

## EJERCICIOS DE FP

### TALLER No. #3



---

<b>Grupo No.</b>	8
<b>Nombres de los estudiantes</b>	Sarahi Muñoz, Adriana Astudillo, Alan Nero
<b>Carrera</b>	Ingeniería en Electrónica y Automatización
<b>NRC</b>	20823
<b>Nombre del profesor</b>	Jenny Ruiz
<b>Fecha de realización</b>	17/05/2025
<b>Fecha de entrega</b>	19/05/2025

---



## Mis ejercicios de FP

### **EJERCICIO 1-10 ..... 4**

1.1.	Ejercicio 1- Media de dos valores .....	4
1.2.	Ejercicio 2- Valor absoluto de $x$ al cubo. ....	5
1.3.	Ejercicio 3- Divisible .....	6
1.4.	Ejercicio 4- Intervalo .....	7
1.5.	Ejercicio 5- Conversión de unidades de tiempo.....	8
1.6.	Ejercicio 6 - Media de números positivos.....	11
1.7.	Ejercicio 7- Máximo, mínimo y media de alturas.....	13
1.8.	Ejercicio 8- Raíz Cuadrada Entera.....	16
1.9.	Ejercicio 9- Números primos comprendidos entre dos números dados. ....	17

### **EJERCICIO 11-20 ..... 19**

2.1.	Ejercicio 11- Determinar si una persona es mayor de edad.....	19
2.2.	Ejercicio 12- Calcular si un número es par o impar. ....	20
2.3.	Ejercicio 13- Validar ingreso de notas entre 0 y 10.....	21
2.4.	Ejercicio 14- Imprimir los números del 1 al 5 (while) .....	22
2.5.	Ejercicio 15-Determinar si un número es positivo, negativo o cero.....	23
2.6.	Ejercicio 16- Sumar los primeros 10 números naturales.....	24
2.7.	Ejercicio 17- Pedir números hasta ingresar negativos .....	25
2.8.	Ejercicio 18-Mostrar los números pares entre 1 y 10 .....	26
2.9.	Ejercicio 19- Validar contraseña hasta ingresar la correcta .....	27
2.10.	Ejercicio 20- Calcula el promedio de 5 números ingresados .....	28



## Ejercicio 1-10

### 1.1. Ejercicio 1- Media de dos valores

Desarrolle un programa que lea dos números reales del teclado e imprima su media.

Objetivo	Nombre	Valor	Tipo
Número 1	N_1	Variable	Real
Número 2	N_2	Variable	Real
Media	M	Variable	Real
<a href="https://onlinegdb.com/9xIcd9ATH">https://onlinegdb.com/9xIcd9ATH</a>			

#### ➤ Pseudocódigo

```
Proceso Media_Numeros
    Definir N_1, N_2, M Como Real
    Escribir "Ingresa un Primer numero real: "
    Leer N_1
    Escribir "Ingresa el Segundo numero real: "
    Leer N_2
    M = (N_1 + N_2) / 2
    Escribir "La media entre ellos es: " M
FinProceso
```

#### ➤ Código C++

```
#include <iostream>

using namespace std;

int main() {

    double N_1, N_2, M;

    cout << "Ingresa un primer numero real: ";

    cin >> N_1;

    cout << "Ingresa el segundo numero real: ";

    cin >> N_2;

    M = (N_1 + N_2) / 2;

    cout << "La media entre ellos es: " << M << endl;

    return 0;

}
```



### 1.2.Ejercicio 2- Valor absoluto de x al cubo.

Desarrolle un programa que lea un número real x y escriba por pantalla  $|x|^3$ .

Objetivo	Nombre	Valor	Tipo
Número real	"x"	Variable	Real
Valor Absoluto	"r"	Variable	Real
Cubo	"p"	Variable	Real
<a href="https://onlinegdb.com/Fkix8qJNm">https://onlinegdb.com/Fkix8qJNm</a>			

#### ➤ Pseudocódigo.

```
Proceso Valor_Abs
    Definir x, r, p Como Real
    Escribir "Ingresa un numero aleatorio:"
    Leer x
    r = Abs(x)
    si (x<0) Entonces
        x = -x
    FinSi
    p = (x^3)
    Escribir "El valor absoluto es: " x
    Escribir "El valor de la potencia es: " p
FinProceso
```

#### ➤ Código C++

```
#include <iostream>
#include <cmath>
using namespace std;
int main() {
    double x, r, p;
    cout << "Ingresa un numero aleatorio: ";
    cin >> x;
    r = abs(x);
    if (x < 0) {
        x = -x;
    }
    p = pow(r, 3);
    cout << "El valor absoluto es: " << x << endl;
```



```
cout << "El valor de la potencia es: " << p << endl;  
return 0;  
}
```

### 1.3. Ejercicio 3- Divisible

Desarrolle un programa que lea dos números enteros por teclado y determine si el primero de ellos es divisible por el segundo. Se mostrará por pantalla el resultado.

Objetivo	Nombre	Valor	Tipo
Dividendo	N_1	Variable	Entero
Divisor	N_2	Variable	Entero
Resultado	r	Variable	Entero
i	i	Constante	Entero

<https://onlinegdb.com/2gOhfk-fF>

#### ➤ Pseudocódigo

```
Algoritmo divisibilidad  
    Definir N_1, N_2 ,r, i Como Entero  
    Escribir "Ingresar el Dividendo: "  
    leer N_1  
    Escribir "Ingresar el Divisor: "  
    leer N_2  
    i = N_1 % N_2  
    si(i=0) Entonces  
        Escribir "Los numeros si son divisibles entre si."  
    SiNo  
        Escribir "Los numeros no son divisibles entre si."  
    FinSi  
FinAlgoritmo
```

#### ➤ Código C++

```
#include <iostream>  
  
using namespace std;  
  
int main() {  
    int N_1, N_2, r;  
  
    cout << "Ingresar el Dividendo: ";  
  
    cin >> N_1;  
  
    cout << "Ingresar el Divisor: ";
```



```
cin >> N_2;

r = N_1 % N_2;

if (r == 0) {

    cout << "Los numeros si son divisibles entre si." << endl;

} else {

    cout << "Los numeros no son divisibles entre si." << endl;

}

return 0;

}
```

#### 1.4. Ejercicio 4- Intervalo

Desarrolle un programa que lea un número real del teclado y determine si pertenece al intervalo (0,10), indicando por pantalla el resultado.

Objetivo	Nombre	Valor	Tipo
Numero I.	N_1	Variable	Número real
<a href="https://onlinegdb.com/TcAsv-mmvt">https://onlinegdb.com/TcAsv-mmvt</a>			

##### ➤ Pseudocódigo.

```
Proceso Intervalo
    Definir N_1 Como Real
    Escribir "Ingresa un numero entero mayor a 0: "
    Leer N_1
    Si ((N_1>=0) y (N_1<10)) Entonces
        Escribir "Si esta en el intervalo"
    SiNo
        Escribir "No pertenece al intervalo"
    FinSi
FinProceso
```

##### ➤ Código C++

```
#include<iostream>

using namespace std;

int main() {

    float n_1;

    cout << "Ingresa un numero entero mayor a 0: " << endl;
```



```
cin >> n_1;

if (((n_1>=0) && (n_1<10))) {

    cout << "Si esta en el intervalo" << endl;

} else {

    cout << "No pertenece al intervalo" << endl;

}

return 0;

}
```

### 1.5. Ejercicio 5- Conversión de unidades de tiempo

Desarrolle un programa que lea por teclado un valor entero x e, interpretando este valor como el número de segundos que dura un evento, calcule y muestre por pantalla cuántos días, horas, minutos y segundos representa x.

Objetivo	Nombre	Valor	Tipo
Valor	x	Variable	Entero
Días	Dias	Variable	Entero
Horas	Hrs	Variable	Entero
Minutos	Min	Variable	Entero
Segundos	Seg	Variable	Entero
<a href="https://onlinegdb.com/fn-J2YXmo">https://onlinegdb.com/fn-J2YXmo</a>			

➤ Pseudocódigo.

```
Proceso CONVERSION_UNIDADES_TIEMPO
    Definir x, Dias, Hrs, Min, Seg Como Real
    Escribir 'Insertar lo que desea convertir'
    Leer x
    Dias = x/86400
    x = x MOD 86400
    Hrs = x/3600
    x = x MOD 3600
    Min = x/60
    Seg = x MOD 60
    Escribir 'Segundos: ', Seg
    Escribir 'Minutos: ', Min
    Escribir 'Horas: ', Hrs
    Escribir 'Días: ', Dias
FinProceso
```





## ➤ Código C++

```
#include<iostream>

using namespace std;

int main() {

    float anos;

    float dias;

    float horas;

    float meses;

    float minutos;

    int opcion;

    float semanas;

    float valor;

    cout << "Conversor de Unidades de Tiempo" << endl;

    cout << "1. Segundos a Minutos" << endl;

    cout << "2. Minutos a Horas" << endl;

    cout << "3. Horas a Días" << endl;

    cout << "4. Días a Semanas" << endl;

    cout << "5. Semanas a Meses" << endl;

    cout << "6. Meses a Años" << endl;

    cin >> opcion;

    switch (opcion) {

    case 1:

        cout << "Ingrese el valor en segundos: " << endl;

        cin >> valor;

        minutos = valor/60;
```



```
cout << valor << " segundos son " << minutos << " minutos" << endl;  
break;
```

case 2:

```
cout << "Ingrese el valor en minutos: " << endl;  
cin >> valor;  
horas = valor/60;  
cout << valor << " minutos son " << horas << " horas" << endl;  
break;
```

case 3:

```
cout << "Ingrese el valor en horas: " << endl;  
cin >> valor;  
dias = valor/24;  
cout << valor << " horas son " << dias << " días" << endl;  
break;
```

case 4:

```
cout << "Ingrese el valor en días: " << endl;  
cin >> valor;  
semanas = valor/7;  
cout << valor << " días son " << semanas << " semanas" << endl;  
break;
```

case 5:

```
cout << "Ingrese el valor en semanas: " << endl;  
cin >> valor;  
meses = valor/4;  
cout << valor << " semanas son " << meses << " meses" << endl;  
break;
```



case 6:

```
cout << "Ingrese el valor en meses: " << endl;

cin >> valor;

anos = valor/12;

cout << valor << " meses son " << anos << " años" << endl;

break;
```

default:

```
cout << "Opción inválida" << endl;
```

```
}
```

```
return 0;
```

```
}
```

### 1.6. Ejercicio 6 - Media de números positivos

Desarrolle un programa que lea tres números reales del teclado y escriba por pantalla la media aritmética únicamente de los números positivos leídos. Realice el algoritmo en dos versiones: empleando una iteración con salida en cabeza y con salida en cola.

Objetivo	Nombre	Valor	Tipo
Num_1	n1	Variable	Real
Num_2	n2	Variable	Real
Num_3	n3	Variable	Real
Media	R	Variable	Real
Suma	Mas	Constante	Real
Contador	Cont	Constante	Real
<a href="https://onlinegdb.com/v0uynxrYq">https://onlinegdb.com/v0uynxrYq</a>			

➤ Pseudocódigo.

```
➤ Proceso CalcularMediaPositivos
  Definir n1, n2, n3, mas, cont, r Como Real
  mas=0
  cont=0
  Escribir "Insertar el Primer numero"
  Leer n1
  Si n1>0 Entonces
```



```
        mas = mas+n1
        cont = cont+1
    FinSi
    Escribir "Insertar el Segundo numero"
    Leer n2
    Si n2>0 Entonces
        mas = mas+n2
        cont = cont+1
    FinSi
    Escribir "Insertar el Tercer numero"
    Leer n3
    Si n3>0 Entonces
        mas = mas+n3
        cont = cont+1
    FinSi
    Si cont>0 Entonces
        r = mas/cont
        Escribir 'Media: ', r
    SiNo
        Escribir 'No se introdujeron números positivos.'
    FinSi
FinProceso
```

#### ➤ Código C++

```
#include<iostream>

using namespace std;

int main() {

    float i;

    float l;

    float media;

    float numero;

    i = 0;

    l = 0;

    cout << "Ingrese un número positivo:" << endl;

    cout << "Si desea temrinar ingrese un numero negativo" << endl;

    cin >> numero;

    while (numero>0) {

        i = i+numero;

        l = l+1;
```



```
cout << "Ingrese otro número positivo:" << endl;

cout << "Si desea temrnar ingrese un numero negativo" << endl;

cin >> numero;

}

if (l>0) {

    media = i/l;

    cout << "La media de los números ingresados es: " << media << endl;

} else {

    cout << "No se ingresaron números positivos." << endl;

}

return 0;

}
```

### 1.7. Ejercicio 7- Máximo, mínimo y media de alturas.

Desarrolle un programa que lea por teclado un número indeterminado de alturas y que deje de leer cuando se haya introducido una negativa. El programa debe calcular e imprimir la media, el máximo y el mínimo de las alturas introducidas.

Objetivo	Nombre	Valor	Tipo
Altura	N_1	Variable	Real
Contador	contador	Variable	Entero
Máximo	Máx	Variable	Real
Mínimo	Mín	Variable	Real
Suma	Sum		
Media	Med		
Altura_1	N_0		
<a href="https://onlinegdb.com/B-lZnZAdC">https://onlinegdb.com/B-lZnZAdC</a>			

➤ Pseudocódigo.

```
Proceso MaximoMinimoMediaAlturas
// Declaración de variables
Definir N_1, Sum, Med, Max, Min Como Real;
```



```
Definir contador Como Entero;
Definir N_0 Como Logico;
// Inicialización
Sum <- 0;
contador <- 0;
N_0 <- Verdadero;
// Bucle para leer alturas
Repetir
    Escribir 'Ingrese una altura: '; //Ingresa un numero negativo
para terminar
    Leer N_1;
    Si N_1>=0 Entonces
        // Inicializar maximo y minimo con la primera altura
        Si N_0 Entonces
            Max <- N_1;
            Min <- N_1;
            N_0 <- Falso;
        SiNo
            // Actualizar máximo y mínimo
            Si N_1>Max Entonces
                Max <- N_1;
            FinSi
            Si N_1<Min Entonces
                Min <- N_1;
            FinSi
        FinSi
        // Acumular para la media
        Sum <- Sum+N_1;
        contador <- contador+1;
    FinSi
Hasta Que N_1<0
// Mostrar resultados
Si contador>0 Entonces
    Med <- Sum/contador;
    Escribir 'Media de alturas: ', Med;
    Escribir 'Altura máxima: ', Max;
    Escribir 'Altura mínima: ', Min;
SiNo
    Escribir 'No se ingresaron alturas válidas.';
FinSi
FinProceso
```

### ➤ Código C++

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    float N_1, Sum = 0, Med, Max, Min;
```

```
    int contador = 0;
```



```
bool N_0 = true;

do {

    cout << "Ingrese una altura: "; // Ingrese un número negativo para terminar

    cin >> N_1;

    if (N_1 >= 0) {

        if (N_0) {

            Max = N_1;

            Min = N_1;

            N_0 = false;

        } else {

            if (N_1 > Max) {

                Max = N_1;

            }

            if (N_1 < Min) {

                Min = N_1;

            }

        }

        Sum += N_1;

        contador++;

    }

} while (N_1 >= 0);

if (contador > 0) {

    Med = Sum / contador;

    cout << "Media de alturas: " << Med << endl;

    cout << "Altura máxima: " << Max << endl;

    cout << "Altura mínima: " << Min << endl;

} else {
```

---



```
cout << "No se ingresaron alturas válidas." << endl;
```

```
}
```

```
return 0;
```

```
}
```

### 1.8.Ejercicio 8- Raíz Cuadrada Entera

Desarrolle un programa que calcule la raíz cuadrada entera por defecto de un número N positivo dado, que se define como el mayor número entero  $R > 0$ , tal que  $R^2 \leq N$ .

Objetivo	Nombre	Valor	Tipo
Numero	N_1	Variable	Entero
Contador	i	Variable	Entero
Raíz	r	Constante	Entero
<a href="https://onlinegdb.com/q2ShQ5YOe">https://onlinegdb.com/q2ShQ5YOe</a>			

#### ➤ Pseudocódigo.

```
Proceso Raiz_de_un_numero
    Definir N_1, r, i Como Real
    Escribir "Ingresar un entero mayor a 0: "
    Leer N_1
    i <- Raiz(N_1)
    r1 <- (i>1) Y (i^2<N_1)
    Si (i>1) Entonces
        Escribir "La Raiz es: ", i
    FinSi
FinProceso
```

#### ➤ Código C++

```
#include <iostream>

#include <cmath> // Para la función sqrt()

using namespace std;

int main() {

    double N_1, i;

    bool r1;

    cout << "Ingresar un entero mayor a 0: ";

    cin >> N_1;
```





```

i = sqrt(N_1);
r1 = (i > 1) && (pow(i, 2) < N_1);
if (i > 1) {
    cout << "La Raiz es: " << i << endl;
}
return 0;
}

```

### 1.9.Ejercicio 9- Números primos comprendidos entre dos números dados.

Desarrolle un programa que muestre por pantalla todos los números primos comprendidos entre dos números enteros introducidos por teclado.

Objetivo	Nombre	Valor	Tipo
Numero 1	N_1	Variable	Entero
Numero 2	N_2	Variable	Entero
Primo	esPrimo	Verdadero/Falso	Logico
Evalúa	i	Variable	Entero
Verificar	j	Variable	Entero
<a href="https://onlinegdb.com/3-RCUiTHy">https://onlinegdb.com/3-RCUiTHy</a>			

➤ Pseudocódigo.

```

Algoritmo NumerosPrimosEnRango
    Definir N_1, N_2, i, j Como Entero;
    Definir esPrimo Como Logico;

    Escribir "Ingrese el primer número (límite inferior):";
    Leer N_1;

    Escribir "Ingrese el segundo número (límite superior):";
    Leer N_2;

    Escribir "Números primos entre ", N_1, " y ", N_2, ":";

    // Asegurar que _1 sea menor que _2
    Si _1 > _2 Entonces
        Definir temp Como Entero;
        temp <- N_1;
        N_1 <- N_2;
        N_2 <- temp;
    FinSi

```



```
// Iterar desde _1 hasta _2
Para i <- N_1 Hasta N_2 Hacer
    // Verificar si i es primo
    esPrimo <- Verdadero;

    Si i <= 1 Entonces
        esPrimo <- Falso;
    Sino
        Para j <- 2 Hasta i / 2 Hacer
            Si i % j == 0 Entonces
                esPrimo <- Falso;
            FinSi
        FinPara
    FinSi
    Si esPrimo Entonces
        Escribir i, " ";
    FinSi
FinPara
FinAlgoritmo
```

#### ➤ Código C++

```
#include <iostream>

using namespace std;

int main() {

    int N_1, N_2, i, j;

    bool esPrimo;

    cout << "Ingresa el primer numero (limite inferior): ";

    cin >> N_1;

    cout << "Ingresa el segundo numero (limite superior): ";

    cin >> N_2;

    cout << "Numeros primos entre " << N_1 << " y " << N_2 << ":" << endl;

    if (N_1 > N_2) {

        int temp = N_1;

        N_1 = N_2;

        N_2 = temp;

    }

    for (i = N_1; i <= N_2; i++) {

        esPrimo = true
```



```
if (i <= 1) {  
    esPrimo = false;  
} else {  
    for (j = 2; j <= i/2; j++) {  
        if (i % j == 0) {  
            esPrimo = false;  
            break;  
        }  
    }  
}  
if (esPrimo) {  
    cout << i << " ";  
}  
}  
cout << endl;  
return 0;  
}
```

## Ejercicio 11-20

### 2.1. Ejercicio 11- Determinar si una persona es mayor de edad.

Objeto	Nombre	Valor	Tipo
Edad	edad	Variable	Entero

➤ Pseudocódigo.

- 1.Inicio
2. Definir edad como Entero
3. Leer edad
4. Si  $\text{edad} \geq 18$



5. Escribir “Es mayor de edad”
6. Sino
7. Escribir” No es mayor de edad”
8. Fin si
9. Fin algoritmo

➤ **CÓDIGO C++**

```
#include <iostream>

using namespace std;

int main() {

    int edad;

    cout << "Por favor, ingrese su edad: ";

    cin >> edad;

    if (edad >= 18) {

        cout << "Eres mayor de edad." << endl;

    } else {

        cout << "Eres menor de edad." << endl;

    }

    return 0;

}
```

*2.2.Ejercicio 12- Calcular si un número es par o impar.*

Objeto	Nombre	Valor	Tipo
Número	N	Variable	Real

➤ **Pseudocódigo.**

1. Inicio
2. Definir N como Real
3. Escribir “Ingrese un número cualquiera”
4. Leer N



5. Si  $N \text{ MOD } 2 == 0$
6. Escribir "Es par"
7. Sino
8. Escribir "Es impar"
9. Fin si
10. Fin algoritmo

➤ Código en C++

```
#include <iostream>

using namespace std;

int main() {

    int numero;

    cout << "Ingrese un numero entero: ";

    cin >> numero;

    if (numero % 2 == 0) {

        cout << "El numero " << numero << " es par." << endl;

    } else {

        cout << "El numero " << numero << " es impar." << endl;

    }

    return 0;

}
```

*2.3.Ejercicio 13- Validar ingreso de notas entre 0 y 10*

Objeto	Nombre	Valor	Tipo
Nota	nota	Variable	Real

➤ Pseudocódigo

1. Inicio
2. Definir nota como Real
3. Repetir



4. Escribir “Ingresa la nota”
5. Leer nota
6. si  $\text{nota} < 0$  o  $\text{nota} > 10$  Entonces
7. Escribir “Nota no valida”
8. Fin si
9. Hasta que  $\text{nota} \geq$  y  $\text{nota} \leq 10$
10. Escribir “Nota valida:”, nota.
11. Fin algoritmo

➤ Código C++

```
#include <iostream>

using namespace std;

int main() {

    float nota;

    do {

        cout << "Ingresa una nota entre 0 y 10: ";

        cin >> nota;

        if(nota < 0 || nota > 10) {

            cout << "Nota invalida. Debe estar entre 0 y 10." << endl;

        }

    } while(nota < 0 || nota > 10);

    cout << "Nota valida ingresada: " << nota << endl;

    return 0;

}
```

*2.4.Ejercicio 14- Imprimir los números del 1 al 5 (while)*

Objeto	Nombre	Valor	Tipo
Números	i	Constante	Entero

➤ Pseudocódigo

Algoritmo imp\_num



```
Definir i Como Entero
i <- 1
Mientras i <= 5 Hacer
    Escribir i
    i <- i + 1
FinMientras
FinAlgoritmo
```

#### ➤ Código C++

```
#include<iostream>

using namespace std;

int main() {

    int i;

    i = 1;

    while (i<=5) {

        cout << i << endl;

        i = i+1;

    }

    return 0;

}
```

#### 2.5.Ejercicio 15-Determinar si un número es positivo, negativo o cero

Objeto	Nombre	Valor	Tipo
Números	N	Variable	Entero

#### ➤ Pseudocódigo

```
Algoritmo Determinar_Signo
    Definir N1 Como Entero
    Leer N1
    Si N1>0 Entonces
        Escribir 'El número es positivo'
    SiNo
        Si N1<0 Entonces
            Escribir 'El número es negativo'
        SiNo
            Escribir 'Es cero'
        FinSi
    FinSi
FinAlgoritmo
```



### ➤ Código C++

```
#include<iostream>
using namespace std;
int main() {
    int n1;
    cin >> n1;
    if (n1>0) {
        cout << "El número es positivo" << endl;
    } else {
        if (n1<0) {
            cout << "El número es negativo" << endl;
        } else {
            cout << "Es cero" << endl;
        }
    }
    return 0;
}
```

### 2.6.Ejercicio 16- Sumar los primeros 10 números naturales

Objeto	Nombre	Valor	Tipo
Números	n	Constante	Entero
# sumar	l	Variable	Entero
# sumar	i	Variable	Entero

### ➤ Pseudocódigo

```
Algoritmo Suma_Numeros
    Definir n,l,i Como Entero
    Escribir "Inserte la cantidad de números a sumar:"
    Leer l
    n <- 0
    Para i <- 1 Hasta l Hacer
        n <- n + i
    FinPara
    Escribir "La suma de los numeros es: " n
FinAlgoritmo
```

### ➤ Código C++

```
#include<iostream>

using namespace std;

int main() {
```





```
int i;

int L;

int n;

cout << "Inserte la cantidad de números a sumar:" << endl;

cin >> L;

n = 0;

for (i=1; i<=L; ++i) {

    n = n+i;

}

cout << "La suma de los numeros es: " << n << endl;

return 0;

}
```

### 2.7. Ejercicio 17- Pedir números hasta ingresar negativos

Objeto	Nombre	Valor	Tipo
Número	N_1	Variable	Entero

#### ➤ Pseudocódigo

Algoritmo HastaNegativo

Definir N\_1 Como Real

Repetir

    Escribir "Ingrese un número:"

    Leer N\_1

Hasta Que N\_1 < 0

    Escribir "Fin del programa"

FinAlgoritmo

#### ➤ Código C++

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```



```
int N_1;

cout << "Ingrese números (negativo para terminar):\n";

while (cout << "> ", cin >> N_1) {

    if (N_1 < 0) break;

    cout << "Número ingresado: " << N_1 << endl;

}

cout << "Fin del programa (se ingresó un negativo o entrada inválida).\n";

return 0;

}
```

### 2.8.Ejercicio 18-Mostrar los números pares entre 1 y 10

Objeto	Nombre	Valor	Tipo
Número	2	Constante	Entero
Par	X	Variable	Entero
<a href="https://onlinegdb.com/-AodpXv4J">https://onlinegdb.com/-AodpXv4J</a>			

#### ➤ Pseudocódigo

Algoritmo Pares1a10

Definir X Como Entero

Para X desde 1 hasta 10 Hacer

Si  $X \% 2 = 0$  Entonces

Escribir X

FinSi

FinPara

FinAlgoritmo

#### ➤ Código C++

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int X; // Definir X como entero
```



```
for (X = 1; X <= 10; X++) {  
    if (X % 2 == 0) {  
        cout << X << " "; // Mostrar X  
    }  
}  
  
return 0;  
}
```

### 2.9.Ejercicio 19- Validar contraseña hasta ingresar la correcta

Objeto	Nombre	Valor	Tipo
Contraseña	CONTRASENA_CORRECTA	Constante	verdadero
Ingreso	Contraseña ingresada	Variable	
<a href="https://onlinegdb.com/WZldZbmBo">https://onlinegdb.com/WZldZbmBo</a>			

#### ➤ Pseudocódigo

Algoritmo ValidarContraseña

Definir CONTRASENA\_CORRECTA, intento Como Cadena

contraseña <- "seC01"

Repetir

    Escribir "Ingrese la contraseña:"

    Leer Contraseña ingresada

Hasta Contraseña ingresada = CONTRASENA\_CORRECTA

    Escribir "Contraseña correcta"

FinAlgoritmo

#### ➤ Código C++

```
#include <iostream>
```

```
#include <string> // Para usar el tipo string
```

```
using namespace std;
```

```
int main() {
```

```
    const string CONTRASENA_CORRECTA = "seC01"; // Contraseña definida
```



```
string contrasenaIngresada;

bool esCorrecta = false;

cout << "=== Sistema de Validacion de Contraseña ===" << endl;

while (!esCorrecta) {

    cout << "Ingrese la contraseña: ";

    cin >> contrasenaIngresada;

    if (contrasenaIngresada == CONTRASENA_CORRECTA) {

        esCorrecta = true;

        cout << "¡Contraseña correcta! Acceso concedido." << endl;

    } else {

        cout << "Contraseña incorrecta. Intente nuevamente." << endl;

    }

}

return 0;

}
```

#### 2.10. Ejercicio 20- Calcula el promedio de 5 números ingresados

Objeto	Nombre	Valor	Tipo
Números 1	N_1	Variable	Real
Números 2	N_2	Variable	Real
Números 3	N_3	Variable	Real
Números 4	N_4	Variable	Real
Números 5	N_5	Variable	Real
Promedio	X	Variable	Real
<a href="https://onlinegdb.com/krBvSj-Sjo">https://onlinegdb.com/krBvSj-Sjo</a>			

#### ➤ Pseudocódigo

Algoritmo Promedio5Numeros

Definir N1, N2, N3, N4, N5, X promedio Como Real

Escribir "Ingrese 5 números:"



Leer N1, N2, N3, N4, N5

$X \leftarrow (N1 + N2 + N3 + N4 + N5) / 5$

Escribir "El promedio es:", X

FinAlgoritmo

#### ➤ Código C++

```
#include <iostream>
```

```
#include <iomanip>
```

```
using namespace std;
```

```
float leerNotaValida(int numeroNota) {
```

```
    float nota;
```

```
    do {
```

```
        cout << "Ingrese la nota " << numeroNota << " (0-10): ";
```

```
        cin >> nota;
```

```
        if (cin.fail() || nota < 0 || nota > 10) {
```

```
            cin.clear();
```

```
            cin.ignore(1000, '\n');
```

```
            cout << "Error: Debe ingresar un número entre 0 y 10\n";
```

```
        } else {
```

```
            break;
```

```
        }
```

```
    } while (true);
```

```
    return nota;
```

```
}
```

```
int main() {
```

```
    float N_1, N_2, N_3, N_4, N_5, promedio;
```

```
    cout << "CALCULADORA DE PROMEDIO (5 NOTAS)\n";
```

```
    cout << "-----\n"
```

```
    N_1 = leerNotaValida(1);
```



```
N_2 = leerNotaValida(2);
```

```
N_3 = leerNotaValida(3);
```

```
N_4 = leerNotaValida(4);
```

```
N_5 = leerNotaValida(5);
```

```
promedio = (N_1 + N_2 + N_3 + N_4 + N_5) / 5;
```

```
cout << fixed << setprecision(2);
```

```
cout << "\nEl promedio de las notas es: " << promedio << endl;
```

```
if (promedio >= 6.0) {
```

```
    cout << "Aprobado" << endl;
```

```
} else {
```

```
    cout << "Reprobado" << endl;
```

```
}
```

```
return 0;
```

```
}
```