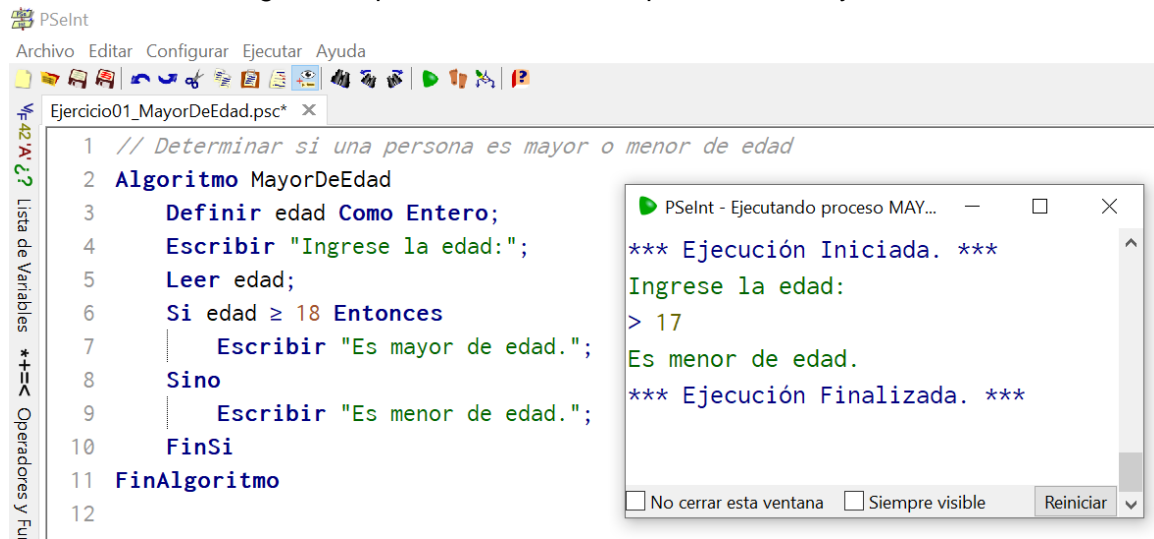


## Realizar los siguientes ejercicios de estructuras de controles

### Estructura SI - SINO

1. Escriba un algoritmo que determine si una persona es mayor o menor de edad.



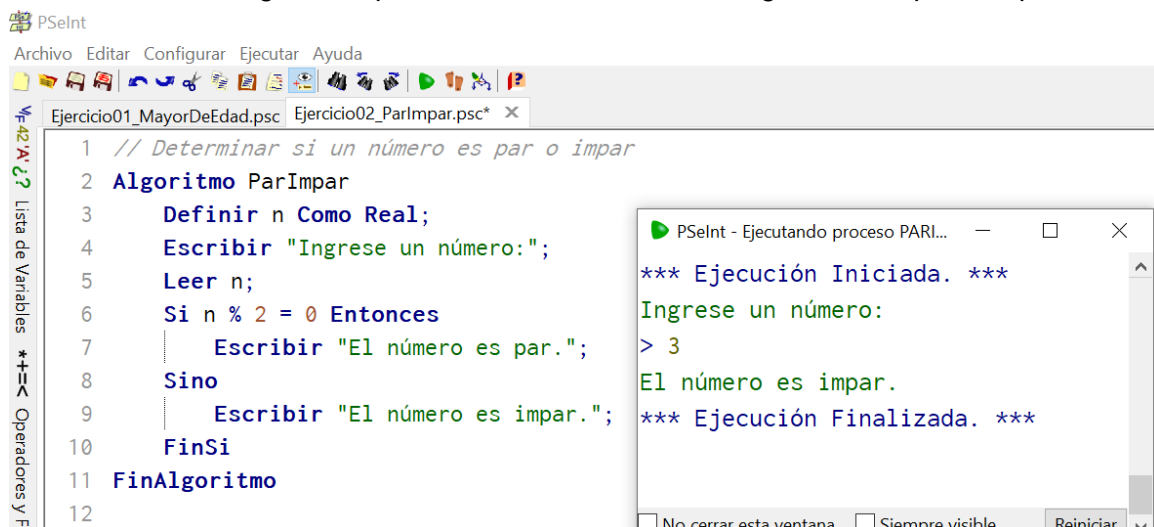
The screenshot shows the PSeInt IDE with the file 'Ejercicio01\_MayorDeEdad.psc' open. The code is as follows:

```
1 // Determinar si una persona es mayor o menor de edad
2 Algoritmo MayorDeEdad
3     Definir edad Como Entero;
4     Escribir "Ingrese la edad:";
5     Leer edad;
6     Si edad ≥ 18 Entonces
7         Escribir "Es mayor de edad.";
8     Sino
9         Escribir "Es menor de edad.";
10    FinSi
11 FinAlgoritmo
12
```

The execution window shows the following output:

```
*** Ejecución Iniciada. ***
Ingrese la edad:
> 17
Es menor de edad.
*** Ejecución Finalizada. ***
```

2. Escriba un algoritmo que determine si un número ingresado es par o impar.



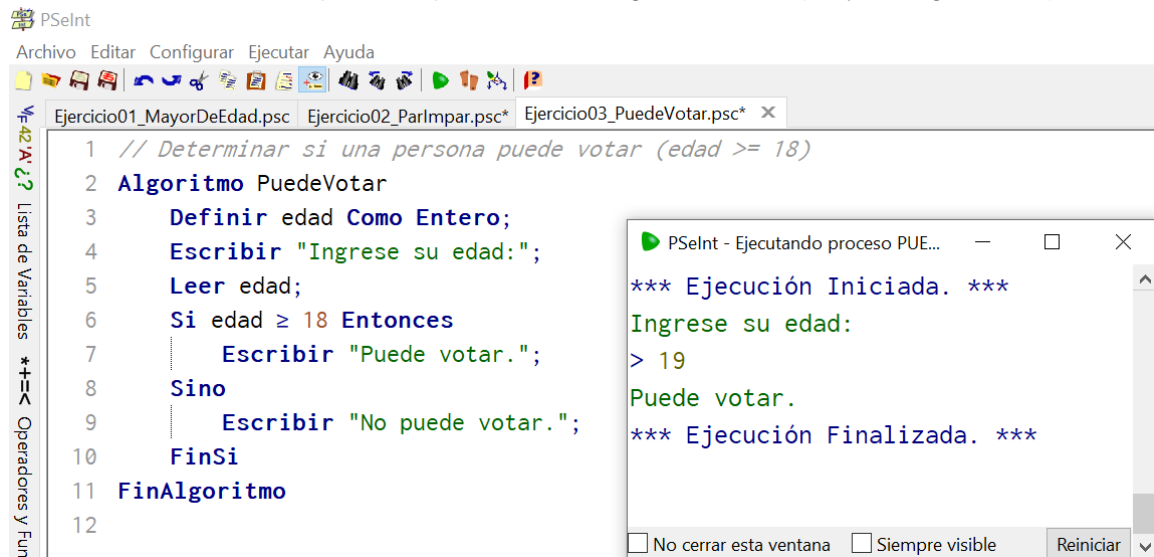
The screenshot shows the PSeInt IDE with the file 'Ejercicio02\_ParImpar.psc' open. The code is as follows:

```
1 // Determinar si un número es par o impar
2 Algoritmo ParImpar
3     Definir n Como Real;
4     Escribir "Ingrese un número:";
5     Leer n;
6     Si n % 2 = 0 Entonces
7         Escribir "El número es par.";
8     Sino
9         Escribir "El número es impar.";
10    FinSi
11 FinAlgoritmo
12
```

The execution window shows the following output:

```
*** Ejecución Iniciada. ***
Ingrese un número:
> 3
El número es impar.
*** Ejecución Finalizada. ***
```

3. 3. Determine si una persona puede votar según su edad (mayor o igual a 18).



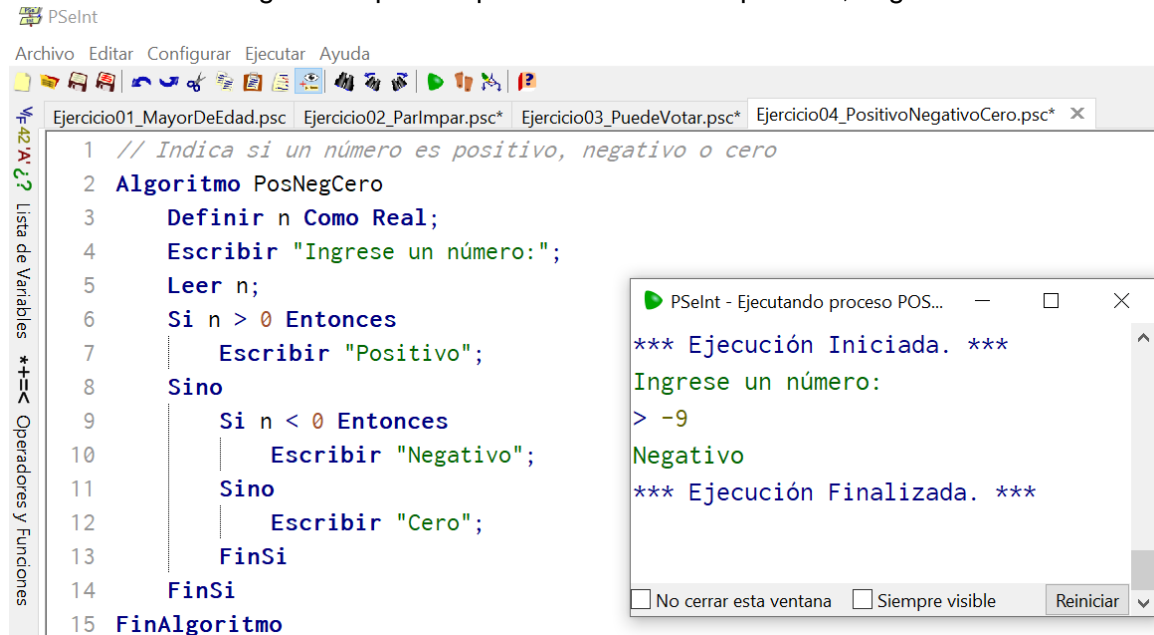
The screenshot shows the PSeInt IDE with the file 'Ejercicio03\_PuedeVotar.psc' open. The code is as follows:

```
1 // Determinar si una persona puede votar (edad >= 18)
2 Algoritmo PuedeVotar
3     Definir edad Como Entero;
4     Escribir "Ingrese su edad:";
5     Leer edad;
6     Si edad >= 18 Entonces
7         Escribir "Puede votar.";
8     Sino
9         Escribir "No puede votar.";
10    FinSi
11 FinAlgoritmo
12
```

The execution window shows the following output:

```
*** Ejecución Iniciada. ***
Ingrese su edad:
> 19
Puede votar.
*** Ejecución Finalizada. ***
```

4. 4. Escriba un algoritmo que indique si un número es positivo, negativo o cero.



The screenshot shows the PSeInt IDE with the file 'Ejercicio04\_PositivoNegativoCero.psc' open. The code is as follows:

```
1 // Indica si un número es positivo, negativo o cero
2 Algoritmo PosNegCero
3     Definir n Como Real;
4     Escribir "Ingrese un número:";
5     Leer n;
6     Si n > 0 Entonces
7         Escribir "Positivo";
8     Sino
9         Si n < 0 Entonces
10            Escribir "Negativo";
11        Sino
12            Escribir "Cero";
13        FinSi
14    FinSi
15 FinAlgoritmo
```

The execution window shows the following output:

```
*** Ejecución Iniciada. ***
Ingrese un número:
> -9
Negativo
*** Ejecución Finalizada. ***
```

5. 5. Determine si un estudiante aprueba o reprueba según su nota (mayor o igual a 70).

The screenshot shows the PSeInt IDE with a project named 'Ejercicio01\_MayorDeEdad.psc'. The code is as follows:

```
1 // Determinar si un estudiante aprueba (nota >= 70)
2 Algoritmo AprobadoReprobado
3     Definir nota Como Real;
4     Escribir "Ingrese la nota (0-100):";
5     Leer nota;
6     Si nota >= 70 Entonces
7         Escribir "Aprobado";
8     Sino
9         Escribir "Reprobado";
10    FinSi
11 FinAlgoritmo
12
```

The execution window shows the following output:

```
*** Ejecución Iniciada. ***
Ingrese la nota (0-100):
> 71
Aprobado
*** Ejecución Finalizada. ***
```

## Estructura SEGUN

6. 1. Muestre el nombre del mes correspondiente a un número del 1 al 12.

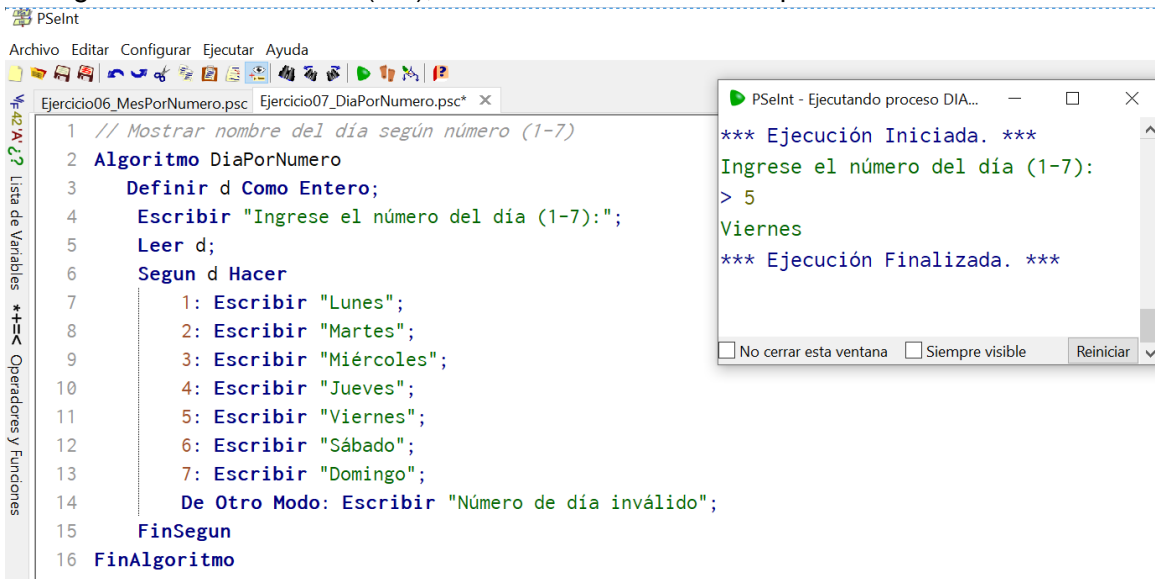
The screenshot shows the PSeInt IDE with a project named 'Ejercicio05\_AprobadoReprobado'. The code is as follows:

```
1 // Mostrar nombre del mes según número (1-12)
2 Algoritmo MesPorNumero
3     Definir m Como entero;
4     Escribir "Ingrese el número del mes (1-12):";
5     Leer m;
6     Segun m Hacer
7         1: Escribir "Enero";
8         2: Escribir "Febrero";
9         3: Escribir "Marzo";
10        4: Escribir "Abril";
11        5: Escribir "Mayo";
12        6: Escribir "Junio";
13        7: Escribir "Julio";
14        8: Escribir "Agosto";
15        9: Escribir "Septiembre";
16        10: Escribir "Octubre";
17        11: Escribir "Noviembre";
18        12: Escribir "Diciembre";
19        De Otro Modo: Escribir "Número de mes inválido";
20    FinSegun
21 FinAlgoritmo
```

The execution window shows the following output:

```
*** Ejecución Iniciada. ***
Ingrese el número del mes (1-12):
> 10
Octubre
*** Ejecución Finalizada. ***
```

7. 2. Según el número del día (1-7), mostrar su nombre correspondiente.



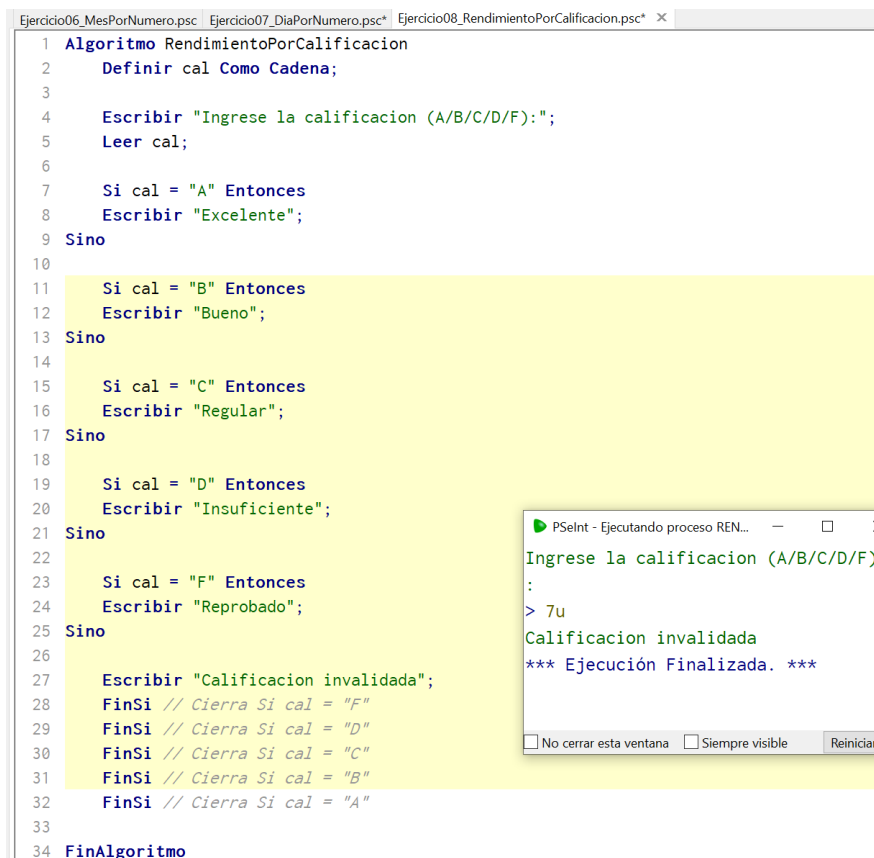
```
1 // Mostrar nombre del día según número (1-7)
2 Algoritmo DiaPorNumero
3     Definir d Como Entero;
4     Escribir "Ingrese el número del día (1-7)";
5     Leer d;
6     Segun d Hacer
7         1: Escribir "Lunes";
8         2: Escribir "Martes";
9         3: Escribir "Miércoles";
10        4: Escribir "Jueves";
11        5: Escribir "Viernes";
12        6: Escribir "Sábado";
13        7: Escribir "Domingo";
14        De Otro Modo: Escribir "Número de día inválido";
15    FinSegun
16 FinAlgoritmo
```

PSeInt - Ejecutando proceso DIA...

```
*** Ejecución Iniciada. ***
Ingrese el número del día (1-7):
> 5
Viernes
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☐ Siempre visible

8. 3. Según la calificación (A, B, C, D, F), mostrar el mensaje de rendimiento del estudiante.



```
1 Algoritmo RendimientoPorCalificacion
2     Definir cal Como Cadena;
3
4     Escribir "Ingrese la calificacion (A/B/C/D/F)";
5     Leer cal;
6
7     Si cal = "A" Entonces
8         Escribir "Excelente";
9     Sino
10
11         Si cal = "B" Entonces
12             Escribir "Bueno";
13         Sino
14
15             Si cal = "C" Entonces
16                 Escribir "Regular";
17             Sino
18
19                 Si cal = "D" Entonces
20                     Escribir "Insuficiente";
21             Sino
22
23                 Si cal = "F" Entonces
24                     Escribir "Reprobado";
25             Sino
26
27                 Escribir "Calificacion invalidada";
28             FinSi // Cierra Si cal = "F"
29             FinSi // Cierra Si cal = "D"
30             FinSi // Cierra Si cal = "C"
31             FinSi // Cierra Si cal = "B"
32             FinSi // Cierra Si cal = "A"
33
34 FinAlgoritmo
```

PSeInt - Ejecutando proceso REN...

```
Ingrese la calificacion (A/B/C/D/F)
:
> 7u
Calificacion invalidada
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☐ Siempre visible

9. 4. Según el tipo de operación (+, -, \*, /), mostrar el resultado entre dos números.

Ejercicio06\_MesPorNumero.psc | Ejercicio07\_DiaPorNumero.psc\* | Ejercicio08\_RendimientoPorCalificacio

```
1 // Muestra un mensaje distinto según se ingrese 1,2 o 3
2 Algoritmo MensajePersonalizado
3     Definir n Como real;
4     Escribir "Ingrese un número (1-3):";
5     Leer n;
6     Segun n Hacer
7         1: Escribir "Elegiste la opción 1 - Bienvenido";
8         2: Escribir "Elegiste la opción 2 - Buen trabajo";
9         3: Escribir "Elegiste la opción 3 - Excelente";
10        De Otro Modo: Escribir "Opción no válida";
11    FinSegun
12 FinAlgoritmo
13
```

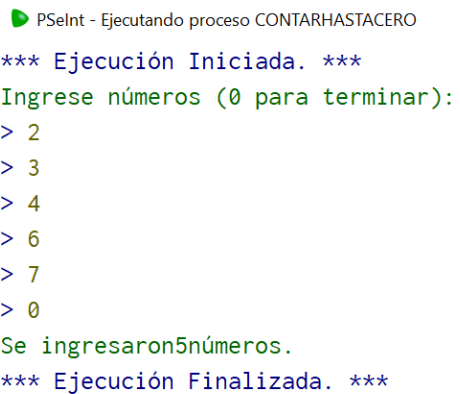
10. 5. Según el número ingresado (1-3), mostrar un mensaje personalizado diferente para cada uno.

```
1 // Muestra un mensaje distinto según se ingrese 1,2 o 3
2 Algoritmo MensajePersonalizado
3     Definir n Como real;
4     Escribir "Ingrese un número (1-3):";
5     Leer n;
6     Segun n Hacer
7         1: Escribir "Elegiste la opción 1 - Bienvenido";
8         2: Escribir "Elegiste la opción 2 - Buen trabajo";
9         3: Escribir "Elegiste la opción 3 - Excelente";
10        De Otro Modo: Escribir "Opción no válida";
11    FinSegun
12 FinAlgoritmo
13
```

## Estructura MIENTRAS

11. 1. Leer números hasta que el usuario ingrese 0 y mostrar cuántos números ingresó.

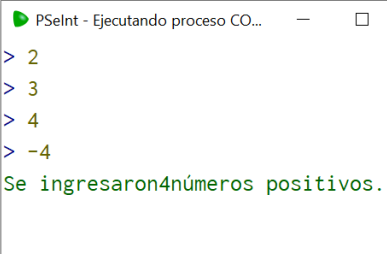
```
1 // Leer números hasta que el usuario ingrese 0 y contar cuántos ingresó
2 Algoritmo ContarHastaCero
3   Definir contador,n Como Real;
4   contador ← 0;
5   Escribir "Ingrese números (0 para terminar):";
6   Leer n;
7   Mientras n ≠ 0 Hacer
8     contador ← contador + 1;
9     Leer n;
10  FinMientras
11  Escribir "Se ingresaron", contador, "números.";
12 FinAlgoritmo
13
```



The screenshot shows the execution of the 'ContarHastaCero' algorithm in PSeInt. The window title is 'PSeInt - Ejecutando proceso CONTARHASTACERO'. The output shows the prompt 'Ingrese números (0 para terminar):' followed by five inputs: 2, 3, 4, 6, and 7. After the final input, the user enters 0 to terminate. The program then outputs 'Se ingresaron5números.' and ends with '\*\*\* Ejecución Finalizada. \*\*\*'.

12. 2. Contar cuántos números positivos se ingresan antes de ingresar un número negativo.

```
1 // Cuenta cuántos números positivos se ingresan antes de ingresar un número negativo
2 Algoritmo ContarPositivosHastaNegativo
3   Definir contador,n Como Real;
4   contador ← 0;
5   Escribir "Ingrese números (ingrese un número negativo para terminar):";
6   Leer n;
7   Mientras n ≥ 0 Hacer
8     Si n > 0 Entonces
9       contador ← contador + 1;
10    FinSi
11    Leer n;
12  FinMientras
13  Escribir "Se ingresaron", contador, "números positivos.";
14 FinAlgoritmo
15
```



The screenshot shows the execution of the 'ContarPositivosHastaNegativo' algorithm in PSeInt. The window title is 'PSeInt - Ejecutando proceso CO...'. The output shows the prompt 'Ingrese números (ingrese un número negativo para terminar):' followed by four inputs: 2, 3, 4, and -4. After the final input, the user enters a negative number to terminate. The program then outputs 'Se ingresaron4números positivos.' and ends with '\*\*\* Ejecución Finalizada. \*\*\*'.

13. 3. Leer edades hasta que se ingrese una edad fuera del rango 0-120.

```

1 // Lee edades hasta que se ingrese una edad fuera del rango 0-120
2 Algoritmo LeerEdadesRango
3     Definir edad Como real;
4     Escribir "Ingrese edades (ingrese una edad fuera de 0-120 para terminar):";
5     Leer edad;
6     Mientras edad ≥ 0 Y edad ≤ 120 Hacer
7         Escribir "Edad registrada:", edad;
8         Leer edad;
9     FinMientras
10    Escribir "Se ingresó una edad fuera del rango. Fin del programa.";
11 FinAlgoritmo
12

```

PSelnt - Ejecutando proceso LEEREDADESRANGO

```

Edad registrada:10
> 23
Edad registrada:23
> 45
Edad registrada:45
> 120
Edad registrada:120
> 121
Se ingresó una edad fuera del rango. Fin del programa
*** Ejecución Finalizada. ***

```

#### 14. 4. Mostrar los primeros 10 números naturales usando un ciclo mientras.

```

1 // Muestra los primeros 10 números naturales usando ciclo Mientras
2 Algoritmo Primeros10Naturales
3     definir i Como entero;
4     i ← 1;
5     Mientras i ≤ 10 Hacer
6         Escribir i;
7         i ← i + 1;
8     FinMientras
9 FinAlgoritmo
10

```

PSelnt - Ejecutando proceso PRIMEROS10NATURA

```

*** Ejecución Iniciada. ***
1
2
3
4
5
6
7
8
9
10
*** Ejecución Finalizada. ***

```

### 15. 5. Leer números hasta que su suma sea mayor a 100.

```
1 // Lee números y acumula su suma hasta que sea mayor a 100
2 Algoritmo SumarHastaMayor100
3     Definir n, suma Como Real;
4
5     suma ← 0;
6     Escribir "Ingrese números (se sumarán hasta superar 100)";
7
8
9     Mientras suma ≤ 100 Hacer
10         Leer n;
11         suma ← suma + n;
12     FinMientras
13
14     Escribir "La suma total es:", suma;
15 FinAlgoritmo
```

PSelnt - Ejecutando proceso SUMARHASTAMAYOR100

\*\*\* Ejecución Iniciada. \*\*\*

Ingrese números (se sumarán hasta superar 100)

> 60

> 50

La suma total es:110

\*\*\* Ejecución Finalizada. \*\*\*

## Estructura REPETIR - HASTA QUE

### 16. 1. Pedir números y sumarlos hasta que el usuario ingrese 0.

```
1 // Suma números usando Repetir - Hasta que se ingrese 0
2 Algoritmo SumarRepetirHastaCero
3     Definir suma, n Como Real;
4     suma ← 0;
5     Repetir
6         Escribir "Ingrese un número (0 para terminar):";
7         Leer n;
8         suma ← suma + n;
9     Hasta Que n = 0;
10     Escribir "La suma es:", suma;
11 FinAlgoritmo
12
```

PSelnt - Ejecutando proceso SUMARREPETIRHASTACERO

Ingrese un número (0 para terminar):

> 2

Ingrese un número (0 para terminar):

> 3

Ingrese un número (0 para terminar):

> 4

Ingrese un número (0 para terminar):

> 0

La suma es:9

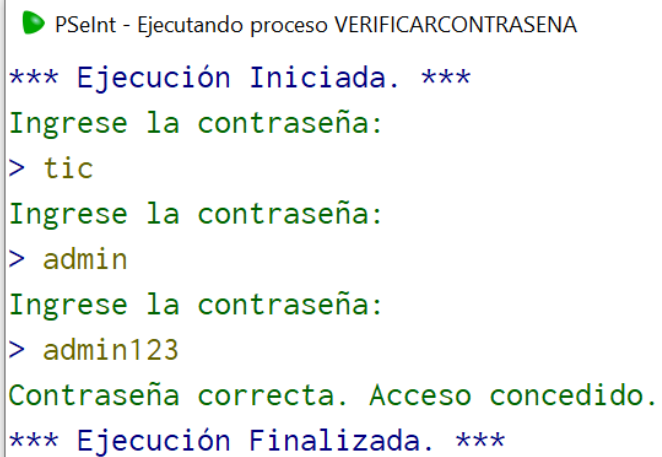
### 17. 2. Solicitar contraseñas hasta que el usuario ingrese la correcta ('admin123').



```

1 // Solicitar contraseñas hasta ingresar la correcta 'admin123'
2 Algoritmo VerificarContrasena
3     Definir pass Como Caracter;
4     Repetir
5         Escribir "Ingrese la contraseña:";
6         Leer pass;
7     Hasta Que pass = "admin123"
8     Escribir "Contraseña correcta. Acceso concedido.";
9 FinAlgoritmo
10

```



```

PSeInt - Ejecutando proceso VERIFICARCONTRASENA

*** Ejecución Iniciada. ***
Ingrese la contraseña:
> tic
Ingrese la contraseña:
> admin
Ingrese la contraseña:
> admin123
Contraseña correcta. Acceso concedido.
*** Ejecución Finalizada. ***

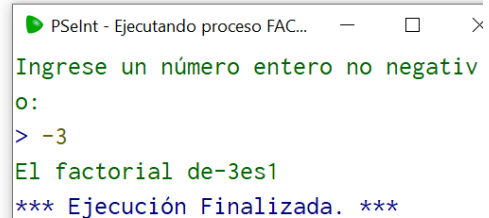
```

18. 3. Calcular el factorial de un número usando un ciclo repetir.

```

// Calcula el factorial de un número usando Repetir
Algoritmo FactorialRepetir
    Definir factorial, n, i Como real;
    Escribir "Ingrese un número entero no negativo:";
    Leer n;
    factorial ← 1;
    i ← 1;
    Repetir
        factorial ← factorial * i;
        i ← i + 1;
    Hasta Que i > n
    Escribir "El factorial de", n, "es", factorial;
FinAlgoritmo

```



```

PSeInt - Ejecutando proceso FAC...

Ingrese un número entero no negativo:
> -3
El factorial de-3es1
*** Ejecución Finalizada. ***

```

19. 4. Leer números hasta que se ingrese un número negativo, mostrando la cantidad de positivos.

```

1 // Lee números hasta que se ingrese un número negativo y muestra la cantidad de positivos
2 Algoritmo ContarPositivosRepetir
3     Definir n, positivos Como real;
4     positivos ← 0;
5     Repetir
6         Escribir "Ingrese un número (número negativo para terminar):";
7         Leer n;
8         Si n > 0 Entonces;
9             positivos ← positivos + 1;
10        FinSi
11    Hasta Que n < 0
12    Escribir "Se ingresaron", positivos, "números positivos.";
13 FinAlgoritmo
14

```

```

PSeInt - Ejecutando proceso CONTARPOSITIVOSREPETIR
*** Ejecución Iniciada. ***
Ingrese un número (número negativo para terminar):
> 2
Ingrese un número (número negativo para terminar):
> 3
Ingrese un número (número negativo para terminar):
> -3
Se ingresaron2números positivos.
*** Ejecución Finalizada. ***

```

20. 5. Ingresar notas hasta que el usuario ingrese -1, y luego mostrar el promedio.

```

1 // Ingresa notas hasta que el usuario ingrese -1 y luego calcula el promedio
2 Algoritmo PromedioNotasHastaMenos1
3     Definir suma, contador, nota Como Real;
4     suma ← 0;
5     contador ← 0;
6     Repetir
7         Escribir "Ingrese una nota (-1 para terminar):";
8         Leer nota;
9         Si nota ≠ -1 Entonces;
10            suma ← suma + nota;
11            contador ← contador + 1;
12        FinSi
13    Hasta Que nota = -1;
14    Si contador > 0 Entonces
15        Escribir "Promedio:", suma / contador;
16    Sino
17        Escribir "No se ingresaron notas.";
18    FinSi
19 FinAlgoritmo
20

```

```

PSeInt - Ejecutando proceso PROMEDIONOTASHASTAMEN
*** Ejecución Iniciada. ***
Ingrese una nota (-1 para terminar):
> 80
Ingrese una nota (-1 para terminar):
> 90
Ingrese una nota (-1 para terminar):
> 92
Ingrese una nota (-1 para terminar):
> -1
Promedio:87.3333333333
*** Ejecución Finalizada. ***

```

## Estructura PARA / Acumuladores / Contadores / Indicadores

21. 1. Calcular el promedio de 5 notas ingresadas.

```
1 // Calcula el promedio de 5 notas usando estructura PARA
2 Algoritmo Promedio5Notas
3     Definir suma, i, nota Como Real;
4     suma ← 0;
5
6     // CORRECCIÓN: Se elimina 'Paso 1'
7     Para i ← 1 Hasta 5 Hacer
8         Escribir "Ingrese la nota", i, ":";
9         Leer nota;
10        suma ← suma + nota;
11    FinPara
12
13    Escribir "El promedio es:", suma / 5;
14 FinAlgoritmo
```

```
PSelnt - Ejecutando proceso PROMEDIO5NOT,
*** Ejecución Iniciada. ***
Ingrese la nota1:
> 80
Ingrese la nota2:
> 77
Ingrese la nota3:
> 90
Ingrese la nota4:
> 99
Ingrese la nota5:
> 100
El promedio es:89.2
*** Ejecución Finalizada. ***
```

22. 2. Mostrar la tabla de multiplicar de un número ingresado (del 1 al 10).

```
// Muestra la tabla de multiplicar de un número del 1 al 10
Algoritmo TablaMultiplicar
    Definir n, i Como real;

    Escribir "Ingrese un número:";
    Leer n;

    Para i ← 1 Hasta 10 Hacer
        Escribir n, "x", i, "=", n * i;
    FinPara

FinAlgoritmo
```

```
PSelnt - Ejecutando proceso TABLAMULTIPLIC
Ingrese un número:
> 1
1x1=1
1x2=2
1x3=3
1x4=4
1x5=5
1x6=6
1x7=7
1x8=8
1x9=9
1x10=10
*** Ejecución Finalizada. **
```

23. 3. Sumar los números del 1 al 50 usando un acumulador.

```
// Suma los números del 1 al 50
Algoritmo Sumar1a50
    Definir i, suma Como real;

    suma ← 0;
    Para i ← 1 Hasta 50 Hacer
        suma ← suma + i;
    FinPara

    Escribir "La suma de 1 a 50 es:", suma;
FinAlgoritmo
```

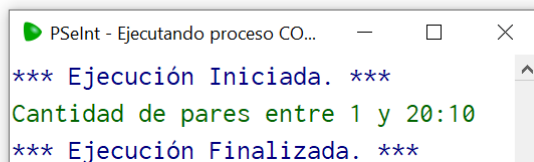
```
▶ PSeInt - Ejecutando proceso SU...
*** Ejecución Iniciada. ***
La suma de 1 a 50 es:1275
*** Ejecución Finalizada. ***
```

24. 4. Contar cuántos números pares hay entre 1 y 20.

```

1 // Cuenta los números pares entre 1 y 20
2 Algoritmo ContarPares1a20
3     definir i, contador como real;
4
5     contador ← 0;
6
7     Para i ← 1 Hasta 20 Hacer
8         Si i % 2 = 0 Entonces // Si el residuo de dividir i entre 2 es cero, es par
9             contador ← contador + 1;
10        FinSi
11    FinPara
12
13    Escribir "Cantidad de pares entre 1 y 20:", contador;
14 FinAlgoritmo

```



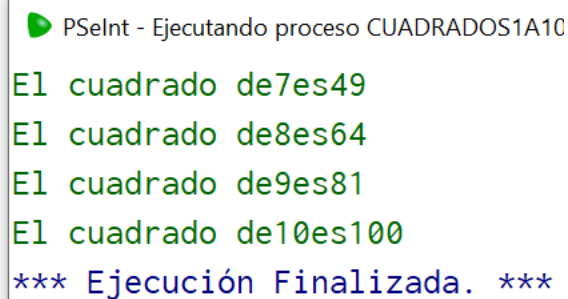
PSeInt - Ejecutando proceso CO...  
 \*\*\* Ejecución Iniciada. \*\*\*  
 Cantidad de pares entre 1 y 20:10  
 \*\*\* Ejecución Finalizada. \*\*\*

25. 5. Mostrar los cuadrados de los números del 1 al 10.

```

1 // Muestra los cuadrados de los números del 1 al 10
2 Algoritmo Cuadrados1a10
3     Definir i Como Real;
4
5     Para i ← 1 Hasta 10 Hacer
6         Escribir "El cuadrado de", i, "es", i * i;
7     FinPara
8
9 FinAlgoritmo

```



PSeInt - Ejecutando proceso CUADRADOS1A10  
 El cuadrado de7es49  
 El cuadrado de8es64  
 El cuadrado de9es81  
 El cuadrado de10es100  
 \*\*\* Ejecución Finalizada. \*\*\*