

Benemérita Universidad Autónoma de Puebla  
Facultad de Ciencias de la Computación

# METODOLOGÍA DE LA PROGRAMACIÓN

## - ÁREA DE PROGRAMACIÓN -

---

UNIDAD 1. ARREGLOS, CADENAS Y REGISTROS  
PROBLEMARIO 1: UNIDAD 1. ARREGLOS, CADENAS Y REGISTROS



Docente:  
Prof.<sup>a</sup>. Guillermina Sánchez Román

Alumno:  
Jesús Huerta Aguilar

Matricula:  
202041509

NRC: 31673  
Sección: 005

PRIMER SEMESTRE

Puebla, Pue.

Fecha de entrega: 04/03/2021

1. Hacer un pseudocódigo para crear un arreglo de 31 días. La estructura registrara las temperaturas registradas para el mes de abril, estas oscilan entre los 7 y 38 grados. El algoritmo debe hacer lo siguiente.
  - a. Obtener la temperatura más alta y baja del mes.
  - b. Obtener la temperatura promedio del mes.

## PSEUDOCÓDIGO

```
1  Algoritmo temperatura
2      Definir t,i,max,min Como Entero
3      Definir ptem Como Real
4      max <- 0
5      min <- 38
6      Dimension t[31]
7      Para i<-1 Hasta 31 Con Paso 1 Hacer
8          // ingreso de temperaturas
9          Repetir
10             Escribir '¿Qué temperatura hubo el día ',i,'?'
11             Leer t[i]
12             Hasta Que t[i]>=7 Y t[i]<=38
13             // suma para el promedio
14             ptem <- ptem+t[i]
15             // definir temperaturas mayores y menores
16             Si t[i]<=min Entonces
17                 min <- t[i]
18             SiNo
19                 Si t[i]>max Entonces
20                     max <- t[i]
21                 FinSi
22             FinSi
23         FinPara
24         // obtener promedio
25         ptem <- ptem/31
26         Escribir 'La temperatura más alta fue: ',max,', y la más baja fue: ',min
27         Escribir 'El promedio es: ',ptem
28     FinAlgoritmo
```

**EJECUCIÓN:** Prueba con los 31 días.

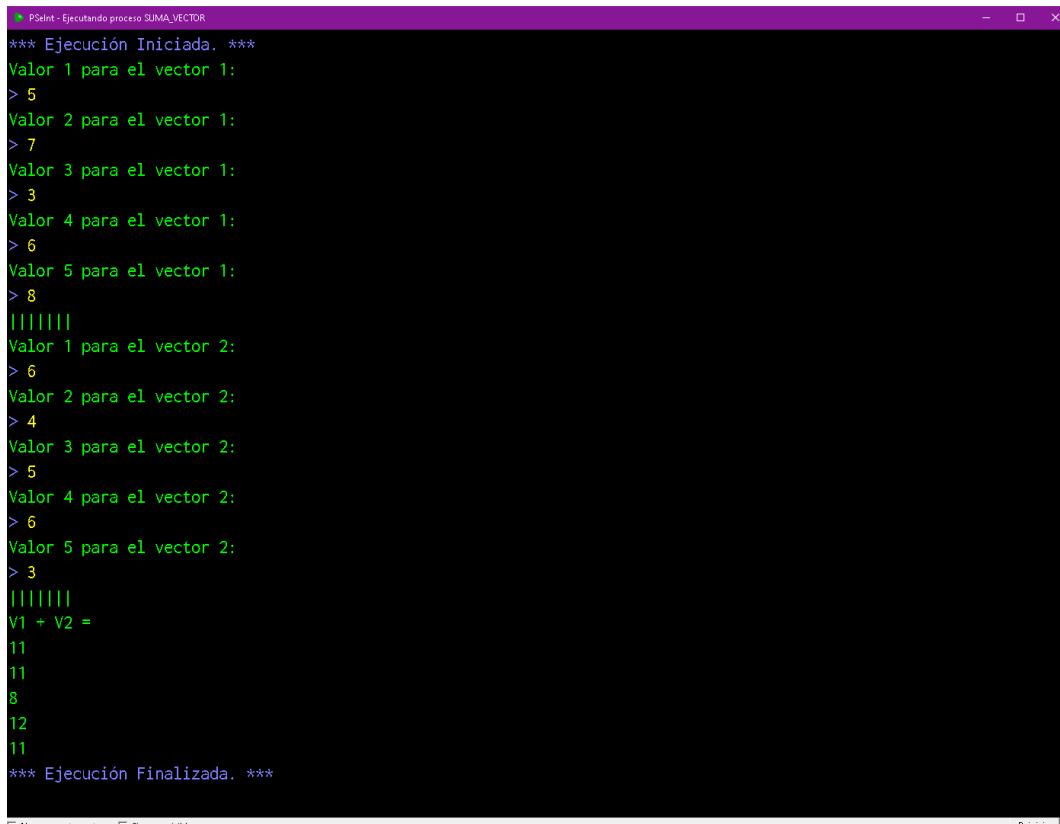
```
PSInt - Ejecutando proceso TEM...
*** Ejecución Iniciada. ***
¿que temperatura hubo el dia 1?
> 9
¿que temperatura hubo el dia 2?
> 11
¿que temperatura hubo el dia 3?
> 12
¿que temperatura hubo el dia 4?
> 15
¿que temperatura hubo el dia 5?
> 14
¿que temperatura hubo el dia 6?
> 16
¿que temperatura hubo el dia 7?
> 2
¿que temperatura hubo el dia 7?
> -9999
¿que temperatura hubo el dia 7?
> 16
¿que temperatura hubo el dia 8?
> 19
¿que temperatura hubo el dia 9?
> 18
¿que temperatura hubo el dia 10?
> 20
¿que temperatura hubo el dia 11?
> 19
¿que temperatura hubo el dia 12?
> 22
¿que temperatura hubo el dia 13?
> 20
¿que temperatura hubo el dia 14?
> 21
¿que temperatura hubo el dia 15?
> 25
¿que temperatura hubo el dia 16?
```

```
PSInt - Ejecutando proceso TEM...
¿que temperatura hubo el dia 16?
> 24
¿que temperatura hubo el dia 17?
> 23
¿que temperatura hubo el dia 18?
> 26
¿que temperatura hubo el dia 19?
> 24
¿que temperatura hubo el dia 20?
> 27
¿que temperatura hubo el dia 21?
> 25
¿que temperatura hubo el dia 22?
> 27
¿que temperatura hubo el dia 23?
> 29
¿que temperatura hubo el dia 24?
> 27
¿que temperatura hubo el dia 25?
> 28
¿que temperatura hubo el dia 26?
> 30
¿que temperatura hubo el dia 27?
> 32
¿que temperatura hubo el dia 28?
> 29
¿que temperatura hubo el dia 29?
> 31
¿que temperatura hubo el dia 30?
> 35
¿que temperatura hubo el dia 31?
> 32
La temperatura mas alta fue: 35,
y la mas baja fue: 9
El promedio es: 22.7741935484
*** Ejecución Finalizada. ***
```

2. Hacer un pseudocódigo para sumar dos vectores de 100 enteros cada uno.

```
1  Algoritmo suma_vector
2      Definir V,i,j Como Entero
3      Dimension V[100,2]
4      Para j←1 Hasta 2 Con Paso 1 Hacer
5          //Cambio de vectores 1 y 2
6          Para i←1 Hasta 100 Con Paso 1 Hacer
7              //Registro de vectores 1 y 2
8              Escribir 'Valor ',i,' para el vector ',j,':'
9              Leer V[i,j]
10         Fin Para
11         Escribir '|||||'
12     Fin Para
13     //suma de vectores 1 y 2
14     Escribir "V1 + V2 ="
15     Para i ← 1 Hasta 100 Con Paso 1 Hacer
16         Escribir V[i,1] + V[i,2]
17     Fin Para
18 FinAlgoritmo
```

**EJECUCIÓN:** Prueba con 5 valores por cada vector.

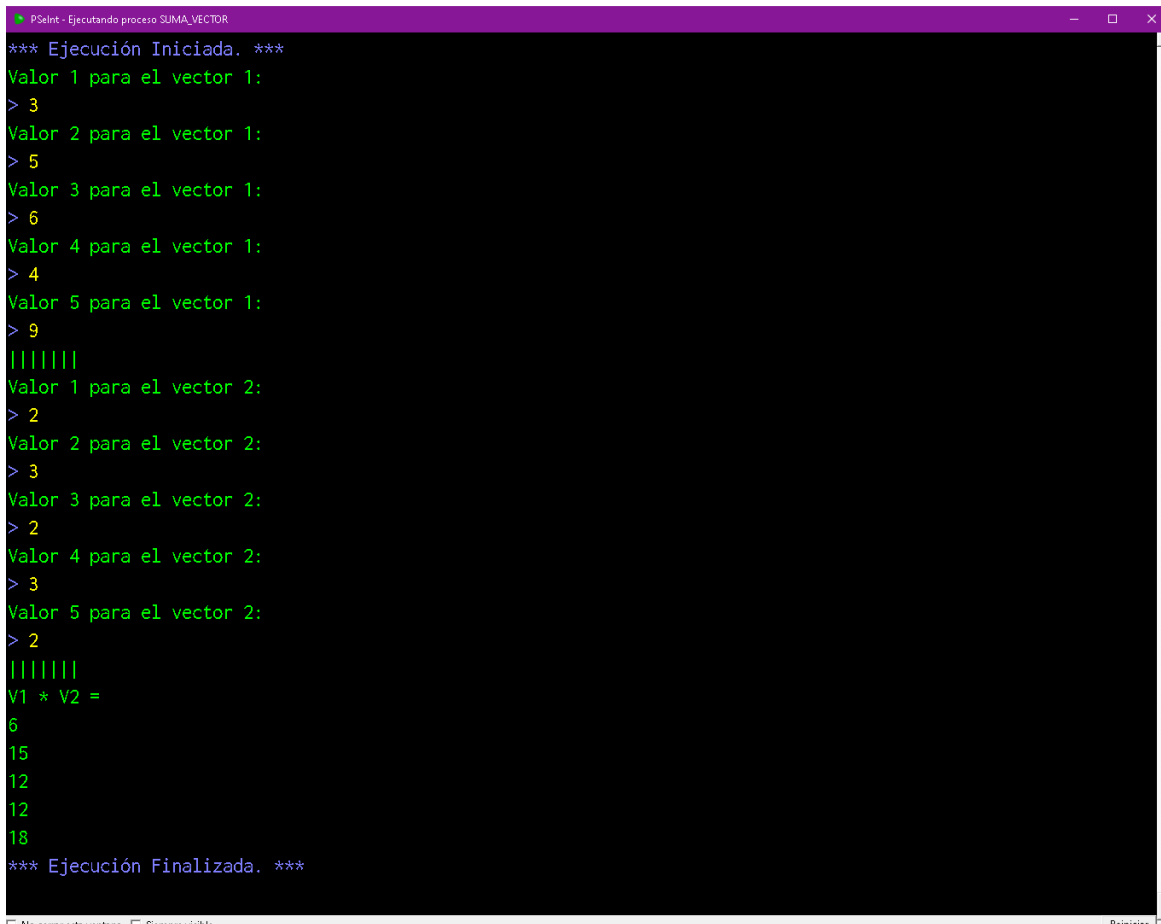


```
PS/Int - Ejecutando proceso SUMA_VECTOR
*** Ejecución Iniciada. ***
Valor 1 para el vector 1:
> 5
Valor 2 para el vector 1:
> 7
Valor 3 para el vector 1:
> 3
Valor 4 para el vector 1:
> 6
Valor 5 para el vector 1:
> 8
|||||
Valor 1 para el vector 2:
> 6
Valor 2 para el vector 2:
> 4
Valor 3 para el vector 2:
> 5
Valor 4 para el vector 2:
> 6
Valor 5 para el vector 2:
> 3
|||||
V1 + V2 =
11
11
8
12
11
*** Ejecución Finalizada. ***
```

3. Hacer un pseudocódigo para multiplicar dos vectores de 50 elementos cada uno.

```
1  Algoritmo suma_vector
2  Definir V,i,j Como Entero
3  TV ← 0
4  Dimension V[50,2]
5  Para j←1 Hasta 2 Con Paso 1 Hacer
6      //Cambio de vectores 1 y 2
7      Para i←1 Hasta 50 Con Paso 1 Hacer
8          //Registro de vectores 1 y 2
9          Escribir 'Valor ',i,' para el vector ',j,':'
10         Leer V[i,j]
11     Fin Para
12     Escribir '|||||'
13 Fin Para
14 //multiplicación de vectores 1 y 2
15 Escribir "V1 * V2 ="
16 Para i ← 1 Hasta 50 Con Paso 1 Hacer
17     Escribir V[i,1]*V[i,2]
18 Fin Para
19 FinAlgoritmo
```

**EJECUCIÓN:** Prueba con 5 valores por cada vector.



```
PS/Int - Ejecutando proceso SUMA_VECTOR
*** Ejecución Iniciada. ***
Valor 1 para el vector 1:
> 3
Valor 2 para el vector 1:
> 5
Valor 3 para el vector 1:
> 6
Valor 4 para el vector 1:
> 4
Valor 5 para el vector 1:
> 9
|||||
Valor 1 para el vector 2:
> 2
Valor 2 para el vector 2:
> 3
Valor 3 para el vector 2:
> 2
Valor 4 para el vector 2:
> 3
Valor 5 para el vector 2:
> 2
|||||
V1 * V2 =
6
15
12
12
18
*** Ejecución Finalizada. ***
```

4. Hacer un pseudocódigo para contar cuantos números son primos en un arreglo de 30 enteros.

## PSEUDOCÓDIGO

```
1  Algoritmo cont_primo
2      Definir nprim,tprim,n,i,j Como Entero
3      tprim ← 0
4      Dimension n[30]
5      Escribir 'Ingrese valores mayores a 0'
6      Para i←1 Hasta 30 Hacer
7          nprim ← 0
8          // registro de números
9          Repetir
10             Escribir 'Entrada ',i,' de 30'
11             Leer n[i]
12             Si n[i] < 0 Entonces
13                 Escribir 'ERROR: Verifica tus datos'
14             FinSi
15             Hasta Que n[i]>0
16             // determinación de un numero primo
17             Para j←1 Hasta n[i] Hacer
18                 Si n[i] MOD j=0 Entonces
19                     nprim ← nprim+1
20                 FinSi
21             FinPara
22             // contador de numeros primos
23             Si nprim=2 Entonces
24                 tprim ← tprim+1
25             FinSi
26         FinPara
27         Escribir 'Cantidad de números primos registrados: ',tprim
28 FinAlgoritmo
```

**EJECUCIÓN:** Prueba con la mitad del arreglo.

```
PSeInt - Ejecutando proceso CONT_PRIMO
*** Ejecución Iniciada. ***
Ingrese valores mayores a 0
Entrada 1 de 30
> 2
Entrada 2 de 30
> 7
Entrada 3 de 30
> 5
Entrada 4 de 30
> 8
Entrada 5 de 30
> 9
Entrada 6 de 30
> 4
Entrada 7 de 30
> 6
Entrada 8 de 30
> 6
Entrada 9 de 30
> 7
Entrada 10 de 30
>
Entrada 10 de 30
> 3
Entrada 11 de 30
> -2345
ERROR: Verifica tus datos
Entrada 11 de 30
> 7
```

```
PSeInt - Ejecutando proceso CONT_PRIMO
Entrada 5 de 30
> 9
Entrada 6 de 30
> 4
Entrada 7 de 30
> 6
Entrada 8 de 30
> 6
Entrada 9 de 30
> 7
Entrada 10 de 30
>
Entrada 10 de 30
> 3
Entrada 11 de 30
> -2345
ERROR: Verifica tus datos
Entrada 11 de 30
> 7
Entrada 12 de 30
> 4
Entrada 13 de 30
> 9
Entrada 14 de 30
> 8
Entrada 15 de 30
> 3
Cantidad de numeros primos registrados: 7
*** Ejecución Finalizada. ***
```

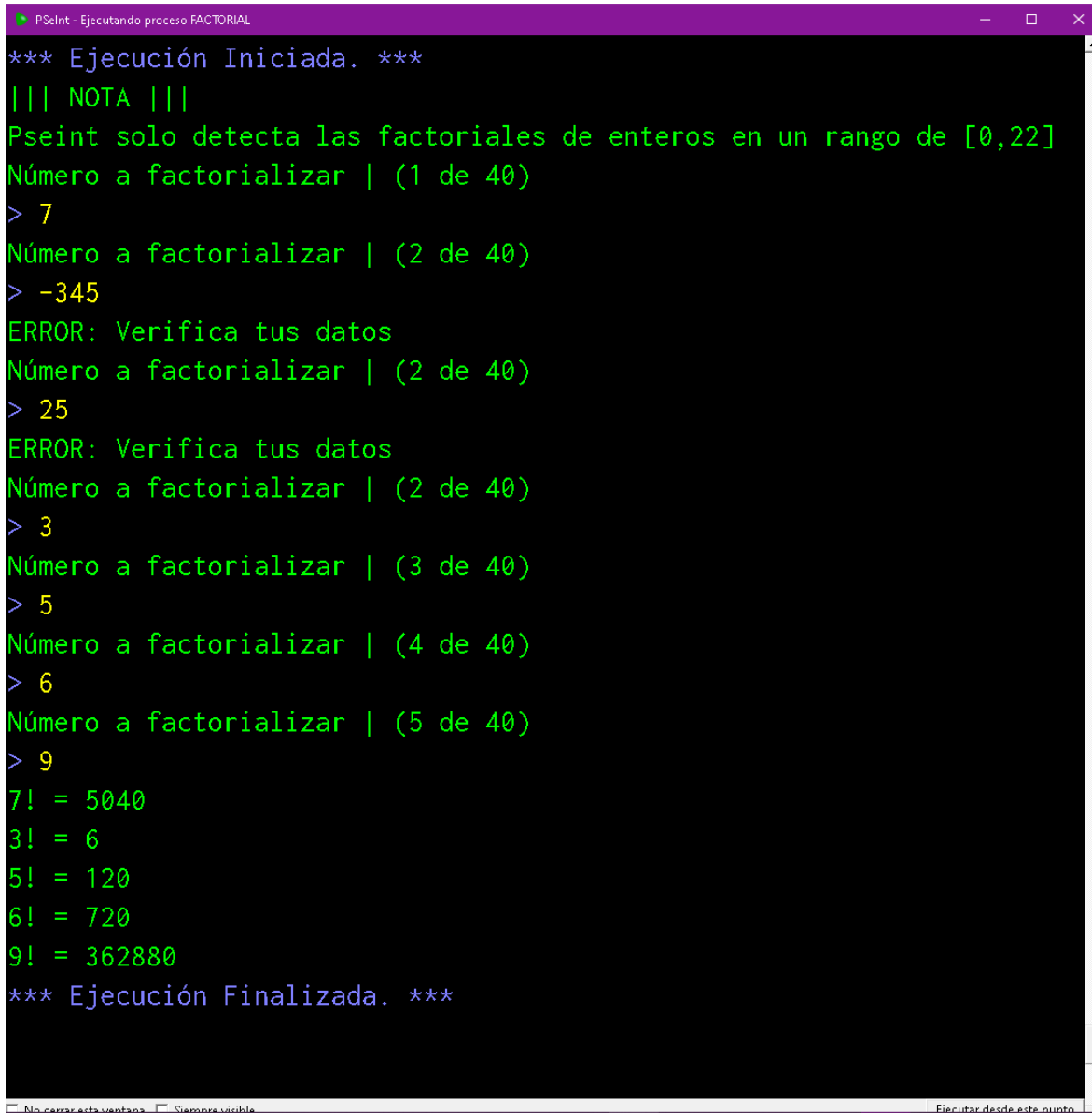
5. Hacer un pseudocódigo para calcular la factorial de cada uno de los números en un arreglo de 40 números enteros.

## PSEUDOCÓDIGO

```
1  Algoritmo factorial
2      Definir n,X,i Como Entero
3      //Pseint no detecta factoriales mayores a 15 como enteros
4      Definir f Como Real
5      //Asignación de dimensiones
6      Dimension n[40]
7      Dimension f[40]
8      //Advertencia al usuario sobre las limitaciones de Pseint
9      Escribir '||| NOTA |||'
10     Escribir 'Pseint solo detecta las factoriales de enteros en un
11     rango de [0,22]'
12     //Ingreso de valores
13     Para i←1 Hasta 40 Con Paso 1 Hacer
14         Repetir
15             Escribir 'Número a factorializar | (' ,i, ' de 40)'
16             Leer n[i]
17             Si n[i] < 0 o n[i] > 22 Entonces
18                 Escribir "ERROR: Verifica tus datos "
19             FinSi
20             Hasta Que n[i] ≥ 0 y n[i] ≤ 22
21             //Factorialización
22             Si n[i] = 0 Entonces
23                 f[i] ← 1
24             SiNo
25                 X ← n[i]
26                 f[i] ← 1
27                 Mientras X > 1 Hacer
28                     f[i] ← f[i] * X
29                     X ← X - 1
30                 Fin Mientras
31             FinSi
32         Fin Para
33         //Impresión de la factorial de cada valor ingresado
34         Escribir n[i], '! = ',f[i]
35     Fin Para
36 FinAlgoritmo
```



**EJECUCIÓN:** Factoriales de 5 números del arreglo.

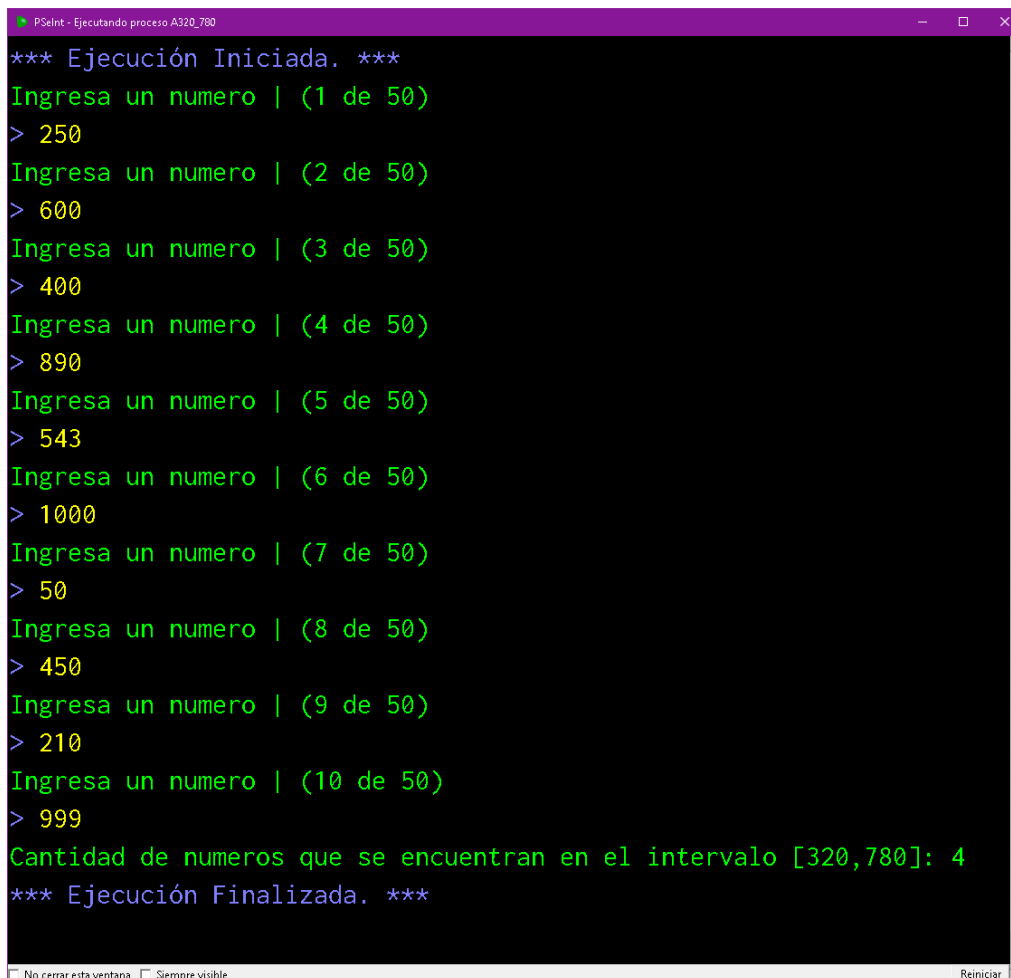


```
*** Ejecución Iniciada. ***  
||| NOTA |||  
Pseint solo detecta las factoriales de enteros en un rango de [0,22]  
Número a factorializar | (1 de 40)  
> 7  
Número a factorializar | (2 de 40)  
> -345  
ERROR: Verifica tus datos  
Número a factorializar | (2 de 40)  
> 25  
ERROR: Verifica tus datos  
Número a factorializar | (2 de 40)  
> 3  
Número a factorializar | (3 de 40)  
> 5  
Número a factorializar | (4 de 40)  
> 6  
Número a factorializar | (5 de 40)  
> 9  
7! = 5040  
3! = 6  
5! = 120  
6! = 720  
9! = 362880  
*** Ejecución Finalizada. ***
```

6. Hacer un pseudocódigo para contar cuantos, de los números almacenados en un arreglo de 50 números enteros, se encuentran entre 320 y 780.

```
1  Algoritmo A320_780
2      Definir n,x,i Como Entero
3      Dimension n[50]
4      x ← 0
5      Para i ← 1 Hasta 50 Con Paso 1 Hacer
6          //Ingreso de valores
7          Escribir 'Ingresa un numero | (' ,i, ' de 50) '
8          Leer n[i]
9          //Clasificador
10         Si n[i] ≥ 320 y n[i] ≤ 780 Entonces
11             x ← x + 1
12         FinSi
13     FinPara
14     Escribir 'Cantidad de números que se encuentran en el intervalo
15     [320,780]: ' ,x
16 FinAlgoritmo
```

**EJECUCIÓN:** Prueba con 10 valores.



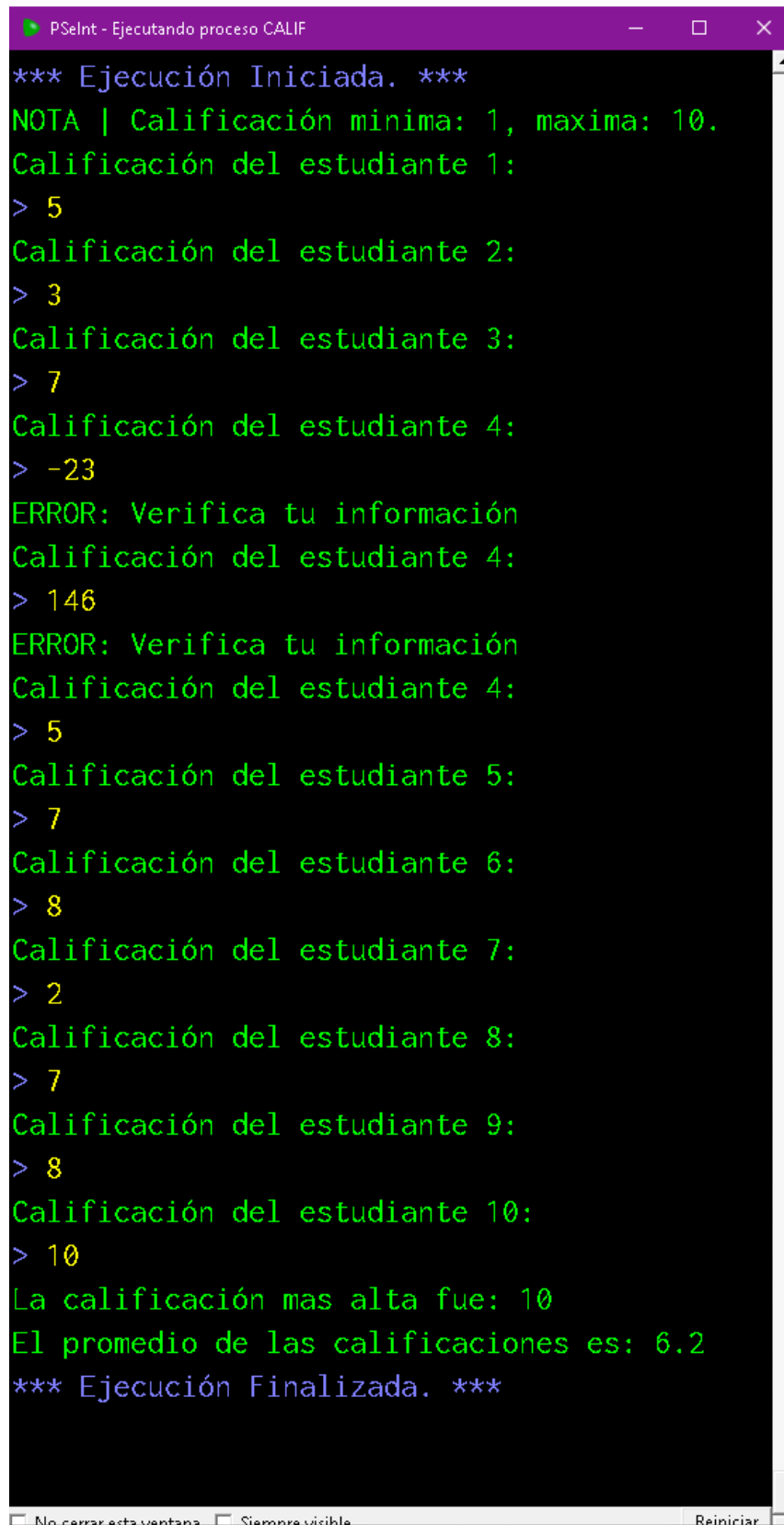
```
*** Ejecución Iniciada. ***
Ingresa un numero | (1 de 50)
> 250
Ingresa un numero | (2 de 50)
> 600
Ingresa un numero | (3 de 50)
> 400
Ingresa un numero | (4 de 50)
> 890
Ingresa un numero | (5 de 50)
> 543
Ingresa un numero | (6 de 50)
> 1000
Ingresa un numero | (7 de 50)
> 50
Ingresa un numero | (8 de 50)
> 450
Ingresa un numero | (9 de 50)
> 210
Ingresa un numero | (10 de 50)
> 999
Cantidad de numeros que se encuentran en el intervalo [320,780]: 4
*** Ejecución Finalizada. ***
```

7. Hacer un pseudocódigo para leer calificaciones en un arreglo de 200, posteriormente el algoritmo debe encontrar la calificación más alta e imprimirla, y finalmente imprimir el promedio de todo el arreglo.

## PSEUDOCÓDIGO

```
1  Algoritmo calif
2      Definir c,pc,rec Como Real
3      Definir i Como Entero
4      Dimension c[200]
5      rec ← 0
6      pc ← 0
7      //Ingreso de calificaciones
8      Escribir 'NOTA | Calificación mínima: 1, máxima: 10.'
9      Para i ← 1 Hasta 200 Con Paso 1 Hacer
10         //Calificación por estudiante
11         Repetir
12             Escribir 'Calificación del estudiante ',i,':'
13             Leer c[i]
14             Si c[i] < 0 o c[i] > 10 Entonces
15                 Escribir 'ERROR: Verifica tu información'
16             FinSi
17         Hasta Que c[i] > 0 y c[i] ≤ 10
18         //Suma para el promedio
19         pc ← pc + c[i]
20         //Encontrar la mayor calificación
21         Si c[i] > rec Entonces
22             rec ← c[i]
23         FinSi
24     FinPara
25     //Promedio de calificaciones
26     pc ← pc/200
27     Escribir 'La calificación más alta fue: ',rec
28     Escribir 'El promedio de las calificaciones es: ',pc
29 FinAlgoritmo
```

**EJECUCIÓN:** Ingreso de calificaciones de 10 estudiantes.



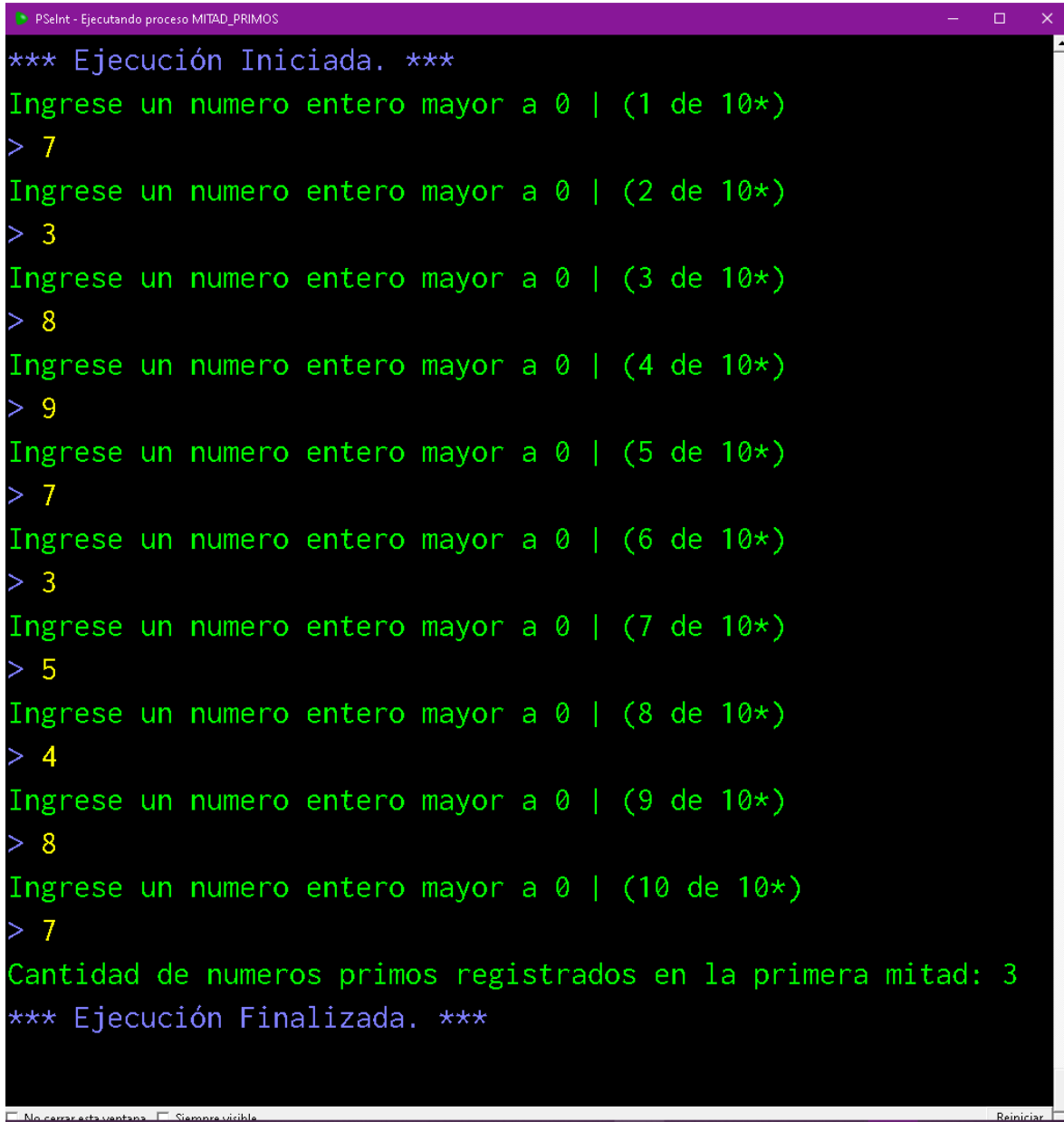
```
PSelnt - Ejecutando proceso CALIF
*** Ejecución Iniciada. ***
NOTA | Calificación mínima: 1, máxima: 10.
Calificación del estudiante 1:
> 5
Calificación del estudiante 2:
> 3
Calificación del estudiante 3:
> 7
Calificación del estudiante 4:
> -23
ERROR: Verifica tu información
Calificación del estudiante 4:
> 146
ERROR: Verifica tu información
Calificación del estudiante 4:
> 5
Calificación del estudiante 5:
> 7
Calificación del estudiante 6:
> 8
Calificación del estudiante 7:
> 2
Calificación del estudiante 8:
> 7
Calificación del estudiante 9:
> 8
Calificación del estudiante 10:
> 10
La calificación mas alta fue: 10
El promedio de las calificaciones es: 6.2
*** Ejecución Finalizada. ***
☐ No cerrar esta ventana ☐ Siempre visible 
```

8. Hacer un pseudocódigo para leer y almacenar 40 números enteros en un arreglo, posteriormente contar cuantos números primos hay en la primera mitad del arreglo.

## PSEUDOCÓDIGO

```
1  Algoritmo mitad_primos
2      Definir nprim,tprim,n,i,j Como Entero
3      Dimension n[40]
4      tprim ← 0
5      Para i ← 1 Hasta 40 Con Paso 1 Hacer
6          nprim ← 0
7          //Registro de números
8          Repetir
9              Escribir 'Ingrese un numero entero mayor a 0 | (' ,i, ' de
10             40) '
11             Leer n[i]
12             Hasta Que n[i]>0
13             //Limitador de la mitad del arreglo
14             Si n[i] ≤ 20 Entonces
15                 // determinación de un numero primo
16                 Para j←1 Hasta n[i] Hacer
17                     Si n[i] MOD j=0 Entonces
18                         nprim ← nprim+1
19                     FinSi
20                 FinPara
21                 // contador de números primos
22                 Si nprim=2 Entonces
23                     tprim ← tprim+1
24                 FinSi
25             FinSi
26         FinPara
27         Escribir 'Cantidad de números primos registrados en la primera
mitad: ',tprim
FinAlgoritmo
```

**EJECUCIÓN:** Prueba con 10 valores.



```
*** Ejecución Iniciada. ***
Ingrese un numero entero mayor a 0 | (1 de 10*)
> 7
Ingrese un numero entero mayor a 0 | (2 de 10*)
> 3
Ingrese un numero entero mayor a 0 | (3 de 10*)
> 8
Ingrese un numero entero mayor a 0 | (4 de 10*)
> 9
Ingrese un numero entero mayor a 0 | (5 de 10*)
> 7
Ingrese un numero entero mayor a 0 | (6 de 10*)
> 3
Ingrese un numero entero mayor a 0 | (7 de 10*)
> 5
Ingrese un numero entero mayor a 0 | (8 de 10*)
> 4
Ingrese un numero entero mayor a 0 | (9 de 10*)
> 8
Ingrese un numero entero mayor a 0 | (10 de 10*)
> 7
Cantidad de numeros primos registrados en la primera mitad: 3
*** Ejecución Finalizada. ***
```

☐ No cerrar ventana ☐ Siempre visible Reiniciar