion invalida." << endl;
45             break;
46     }
47     cout << endl;
48 } while(opcion != 4);
49
50
51
```

Ejercicio 27

Elabore un programa que muestre un menú con las siguientes opciones:

1. Ingresar un número
2. Verificar si el número está en el rango de 1 a 100
3. Verificar si el número es par o impar
4. Salir



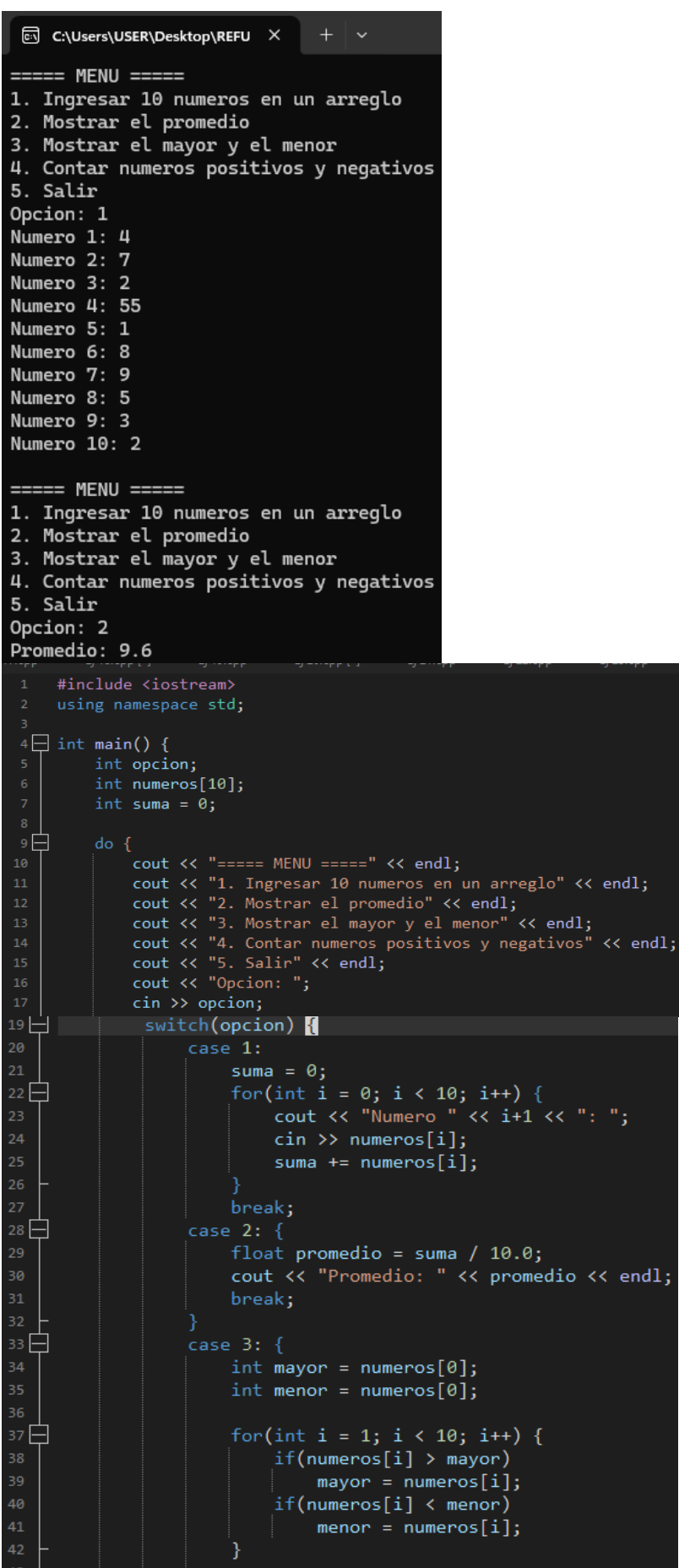
```
C:\Users\USER\Desktop\REFU  X + v
=== MENU ===
1. Ingresar un numero.
2. Verificar el rango.
3. Verificar si es par o impar
4. Salir.
Opcion: 1
Ingrese un numero: 5
=== MENU ===
1. Ingresar un numero.
2. Verificar el rango.
3. Verificar si es par o impar
4. Salir.
Opcion: 2
Esta dentro del rango.
=== MENU ===
1. Ingresar un numero.
2. Verificar el rango.
3. Verificar si es par o impar
4. Salir.
Opcion: 3
El numero es impar
=== MENU ===
1. Ingresar un numero.
2. Verificar el rango.
3. Verificar si es par o impar
4. Salir.
Opcion:
```

```
1  #include<iostream>
2
3  using namespace std;
4
5  int main(){
6
7      int num;
8      int opcion;
9
10     do{
11         cout<<"=== MENU ==="<<endl;
12         cout<<"1. Ingresar un numero."<<endl;
13         cout<<"2. Verificar el rango."<<endl;
14         cout<<"3. Verificar si es par o impar"<<endl;
15         cout<<"4. Salir."<<endl;
16         cout<<"Opcion: ";
17         cin>>opcion;
18
19         switch (opcion) {
20             case 1:
21                 cout<<"Ingrese un numero: ";
22                 cin>>num;
23
24                 break;
25             case 2:
26
27                 if(num > 1 && num < 100){
28                     cout<<"Esta dentro del rango."<<endl;
29                 }else{
30                     cout<<"Fuera del rango"<<endl;
31                 }
32                 break;
33             case 3:
34                 if(num % 2 == 0){
35                     cout<<"El numero es par."<<endl;
36                 }else{
37                     cout<<"El numero es impar"<<endl;
38                 }
39                 break;
40             case 4:
41                 cout<<"Saliendoo.."<<endl;
42                 break;
43             default:
44                 cout<<"Opcion invalida."<<endl;
45                 break;
46         }
47     }while(opcion !=4);
48
49     return 0;
50 }
```

Ejercicio 28

Desarrolle un programa que muestre un menú con las siguientes opciones:

1. Ingresar 10 números en un arreglo
2. Mostrar el promedio
3. Mostrar el mayor y el menor
4. Contar números positivos y negativos
5. Salir



```
===== MENU =====
1. Ingresar 10 numeros en un arreglo
2. Mostrar el promedio
3. Mostrar el mayor y el menor
4. Contar numeros positivos y negativos
5. Salir
Opcion: 1
Numero 1: 4
Numero 2: 7
Numero 3: 2
Numero 4: 55
Numero 5: 1
Numero 6: 8
Numero 7: 9
Numero 8: 5
Numero 9: 3
Numero 10: 2

===== MENU =====
1. Ingresar 10 numeros en un arreglo
2. Mostrar el promedio
3. Mostrar el mayor y el menor
4. Contar numeros positivos y negativos
5. Salir
Opcion: 2
Promedio: 9.6

1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int opcion;
6      int numeros[10];
7      int suma = 0;
8
9      do {
10         cout << "===== MENU =====" << endl;
11         cout << "1. Ingresar 10 numeros en un arreglo" << endl;
12         cout << "2. Mostrar el promedio" << endl;
13         cout << "3. Mostrar el mayor y el menor" << endl;
14         cout << "4. Contar numeros positivos y negativos" << endl;
15         cout << "5. Salir" << endl;
16         cout << "Opcion: ";
17         cin >> opcion;
18
19         switch(opcion) {
20             case 1:
21                 suma = 0;
22                 for(int i = 0; i < 10; i++) {
23                     cout << "Numero " << i+1 << ": ";
24                     cin >> numeros[i];
25                     suma += numeros[i];
26                 }
27                 break;
28             case 2: {
29                 float promedio = suma / 10.0;
30                 cout << "Promedio: " << promedio << endl;
31                 break;
32             }
33             case 3: {
34                 int mayor = numeros[0];
35                 int menor = numeros[0];
36
37                 for(int i = 1; i < 10; i++) {
38                     if(numeros[i] > mayor)
39                         mayor = numeros[i];
40                     if(numeros[i] < menor)
41                         menor = numeros[i];
42                 }
43             }
```

```
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73

    case 4: {
        int positivos = 0;
        int negativos = 0;

        for(int i = 0; i < 10; i++) {
            if(numeros[i] > 0)
                positivos++;
            else if(numeros[i] < 0)
                negativos++;
        }
        cout << "Positivos: " << positivos << endl;
        cout << "Negativos: " << negativos << endl;
        break;
    }
    case 5:
        cout << "Saliendo del programa..." << endl;
        break;
    default:
        cout << "Opcion invalida." << endl;
}
cout << endl;
} while(opcion != 5);
return 0;
```

F. MATRICES

Ejercicio 29

Elabore un programa que permita ingresar una **matriz 3x3** y determine:

- La suma de cada fila
- La suma total de la matriz

```
C:\Users\USER\Desktop\REFU x + v
Ingrese los valores de la matriz 3x3:
Elemento [0][0]: 6
Elemento [0][1]: 3
Elemento [0][2]: 2
Suma de la fila 1: 11
Elemento [1][0]: 9
Elemento [1][1]: 3
Elemento [1][2]: 2
Suma de la fila 2: 14
Elemento [2][0]: 1
Elemento [2][1]: 5
Elemento [2][2]: 7
Suma de la fila 3: 13

Suma total de la matriz: 38

-----
Process exited after 22.14 seconds with return value 0
Presione una tecla para continuar . . .
```

```
4 int main() {
5     int matriz[3][3];
6     int sumaFila;
7     int sumaTotal = 0;
8
9     cout << "Ingrese los valores de la matriz 3x3:" << endl;
10
11     for(int i = 0; i < 3; i++) {
12         sumaFila = 0;
13
14         for(int j = 0; j < 3; j++) {
15             cout << "Elemento [" << i << "][" << j << "]: ";
16             cin >> matriz[i][j];
17
18             sumaFila += matriz[i][j];
19             sumaTotal += matriz[i][j];
20         }
21
22         cout << "Suma de la fila " << i+1 << ": " << sumaFila << endl;
23     }
24
25     cout << "\nSuma total de la matriz: " << sumaTotal << endl;
26
27     return 0;
28 }
```

Ejercicio 30

Desarrolle un programa que permita ingresar una **matriz 2x4** y determine:

- El número mayor
- La posición (fila y columna) del número mayor

```
C:\Users\USER\Desktop\REFU X + v
Ingrese los valores de la matriz 2x4:
Elemento [0][0]: 1
Elemento [0][1]: 2
Elemento [0][2]: 3
Elemento [0][3]: 2
Elemento [1][0]: 1
Elemento [1][1]: 2
Elemento [1][2]: 2
Elemento [1][3]: 2

Numero mayor: 3
Posicion: Fila 1, Columna 3

-----
Process exited after 29.96 seconds with return value 0
Presione una tecla para continuar . . .
```

```
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int matriz[2][4];
6      int mayor;
7      int filaMayor = 0;
8      int columnaMayor = 0;
9
10     cout << "Ingrese los valores de la matriz 2x4:" << endl;
11
12     for(int i = 0; i < 2; i++) {
13         for(int j = 0; j < 4; j++) {
14             cout << "Elemento [" << i << "][" << j << "]: ";
15             cin >> matriz[i][j];
16
17             if(i == 0 && j == 0) {
18                 mayor = matriz[i][j];
19                 filaMayor = i;
20                 columnaMayor = j;
21             }
22
23             if(matriz[i][j] > mayor) {
24                 mayor = matriz[i][j];
25                 filaMayor = i;
26                 columnaMayor = j;
27             }
28         }
29     }
30
31     cout << "\nNumero mayor: " << mayor << endl;
32     cout << "Posicion: Fila " << filaMayor+1
33         << ", Columna " << columnaMayor+1 << endl;
34
35     return 0;
36 }
```

Ejercicio 31

Realice un programa que permita ingresar una **matriz 4x4** y cuente cuántos valores son:

- Positivos
- Negativos
- Iguales a cero

```

C:\Users\USER\Desktop\REFU x + v
Matriz [0][0]: 8
Matriz [0][1]: -5
Matriz [0][2]: 3
Matriz [0][3]: -6
Matriz [1][0]: -9
Matriz [1][1]: 2
Matriz [1][2]: 1
Matriz [1][3]: 11
Matriz [2][0]: -4
Matriz [2][1]: -11
Matriz [2][2]: 10
Matriz [2][3]: 8
Matriz [3][0]: 7
Matriz [3][1]: 9
Matriz [3][2]: 4
Matriz [3][3]: -77
Positivos: 10
Negativos: 6
Igual a cero: 0

-----
Process exited after 69.89 seconds with return value 0
Presione una tecla para continuar . . .

```

```

Ej 28.cpp [*] Ej 29.cpp [*]
3 int main(){
4
5     int matriz[4][4];
6     int positivos = 0;
7     int negativos = 0;
8     int cero = 0;
9
10    for(int i=0; i<4; i++){
11        for(int j=0; j<4; j++){
12            cout<<"Matriz ["<<i<<" "<<j<<"]: ";
13            cin>>matriz[i][j];
14
15            if(matriz[i][j]>0){
16                positivos++;
17            }else if(matriz[i][j]<0){
18                negativos++;
19            }else{
20                cero++;
21            }
22        }
23    }
24    cout<<"Positivos: "<<positivos<<endl;
25    cout<<"Negativos: "<<negativos<<endl;
26    cout<<"Igual a cero: "<<cero<<endl;
27
28    return 0;

```

EJERCICIO 32: struct + vector

Enunciado:

Desarrolle un programa en C++ que defina una estructura Cliente con los campos nombre y edad. El programa debe permitir ingresar varios clientes usando un vector y luego mostrar en pantalla la información de todos los clientes registrados.

```
C:\Users\USER\Desktop\REFU X + v
Numero de clientes: 3
Nombre: Zurie1
Edad: 8
Nombre: Yane
Edad: 47
Nombre: Adriana
Edad: 19
=== CLIENTES ===
Nombre: Zurie1
Edad: 8
Nombre: Yane
Edad: 47
Nombre: Adriana
Edad: 19

-----
Process exited after 23.55 seconds with return value 0
Presione una tecla para continuar . . .

1  #include<iostream>
2  #include<vector>
3
4  using namespace std;
5
6  struct Cliente{
7      string nombre;
8      int edad;
9  };
10
11 int main(){
12
13     vector<Cliente> clientes;
14     int num;
15
16     Cliente c;
17
18     cout<<"Numero de clientes: ";
19     cin>>num;
```



```

11 int main(){
12     vector<Cliente> clientes;
13     int num;
14
15     Cliente c;
16     cout<<"Numero de clientes: ";
17     cin>>num;
18
19     for(int i=0;i<num;i++){
20         cout<<"Nombre: ";
21         cin.ignore();
22         getline(cin,c.nombre);
23         cout<<"Edad: ";
24         cin>>c.edad;
25
26         clientes.push_back(c);
27     }
28     cout<<"=== CLIENTES ==="<<endl;
29     for(int i=0;i<int(clientes.size());i++){
30
31         cout<<"Nombre: "<<clientes[i].nombre<<endl;
32         cout<<"Edad: "<<clientes[i].edad<<endl;
33     }
34
35     return 0;
36 }

```

EJERCICIO 33: struct + array

Enunciado:

Cree un programa en C++ que utilice una estructura Cliente con los campos nombre y edad. Almacene los datos de tres clientes en un arreglo y muestre únicamente los clientes que sean mayores de edad (edad mayor o igual a 18).

```

1  #include<iostream>
2  using namespace std;
3
4  struct Cliente{
5      string nombre;
6      int edad;
7  };
8  int main(){
9      Cliente clientes[3];
10
11     for(int i=0;i<3;i++){
12         cout<<"Nombre "<<i+1<<": ";
13         cin>>clientes[i].nombre;
14         cout<<"Edad: "<<i+1<<": ";
15         cin>>clientes[i].edad;
16     }
17     cout<<"=== CLIENTES MAYORES DE 18 ==="<<endl;
18
19     for(int i=0;i<3;i++){
20         if(clientes[i].edad>=18){
21             cout<<"Nombre: "<<clientes[i].nombre<<endl;
22             cout<<"Edad: "<<clientes[i].edad<<endl;
23         }
24     }
25
26     return 0;
27 }

```

```
C:\Users\USER\Desktop\REFU X + v
Nombre 1: Emely
Edad: 1: 21
Nombre 2: Maite
Edad: 2: 8
Nombre 3: Cristipher
Edad: 3: 22
=== CLIENETES MAYORES DE 18 ===
Nombre: Emely
Edad: 21
Nombre: Cristipher
Edad: 22

-----
Process exited after 20.75 seconds with return value 0
Presione una tecla para continuar . . .
```

EJERCICIO 34: struct + vector + archivo

Enunciado:

Realice un programa en C++ que use una estructura Cliente con los campos nombre y edad. El programa debe almacenar los datos en un vector, guardarlos en un archivo de texto y permitir buscar un cliente por su nombre, mostrando su edad si se encuentra registrado.

```
1  #include<iostream>
2  #include<vector>
3  #include<fstream>
4  using namespace std;
5
6  struct Cliente{
7      string nombre;
8      int edad;
9  };
10
11  int main(){
12
13      vector<Cliente> clientes;
14      Cliente c;
15      string buscar;
16      int num;
17      int encontrado = 0;
18
19      cout<<"Cuantos clienets: ";
20      cin>>num;
```

```

22     cout<<"====="<<endl;
23     for(int i=0;i<num;i++){
24         cout<<"Nombre: ";
25         cin>>c.nombre;
26         cout<<"Edad: ";
27         cin>>c.edad;
28
29         clientes.push_back(c);
30     }
31
32     fstream archivo("clientes.txt");
33     for(int i=0;i<clientes.size();i++){
34         archivo<<clientes[i].nombre<<" "<<clientes[i].edad<<endl;
35     }
36     archivo.close();
37
38     cout<<"Ingrese el nombre a buscar: ";
39     cin>>buscar;
40
41     cout<<endl;
42
43     cout<<"==== CLIENTE =====<<endl;
44
45     for(int i=0;i<clientes.size();i++){
46         if(clientes[i].nombre==buscar){
47             cout<<"Nombre: "<<clientes[i].nombre<<endl;
48             cout<<"Edad: "<<clientes[i].edad<<endl;
49
50             encontrado = 1;
51         }
52     }
53
54     if(encontrado==0){
55         cout<<"No se encontro"<<endl;
56     }
57
58     return 0;
59 }
60

```

```

C:\Users\USER\Desktop\REFU x + v
Cuantos clienets: 2
=====
Nombre: Emely
Edad: 21
Nombre: Crsitopher
Edad: 20
Ingrese el nombre a buscar: Emely

==== CLIENTE =====
Nombre: Emely
Edad: 21

-----
Process exited after 16.2 seconds with return value 0
Presione una tecla para continuar . . .

```

EJERCICIO 35: struct + vector + archivo + cálculos

Enunciado:

Desarrolle un programa en C++ que defina una estructura Cliente con los campos nombre, edad y consumoMensual. El programa debe almacenar los datos en un vector, guardarlos en un archivo, leerlos posteriormente y calcular el consumo total, el consumo promedio y el cliente con mayor consumo.

```
C:\Users\USER\Desktop\REFU X + v
Cuantos clientes desea ingresar: 2

Cliente 1
Nombre: Emely
Edad: 20
Consumo mensual: 30.99

Cliente 2
Nombre: Maite
Edad: 19
Consumo mensual: 10.99

--- RESULTADOS ---
Consumo total: 41.98
Consumo promedio: 20.99
Cliente con mayor consumo: Emely

-----
Process exited after 23.26 seconds with return value 0
Presione una tecla para continuar . . .
```

EJERCICIO 36: struct + archivo + búsqueda y cálculos

Enunciado:

Cree un programa en C++ que utilice una estructura Estudiante con los campos nombre, nota1, nota2 y nota3. El programa debe guardar los datos en un archivo, leerlos, calcular el promedio de cada estudiante y permitir buscar un estudiante por nombre para mostrar sus notas, su promedio y si aprueba o reprueba (promedio mayor o igual a 7).

```
1  #include<iostream>
2  #include<vector>
3  #include<fstream>
4
5  using namespace std;
6
7  struct Estudiante{
8      string nombre;
9      float nota1;
10     float nota2;
11     float nota3;
12 };
13
14 int main(){
15
16     vector<Estudiante> estudiantes;
17     Estudiante estudiante;
18     int cantidad;
19     string buscar;
20
21     cout<<"Cuantos estudiantes desea ingresar: ";
22     cin>>cantidad;
```

```
24     for(int i=0;i<cantidad;i++){
25         cout<<"\nEstudiante "<<i+1<<endl;
26         cout<<"Nombre: ";
27         cin>>estudiante.nombre;
28         cout<<"Nota1: ";
29         cin>>estudiante.nota1;
30         cout<<"Nota2: ";
31         cin>>estudiante.nota2;
32         cout<<"Nota3: ";
33         cin>>estudiante.nota3;
34
35         estudiantes.push_back(estudiante);
36     }
37
38     ofstream archivo("estudiantes.txt");
39     for(int i=0;i<estudiantes.size();i++){
40         archivo<<estudiantes[i].nombre<<" "
41             <<estudiantes[i].nota1<<" "
42             <<estudiantes[i].nota2<<" "
43             <<estudiantes[i].nota3<<endl;
44     }
45     archivo.close();
46
47     estudiantes.clear();
48
49
53     cout<<"\nIngrese el nombre a buscar: ";
54     cin>>buscar;
55
56     bool encontrado=false;
57
58     for(int i=0;i<estudiantes.size();i++){
59         if(estudiantes[i].nombre==buscar){
60
61             float promedio=(estudiantes[i].nota1+
62                 estudiantes[i].nota2+
63                 estudiantes[i].nota3)/3;
64
65             cout<<"\nNota1:"<<estudiantes[i].nota1<<endl;
66             cout<<"Nota2:"<<estudiantes[i].nota2<<endl;
67             cout<<"Nota3:"<<estudiantes[i].nota3<<endl;
68             cout<<"Promedio:"<<promedio<<endl;
69
70             if(promedio>=7)
71                 cout<<"Estado:APRUEBA"<<endl;
72             else
73                 cout<<"Estado:REPRUEBA"<<endl;
74             encontrado=true;
75             break;
76         }
77     }
78     if(!encontrado){
79         cout<<"Estudiante no registrado"<<endl;
80     }
```

```

C:\Users\USER\Desktop\REFU X + v
Cuantos estudiantes desea ingresar: 2

Estudiante 1
Nombre: Emely
Nota1: 9
Nota2: 9.80
Nota3: 8

Estudiante 2
Nombre: Cris
Nota1: 7.50
Nota2: 6
Nota3: 9

Ingrese el nombre a buscar: Emely

Nota1:9
Nota2:9.8
Nota3:8
Promedio:8.93333
Estado:APRUEBA

-----
Process exited after 28.67 seconds with return value 0
Presione una tecla para continuar . . .

```

EJERCICIO 37: funciones + paso por referencia + vector

Enunciado:

Desarrolle un programa en C++ que utilice funciones con paso de parámetros por referencia y un vector de números enteros. El programa debe incluir funciones para llenar el vector y para calcular la suma, el promedio, el valor mayor y el valor menor de los elementos almacenados.

```

1  #include<iostream>
2  #include<vector>
3
4  using namespace std;
5
6  void llenarVector(vector<int> &numeros){
7      int i=0;
8      while (i<numeros.size()){
9          cout<<"Ingrese un numero "<<i+1<<":
10         cin>>numeros[i];
11         i++;
12     }
13 }
14
15 void calculo(vector<int> &numeros){
16     int i=0;
17     int suma=0;
18     int mayor=numeros[0];
19     int menor=numeros[0];
20     float promedio;
21

```

```

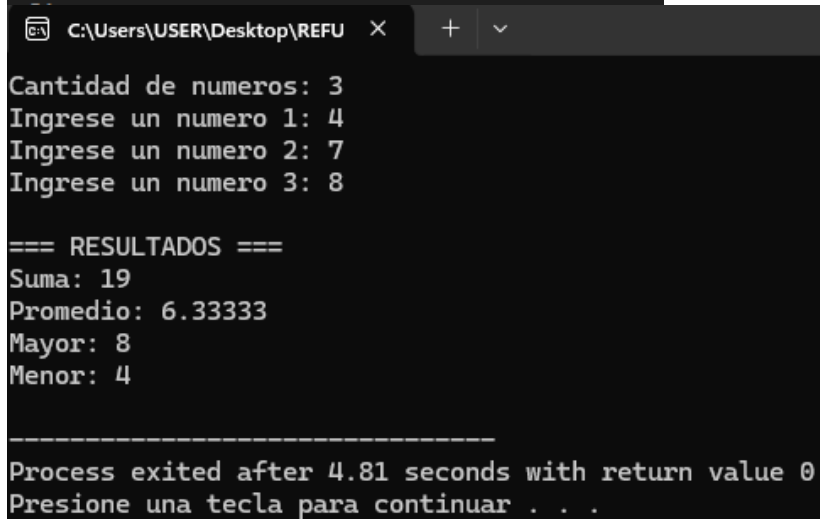
22 while (i < numeros.size()) {
23     suma += numeros[i];
24
25     if (numeros[i] > mayor)
26         mayor = numeros[i];
27
28     if (numeros[i] < menor)
29         menor = numeros[i];
30
31     i++;
32 }
33 promedio = (float)suma / numeros.size();
34
35 cout << "\n=== RESULTADOS ===" << endl;
36 cout << "Suma: " << suma << endl;
37 cout << "Promedio: " << promedio << endl;
38 cout << "Mayor: " << mayor << endl;
39 cout << "Menor: " << menor << endl;
40 }

```

```

43 int main() {
44     int n;
45
46     cout << "Cantidad de numeros: ";
47     cin >> n;
48     vector<int> numeros(n);
49
50     llenarVector(numeros);
51     calculo(numeros);
52     return 0;
53 }

```



The screenshot shows a Windows command prompt window with the title "C:\Users\USER\Desktop\REFU". The program prompts for the number of elements and then for each element. The output shows the sum, average, maximum, and minimum of the entered numbers.

```

C:\Users\USER\Desktop\REFU >
Cantidad de numeros: 3
Ingrese un numero 1: 4
Ingrese un numero 2: 7
Ingrese un numero 3: 8

=== RESULTADOS ===
Suma: 19
Promedio: 6.33333
Mayor: 8
Menor: 4

-----
Process exited after 4.81 seconds with return value 0
Presione una tecla para continuar . . .

```

EJERCICIO 38: funciones + paso por referencia + array

Enunciado:

Cree un programa en C++ que utilice un arreglo de números reales y funciones con paso por referencia. El programa debe llenar el arreglo mediante una función y calcular el total de ventas, el promedio de ventas y cuántas ventas son mayores al promedio.

```
1  #include<iostream>
2  #include<vector>
3
4  using namespace std;
5
6  void llenarArreglo(float ventas[], int n){
7      for (int i = 0; i < n; i++) {
8          cout << "Ingrese venta " << i + 1 << ": ";
9          cin >> ventas[i];
10     }
11 }
12 void calcularDatos(float ventas[], int n){
13     float total = 0, promedio;
14     int mayores = 0;
15     for(int i=0;i<n;i++){
16         total += ventas[i];
17     }
18     promedio = total / n;
19
20     for(int i=0;i<n;i++){
21         if(ventas[i]>promedio){
22             mayores ++;
23         }
24     }
25
26     cout << "=== RESULTADOS ===" << endl;
27     cout << "Total de ventas: " << total << endl;
28     cout << "Promedio de ventas: " << promedio << endl;
29     cout << "Ventas mayores al promedio: " << mayores << endl;
30 }
31
32
33 int main() {
34     int n;
35
36     cout << "Cantidad de ventas: ";
37     cin >> n;
38
39     float ventas[n];
40
41     llenarArreglo(ventas, n);
42     calcularDatos(ventas, n);
43
44     return 0;
45 }
```

```
C:\Users\USER\Desktop\REFU x + v
Cantidad de ventas: 2
Ingrese venta 1: 20
Ingrese venta 2: 90
=== RESULTADOS ===
Total de ventas: 110
Promedio de ventas: 55
Ventas mayores al promedio: 1

-----
Process exited after 11.71 seconds with return value 0
Presione una tecla para continuar . . .
```

EJERCICIO 39: funciones + paso por valor + vector

Enunciado:

Desarrolle un programa en C++ que utilice funciones con paso de parámetros por valor y un vector de números enteros. El programa debe pasar el vector a las funciones para calcular la suma de los

Prof. Ing. Yadira Franco

elementos, el promedio y determinar cuántos valores son pares e impares, sin modificar el contenido original del vector.

```
1  #include <iostream>
2  #include <vector>
3
4  using namespace std;
5
6  int calcularSuma(vector<int> numeros) {
7      int suma = 0;
8
9      for (int i = 0; i < numeros.size(); i++) {
10         suma += numeros[i];
11     }
12
13     return suma;
14 }
15
16 float calcularPromedio(vector<int> numeros) {
17     int suma = 0;
18
19     for (int i = 0; i < numeros.size(); i++) {
20         suma += numeros[i];
21     }
22
23     return (float)suma / numeros.size();
24 }
25
26 void contarParesImpares(vector<int> numeros) {
27     int pares = 0, impares = 0;
28
29     for (int i = 0; i < numeros.size(); i++) {
30         if (numeros[i] % 2 == 0)
31             pares++;
32         else
33             impares++;
34     }
35     cout << "Pares: " << pares << endl;
36     cout << "Impares: " << impares << endl;
37 }
38
39 int main() {
40     int n;
41
42     cout << "Cantidad de numeros: ";
43     cin >> n;
44     vector<int> numeros(n);
45
46     for (int i = 0; i < numeros.size(); i++) {
47         cout << "Ingrese numero " << i + 1 << ": ";
48         cin >> numeros[i];
49     }
50     int suma = calcularSuma(numeros);
51     float promedio = calcularPromedio(numeros);
52
53     cout << "Suma: " << suma << endl;
54     cout << "Promedio: " << promedio << endl;
55
56     contarParesImpares(numeros);
57
58     return 0;
59 }
```

```
C:\Users\USER\Desktop\REFU X + v
Cantidad de numeros: 2
Ingrese numero 1: 8
Ingrese numero 2: 9
Suma: 17
Promedio: 8.5
Pares: 1
Impares: 1

-----
Process exited after 94.43 seconds with return value 0
Presione una tecla para continuar . . .
```

EJERCICIO 40: funciones + paso por valor + array

Enunciado:

Cree un programa en C++ que utilice un arreglo de números enteros y funciones con paso por valor. El programa debe enviar el arreglo a las funciones para calcular el valor mayor, el valor menor y la cantidad de números positivos, negativos y ceros, sin alterar los datos originales.

```
1  #include <iostream>
2
3  using namespace std;
4
5  int mayorValor(int numeros[], int n) {
6      int mayor = numeros[0];
7
8      for (int i = 1; i < n; i++) {
9          if (numeros[i] > mayor)
10             mayor = numeros[i];
11      }
12      return mayor;
13  }
14
15  int menorValor(int numeros[], int n) {
16      int menor = numeros[0];
17
18      for (int i = 1; i < n; i++) {
19          if (numeros[i] < menor)
20             menor = numeros[i];
21      }
22      return menor;
23  }
```

```

25 void contarNumeros(int numeros[], int n) {
26     int positivos = 0, negativos = 0, ceros = 0;
27
28     for (int i = 0; i < n; i++) {
29         if (numeros[i] > 0)
30             positivos++;
31         else if (numeros[i] < 0)
32             negativos++;
33         else
34             ceros++;
35     }
36     cout << "Positivos: " << positivos << endl;
37     cout << "Negativos: " << negativos << endl;
38     cout << "Ceros: " << ceros << endl;
39 }
41 int main() {
42     int n;
43
44     cout << "Cantidad de numeros: ";
45     cin >> n;
46     int numeros[n];
47
48     for (int i = 0; i < n; i++) {
49         cout << "Ingrese numero " << i + 1 << ": ";
50         cin >> numeros[i];
51     }
52     cout << "Mayor: " << mayorValor(numeros, n) << endl;
53     cout << "Menor: " << menorValor(numeros, n) << endl;
54
55     contarNumeros(numeros, n);
56
57     return 0;
58 }

```

```

C:\Users\USER\Desktop\REFU x + v
Cantidad de numeros: 2
Ingrese numero 1: 2
Ingrese numero 2: 5
Mayor: 5
Menor: 2
Positivos: 2
Negativos: 0
Ceros: 0

-----
Process exited after 10.61 seconds with return value 0
Presione una tecla para continuar . . .

```

EJERCICIO 41: EJERCICIO INTEGRADOR

Enunciado:

Desarrolle un programa en C++ que permita gestionar las ventas diarias de una tienda.

El programa 41.1 CALIFICACIONES DE UN ESTUDIANTE

Enunciado:

Desarrolle un programa en C++ que permita gestionar las **calificaciones de un estudiante**.

El programa debe cumplir con los siguientes requisitos:

- Usar un **vector de números reales** para almacenar las calificaciones.

- Utilizar **funciones** para ingresar las calificaciones y realizar los cálculos.
- Guardar las calificaciones en un **archivo de texto**.
- Leer los datos desde el archivo.
- Calcular el **total de calificaciones** y el **promedio**.
- Mostrar cuántas calificaciones están **por encima del promedio** y cuántas están **por debajo del promedio**.
- Evitar repetir cálculos innecesarios.

```
1  #include <iostream>
2  #include <vector>
3  #include <fstream>
4
5  using namespace std;
6
7  void ingresarCalificaciones(vector<double> &calificaciones, int n) {
8      for (int i = 0; i < n; i++) {
9          cout << "Ingrese calificacion " << i + 1 << ": ";
10         cin >> calificaciones[i];
11     }
12 }
13
14 void guardarArchivo(const vector<double> &calificaciones) {
15     ofstream archivo("calificaciones.txt");
16     for (int i = 0; i < calificaciones.size(); i++) {
17         archivo << calificaciones[i] << endl;
18     }
19     archivo.close();
20 }
21
22 vector<double> leerArchivo() {
23     vector<double> calificaciones;
24     ifstream archivo("calificaciones.txt");
25     double valor;
26
27     while (archivo >> valor) {
28         calificaciones.push_back(valor);
29     }
30
31     archivo.close();
32     return calificaciones;
33 }
34
35 double calcularTotal(const vector<double> &calificaciones) {
36     double total = 0;
37     for (int i = 0; i < calificaciones.size(); i++) {
38         total += calificaciones[i];
39     }
40     return total;
41 }
```

```
43 void contarPromedio(const vector<double> &calificaciones, double promedio) {  
44     int mayores = 0, menores = 0;  
45  
46     for (int i = 0; i < calificaciones.size(); i++) {  
47         if (calificaciones[i] > promedio)  
48             mayores++;  
49         else if (calificaciones[i] < promedio)  
50             menores++;  
51     }  
52  
53     cout << "Mayores al promedio: " << mayores << endl;  
54     cout << "Menores al promedio: " << menores << endl;  
55 }  
56  
57 int main() {  
58     int n;  
59     cout << "Cantidad de calificaciones: ";  
60     cin >> n;  
61  
62     vector<double> calificaciones(n);  
63  
64     ingresarCalificaciones(calificaciones, n);  
65     guardarArchivo(calificaciones);  
66  
67     vector<double> datos = leerArchivo();  
68  
69     double total = calcularTotal(datos);  
70     double promedio = total / datos.size();  
71  
72     cout << "Total: " << total << endl;  
73     cout << "Promedio: " << promedio << endl;  
74  
75     contarPromedio(datos, promedio);  
76  
77     return 0;  
78 }
```

```
C:\Users\USER\Desktop\REFU X + v  
Cantidad de calificaciones: 3  
Ingrese calificacion 1: 9  
Ingrese calificacion 2: 8  
Ingrese calificacion 3: 7  
Total: 24  
Promedio: 8  
Mayores al promedio: 1  
Menores al promedio: 1  
  
-----  
Process exited after 7.881 seconds with return value 0  
Presione una tecla para continuar . . .
```

El programa 41.2: VENTAS

Prof. Ing. Yadira Franco

El programa debe desarrollarse en **C++** y compilar correctamente.

Se debe utilizar un **vector<double>** para ingresar y almacenar inicialmente los valores de las ventas.

El ingreso de las ventas debe realizarse mediante una **función** que reciba el vector por referencia.

Una vez ingresados los datos, las ventas deben **guardarse en un archivo de texto** llamado ventas.txt.

El programa debe **leer las ventas desde el archivo** y almacenar esos valores en un **nuevo vector<double>**.

Todos los **cálculos** deben realizarse **únicamente** con el vector cargado desde el archivo.

Se deben usar funciones para:

- Calcular el **total de ventas**.
- Calcular el **promedio de ventas** (este cálculo debe hacerse una sola vez).

Utilizando el promedio calculado, el programa debe determinar:

- Cuántas ventas están **por encima del promedio**.
- Cuántas ventas están **por debajo del promedio**.

El programa debe mostrar en pantalla:

- Total de ventas.
- Promedio de ventas.
- Cantidad de ventas mayores y menores al promedio.

No se permite repetir cálculos innecesarios ni recalcular el promedio dentro de ciclos.

No se permite el uso de variables globales.

El código debe estar correctamente indentado y comentado.

```
1  #include <iostream>
2  #include <vector>
3  #include <fstream>
4
5  using namespace std;
6
7  void ingresarVentas(vector<double> &ventas, int n) {
8      for (int i = 0; i < n; i++) {
9          cout << "Ingrese venta " << i + 1 << ": ";
10         cin >> ventas[i];
11     }
12 }
13
14 void guardarVentas(const vector<double> &ventas) {
15     ofstream archivo("ventas.txt");
16     for (int i = 0; i <= ventas.size(); i++) {
17         archivo << ventas[i] << endl;
18     }
19     archivo.close();
20 }
```

```
22 vector<double> leerVentas() {
23     vector<double> ventas;
24     ifstream archivo("ventas.txt");
25     double valor;
26
27     while (archivo >> valor) {
28         ventas.push_back(valor);
29     }
30
31     archivo.close();
32     return ventas;
33 }
34
35 double calcularTotal(const vector<double> &ventas) {
36     double total = 0;
37     for (int i = 0; i < ventas.size(); i++) {
38         total += ventas[i];
39     }
40     return total;
41 }
42
43 void contarVentas(const vector<double> &ventas, double promedio) {
44     int mayores = 0, menores = 0;
45
46     for (int i = 0; i < ventas.size(); i++) {
47         if (ventas[i] > promedio)
48             mayores++;
49         else if (ventas[i] < promedio)
50             menores++;
51     }
52
53     cout << "Ventas mayores al promedio: " << mayores << endl;
54     cout << "Ventas menores al promedio: " << menores << endl;
55 }
56
57 int main() {
58     int n;
59     cout << "Cantidad de ventas: ";
60     cin >> n;
61
62     vector<double> ventas(n);
63     ingresarVentas(ventas, n);
64     guardarVentas(ventas);
65
66     vector<double> datos = leerVentas();
67
68     double total = calcularTotal(datos);
69     double promedio = total / datos.size();
70
71     cout << "Total de ventas: " << total << endl;
72     cout << "Promedio de ventas: " << promedio << endl;
73
74     contarVentas(datos, promedio);
75
76     return 0;
77 }
78
```

```

C:\Users\USER\Desktop\REFU X + v
Cantidad de ventas: 2
Ingrese venta 1: 9
Ingrese venta 2: 8
Total de ventas: 17
Promedio de ventas: 8.5
Ventas mayores al promedio: 1
Ventas menores al promedio: 1

-----
Process exited after 5.102 seconds with return value 0
Presione una tecla para continuar . . .

```

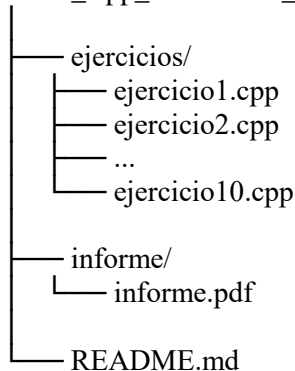
FORMA DE ENTREGA – GITHUB

1 Repositorio

- Crear un repositorio en GitHub con el nombre: **Taller_Cpp_Estructuras_Funciones**

2 Estructura del repositorio

Taller_Cpp_Estructuras_Funciones/



Ejemplo:

Franco - Ejercicio 4 struct vector archivo

INFORME (PDF)

El informe debe contener:

1. Portada
 - Institución
 - Asignatura
 - Nombre del estudiante
 - Docente
 - Fecha
2. Introducción
 - Breve descripción del objetivo del taller
3. Desarrollo
 - Explicación general de lo realizado
 - Capturas de pantalla del código y ejecución
4. Conclusiones
 - Qué se aprendió

- Dificultades encontradas
- 5. Repositorio GitHub
 - Enlace al repositorio

Observaciones importantes

- El código debe compilar correctamente.
- Usar nombres de variables claros.
- Comentar el código cuando sea necesario.
- No copiar código de otros compañeros.

Entrega válida únicamente mediante GitHub.**4. Conclusiones**

Gracias a la realización de este taller aprendí a utilizar de mejor manera las estructuras de control, arreglos y vectores en C++. También reforcé el uso de funciones con paso por valor y por referencia, así como el manejo de archivos para guardar y leer información.

Entre las principales dificultades encontradas estuvo la correcta lógica en algunos cálculos y el manejo adecuado de índices en arreglos y vectores. Sin embargo, al practicar y analizar los errores, logré comprender mejor los conceptos y corregirlos de manera adecuada.