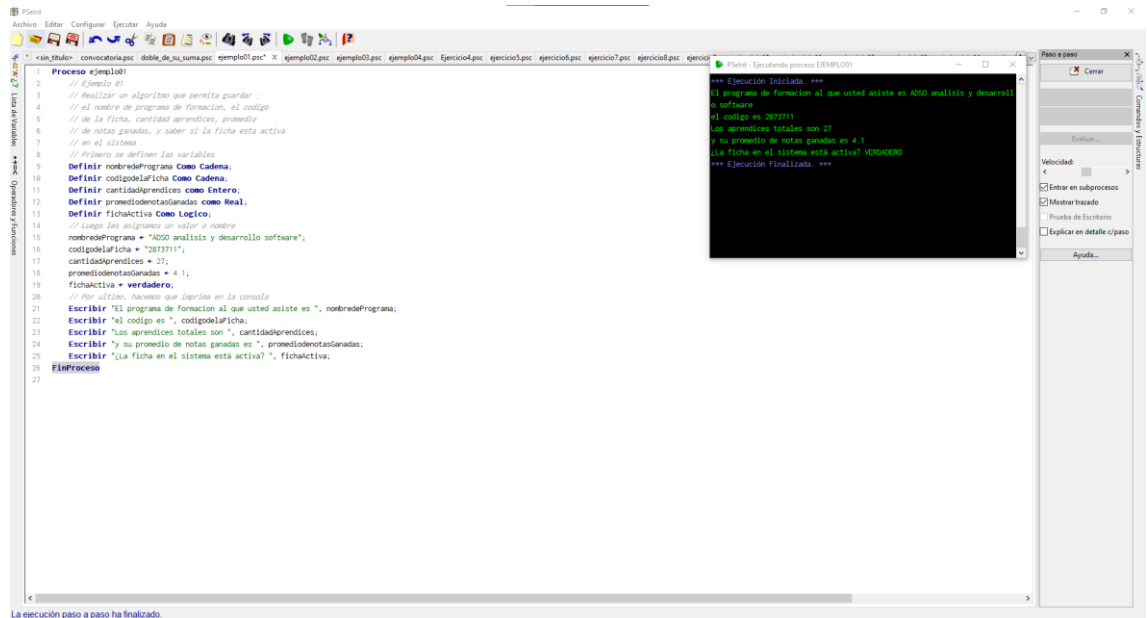


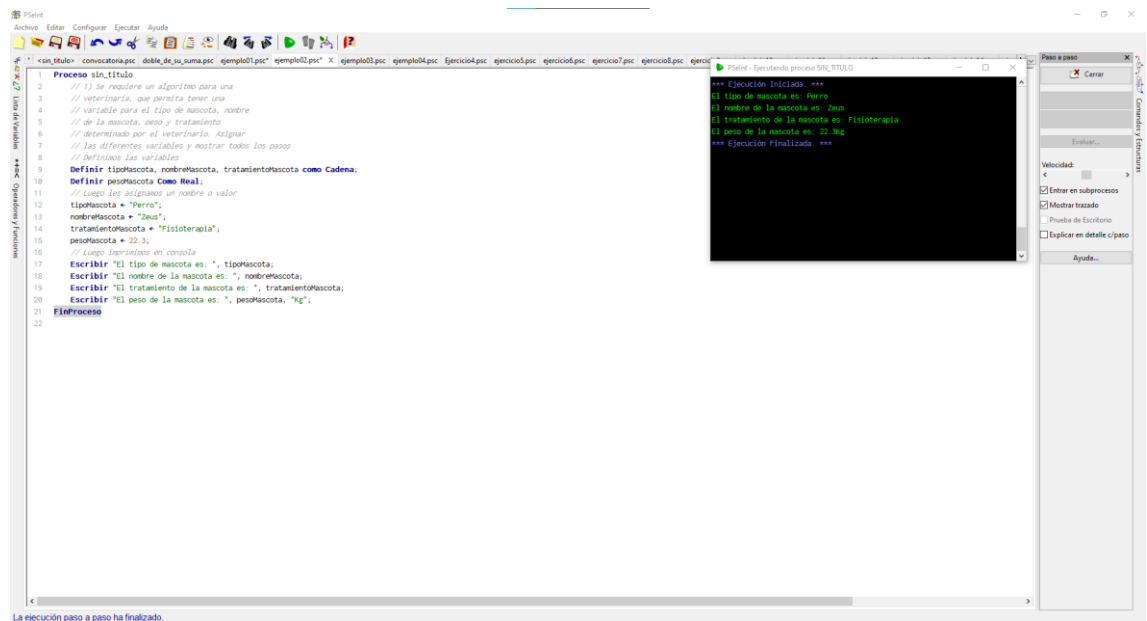
Evidencias algoritmos guía 1 – introducción a la Programación

Ejemplos

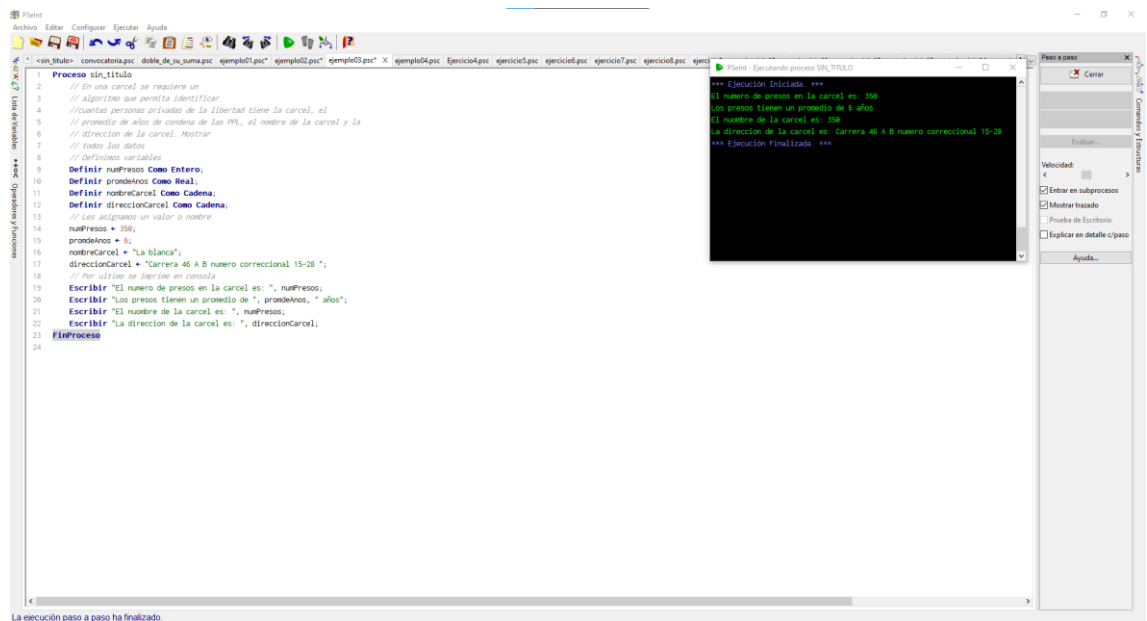
- 1) Se requiere un algoritmo que permita guardar: nombre, programa de formación, cantidad de aprendices, código de la ficha, promedio de notas, saber si la fecha esta activa



- 2) Se requiere un algoritmo para una veterinaria, que permita tener una variable para el tipo de mascota, nombre de la mascota, peso y tratamiento determinado por el veterinario. Asignarlas diferentes variables y mostrar todos los datos



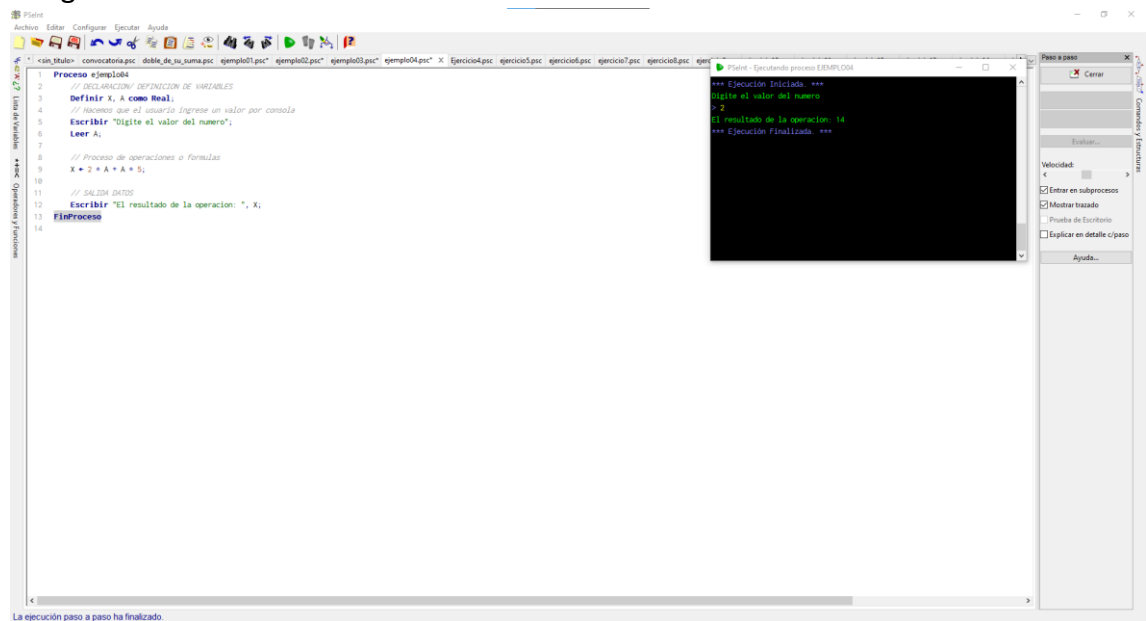
- 3) En una cárcel se requiere un algoritmo que permita identificar cuantas personas privadas de la libertad tiene la cárcel, el promedio de años de condena de las PPL, el nombre de la cárcel la dirección de la cárcel. Mostrar todos los datos.



The screenshot shows the PSeInt IDE with a Pascal program named 'sin_titulo.pas'. The program defines variables for the number of prisoners, average years, prison name, and address. It then prints these values. The execution window shows the following output:

```
*** Ejecución Iniciada ***
El número de presos en la cárcel es: 308
Los presos tienen un promedio de 6 años
El nombre de la cárcel es: 308
La dirección de la cárcel es: Carrera 46 A 8 número correccional 15-28
*** Ejecución Finalizada ***
```

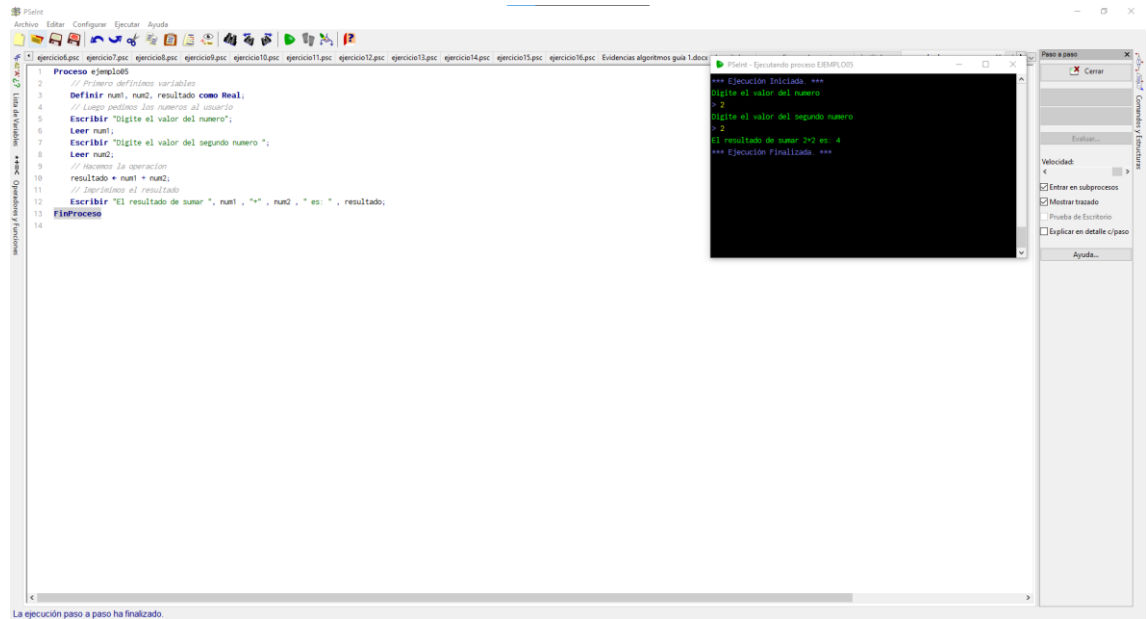
- 4) Se requiere un algoritmo que permita solicitar un número al usuario y calcular la siguiente fórmula: $x \leftarrow -2 * A + A * 5$



The screenshot shows the PSeInt IDE with a Pascal program named 'ejemplo4.pas'. The program declares a variable A, prompts the user for a value, and then calculates the result of the formula $x \leftarrow -2 * A + A * 5$. The execution window shows the following output:

```
*** Ejecución Iniciada ***
Digite el valor del número
3
El resultado de la operación: 14
*** Ejecución Finalizada ***
```

5) Se requiere un algoritmo que pida 2 números y muestre la suma de ambos



The screenshot shows the PSeInt IDE interface. The main editor displays a program named 'Proceso ejemplo5' with the following code:

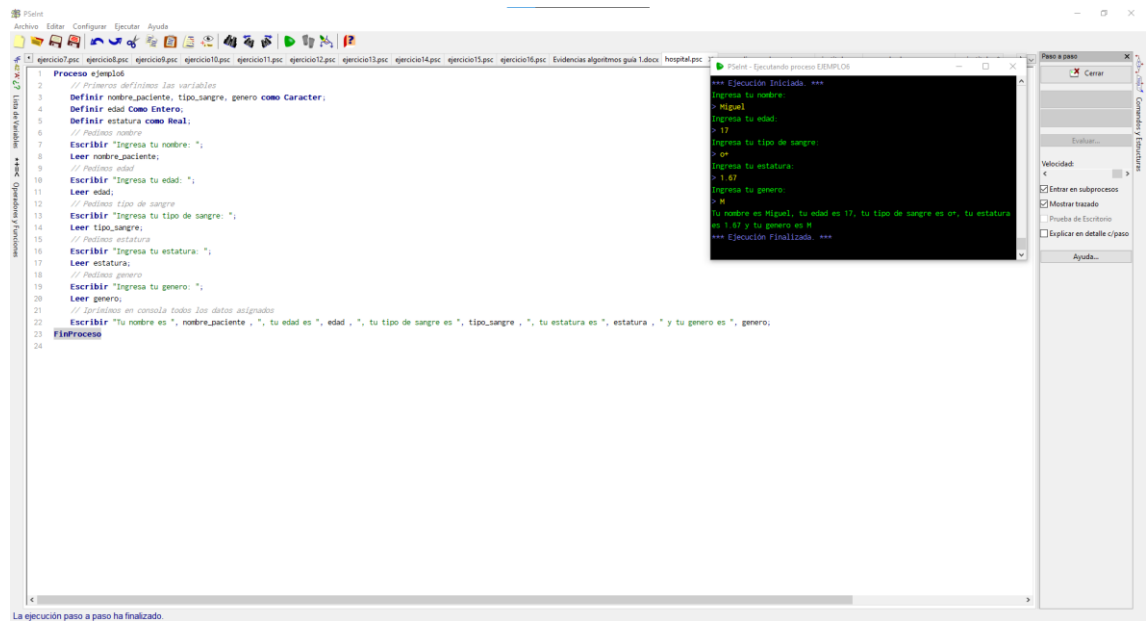
```
1 // Primeros definimos las variables
2 Definir num1, num2, resultado como Real;
3 // Luego pedimos los numeros al usuario
4 Escribir "Digite el valor del numero";
5 Leer num1;
6 Escribir "Digite el valor del segundo numero ";
7 Leer num2;
8 // Hacemos la operacion
9 resultado = num1 + num2;
10 // Mostramos el resultado
11 Escribir "El resultado de sumar ", num1, " + ", num2, " es: ", resultado;
12 FinProceso
```

The output window on the right shows the execution process:

```
*** Ejecución Iniciada ***
Digite el valor del numero
> 2
Digite el valor del segundo numero
> 2
El resultado de sumar 2+2 es: 4
*** Ejecución Finalizada ***
```

The status bar at the bottom indicates 'La ejecución paso a paso ha finalizado'.

6) En un hospital se quiere un algoritmo que solicite al usuario los siguientes datos: Nombre del paciente, edad, tipo de sangre, estatura y género. Mostrar todos los datos al final.



The screenshot shows the PSeInt IDE interface. The main editor displays a program named 'Proceso ejemplo6' with the following code:

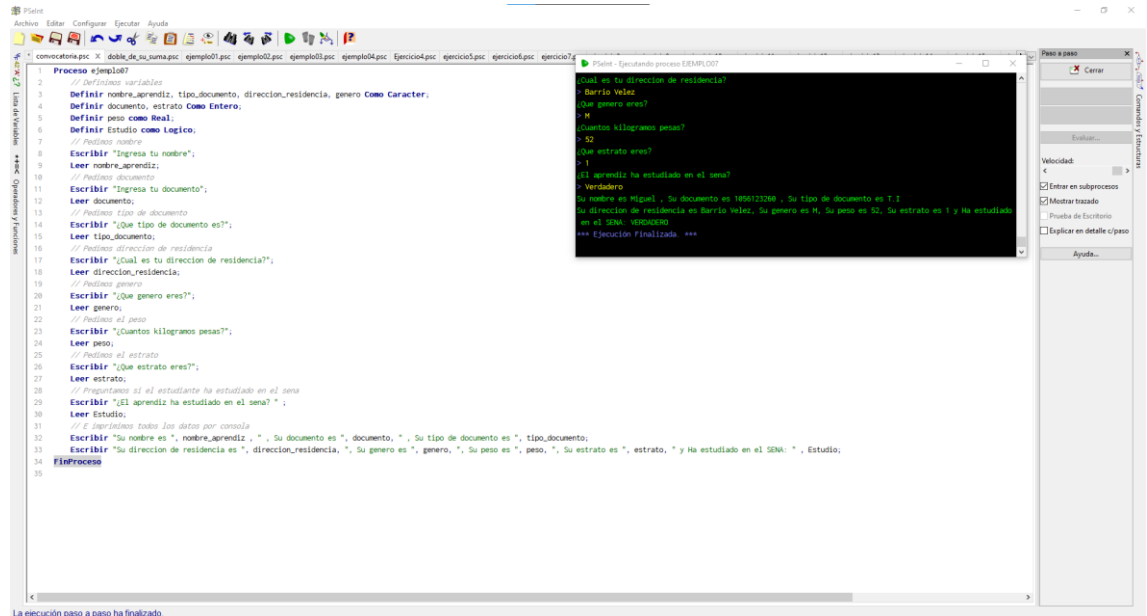
```
1 // Primeros definimos las variables
2 Definir nombre_paciente, tipo_sangre, genero como Caracter;
3 Definir edad Como Entero;
4 Definir estatura como Real;
5 // Pedimos nombre
6 Escribir "Ingresa tu nombre: ";
7 Leer nombre_paciente;
8 // Pedimos edad
9 Escribir "Ingresa tu edad: ";
10 Leer edad;
11 // Pedimos tipo de sangre
12 Escribir "Ingresa tu tipo de sangre: ";
13 Leer tipo_sangre;
14 // Pedimos estatura
15 Escribir "Ingresa tu estatura: ";
16 Leer estatura;
17 // Pedimos genero
18 Escribir "Ingresa tu genero: ";
19 Leer genero;
20 // Imprimos en consola todos los datos asignados
21 Escribir "Tu nombre es ", nombre_paciente, ", tu edad es ", edad, ", tu tipo de sangre es ", tipo_sangre, ", tu estatura es ", estatura, " y tu genero es ", genero;
22 FinProceso
```

The output window on the right shows the execution process:

```
*** Ejecución Iniciada ***
Ingresa tu nombre:
> Miguel
Ingresa tu edad:
> 17
Ingresa tu tipo de sangre:
> O+
Ingresa tu estatura:
> 1.67
Ingresa tu genero:
> M
Tu nombre es Miguel, tu edad es 17, tu tipo de sangre es O+, tu estatura
es 1.67 y tu genero es M
*** Ejecución Finalizada ***
```

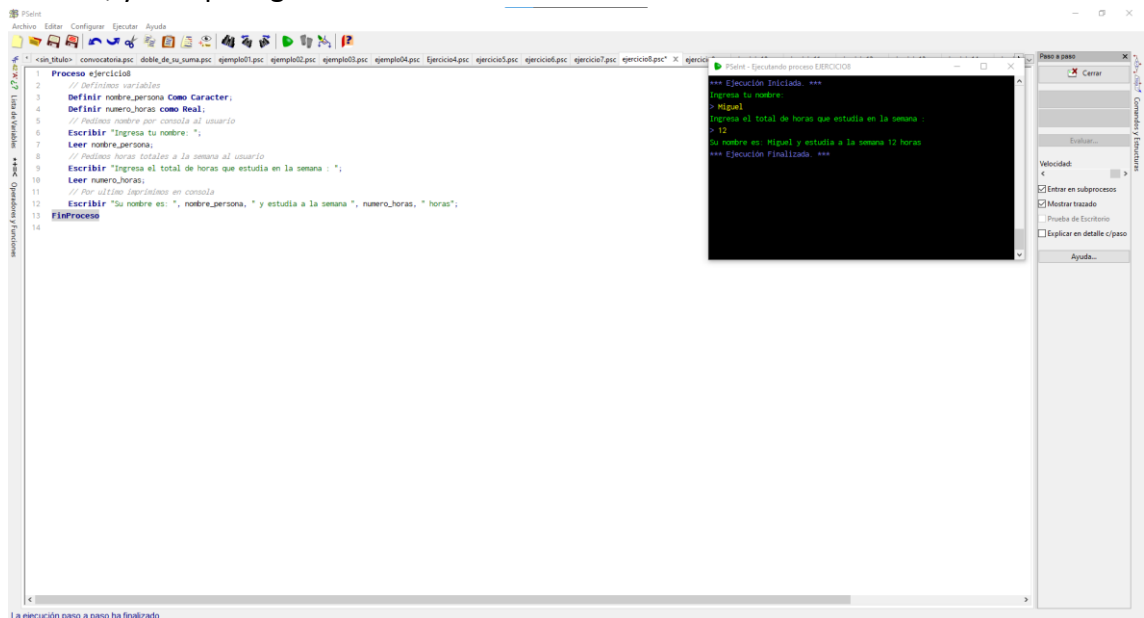
The status bar at the bottom indicates 'La ejecución paso a paso ha finalizado'.

- 7) Se adelanta la convocatoria anual de apoyos de sostenimiento en el SENA Caldas. Se requiere un algoritmo que permita pedir al usuario los siguientes datos del aprendiz: nombre del aprendiz, documento, tipo de documento, dirección de residencia, genero, peso, estrato. Se requiere saber si el aprendiz ha estudiado o no en el SENA. Mostrar toda la información.



Ejercicios

- 1) Hacer un algoritmo que declare una variable para guardar el numero de horas de estudio, y otra para guardar el nombre. Escribir ambos datos.



- 2) Hacer un algoritmo que lea dos números enteros A y B y muestre el doble de su suma.

The screenshot shows the PSeInt IDE with a file named 'Proceso doble_de_su_suma.psc'. The code is as follows:

```
1 Proceso doble_de_su_suma
2 // Definimos las variables
3 Definir num1, num2, resultado Como Entero;
4 // Pedimos valores por consola
5 Escribir "Ingresa el primer numero";
6 Leer num1;
7 Escribir "Ingresa el segundo numero";
8 Leer num2;
9 // Hacemos la operacion
10 resultado ← (num1 + num2) * 2;
11 // Escribimos en consola
12 Escribir "El doble de su suma fue: ", resultado;
13 FinProceso
```

The execution window shows the following output:

```
*** Ejecución Inicializada. ***
Ingresa el primer numero
> 2
Ingresa el segundo numero
> 3
El doble de su suma fue: 10
*** Ejecución Finalizada. ***
```

The 'Paso a paso' (Step by step) panel on the right shows the execution flow, including 'Entrar en subproceso' (Enter sub-process) and 'Mostrar trazado' (Show trace).

- 3) Hacer un algoritmo que declare una variable para guardar el promedio del semestre, otra para guardar el nombre de un estudiante y otra para guardar el número de notas perdidas.

The screenshot shows the PSeInt IDE with a file named 'Proceso ejercicio9.psc'. The code is as follows:

```
1 Proceso ejercicio9
2 // Definimos variables
3 Definir nombre_estudiante Como Caracter;
4 Definir cantidad_perdidas, cantidad_ganadas Como Entero;
5 // Asignamos valores a las variables
6 cantidad_perdidas ← 1;
7 cantidad_ganadas ← 2;
8 // Pedimos nombre del estudiante
9 Escribir "Ingresa el nombre del estudiante";
10 Leer nombre_estudiante;
11 // Por ultimo ponemos todos los datos en consola
12 Escribir "El estudiante ", nombre_estudiante, " tiene ", cantidad_perdidas, " materia perdida y ", cantidad_ganadas, " materias ganadas";
13 FinProceso
```

The execution window shows the following output:

```
*** Ejecución Inicializada. ***
Ingresa el nombre del estudiante
> Miguel
El estudiante Miguel tiene 1 materia perdida y 2 materias ganadas
*** Ejecución Finalizada. ***
```

The 'Paso a paso' (Step by step) panel on the right shows the execution flow, including 'Entrar en subproceso' (Enter sub-process) and 'Mostrar trazado' (Show trace).

- 4) Hacer un algoritmo que lea dos números enteros A y B y muestre el resultado de realizar: $(A + B) * 2 + 10$

The screenshot shows the PSeInt IDE with a file named 'Proceso Ejercicio4'. The code defines variables A and B as integers, prompts the user to enter values, reads the input, and calculates the result using the formula $(A + B) * 2 + 10$. The execution window shows the user entering 2 for both A and B, resulting in 18. The 'Paso a paso' (Step by step) panel on the right shows the execution flow.

```
1 Proceso Ejercicio4
2 // Definimos las variables
3 Definir A, B, resultado Como Entero;
4 // Pedimos valores por consola
5 Escribir "Ingresa el primer numero: ";
6 Leer A;
7 Escribir "Ingresa el segundo numero: ";
8 Leer B;
9 // Hacemos operacion
10 resultado ← (A+B) * 2 + 10;
11 // Imprimimos en consola
12 Escribir "El resultado es: ", resultado;
13 FinProceso
```

La ejecución paso a paso ha finalizado.

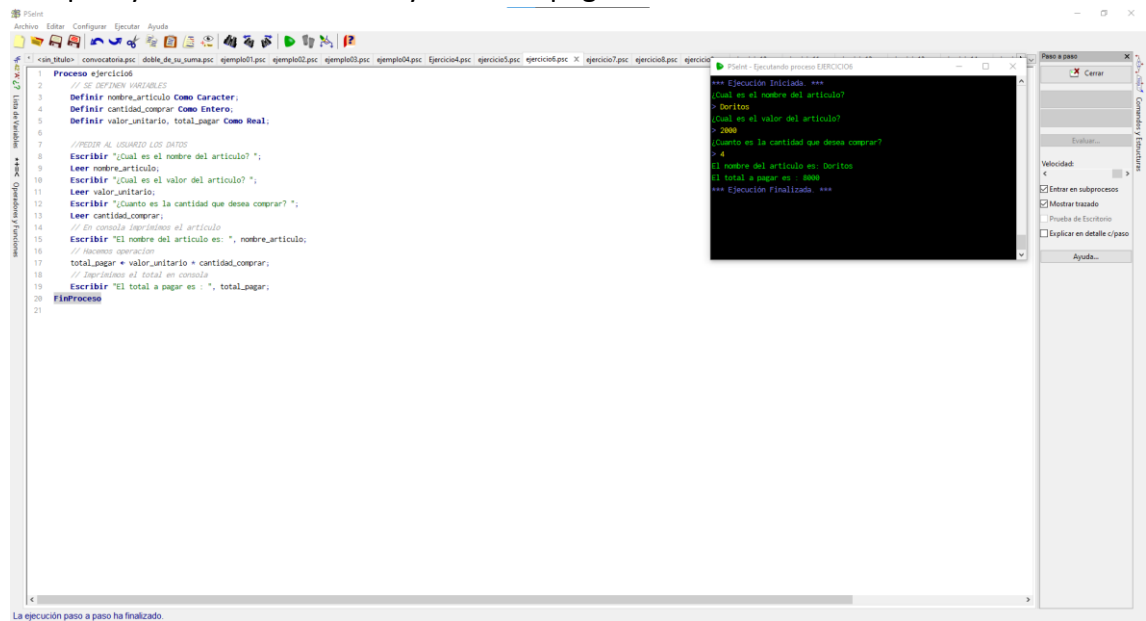
- 5) Hacer un algoritmo que declare una variable para guardar el nombre de una persona, otra para guardar la comida preferida y otra para guardar la cantidad de dinero que posee.

The screenshot shows the PSeInt IDE with a file named 'Proceso Ejercicio5'. The code declares three variables: nombre_persona, comida_favorita, and cantidad_dinero, all as characters. It assigns values to these variables and prints them. The execution window shows the user entering 'Miguel' for the name, 'Frijoles' for the favorite food, and '1 million 500 mil' for the money. The 'Paso a paso' (Step by step) panel on the right shows the execution flow.

```
1 Proceso Ejercicio5
2 // Definimos las variables
3 Definir nombre_persona, comida_favorita, cantidad_dinero Como Caracter;
4 // Les asignamos un valor o u nombre
5 nombre_persona ← "Miguel";
6 comida_favorita ← "Frijoles";
7 cantidad_dinero ← "1 million 500 mil";
8 // Imprimimos en consola
9 Escribir "Su nombre es: ", nombre_persona;
10 Escribir "Su comida favorita es: ", comida_favorita;
11 Escribir "El dinero que posee es: ", cantidad_dinero;
12 FinProceso
```

La ejecución paso a paso ha finalizado.

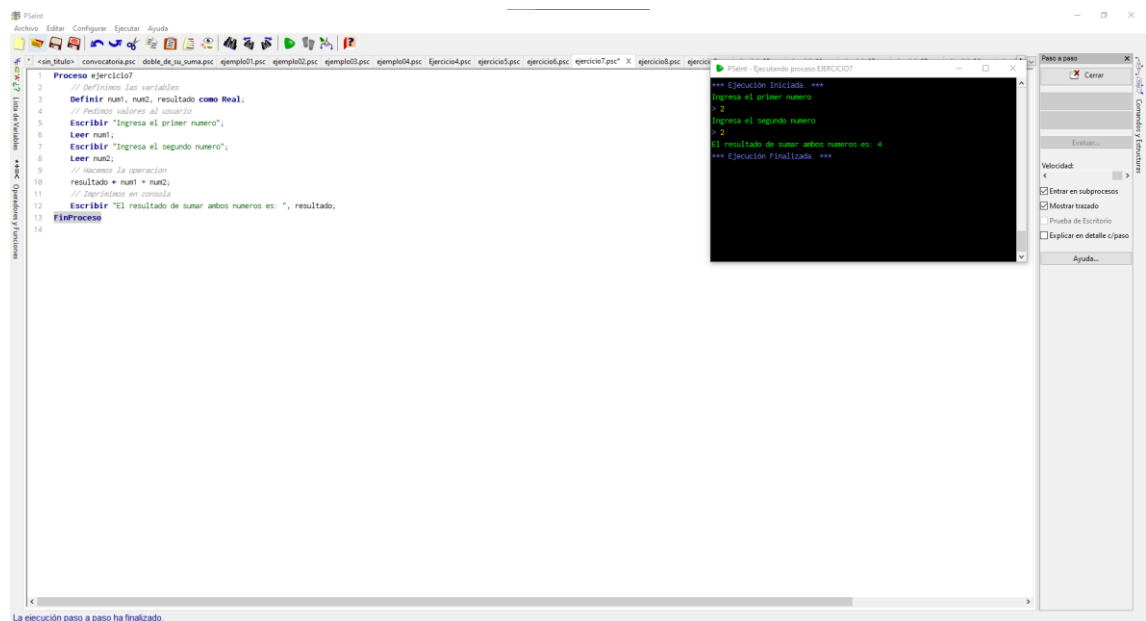
- 6) Hacer un algoritmo que lea el nombre de un artículo, el valor unitario, la cantidad a comprar y muestre el nombre y el total a pagar.



```
1 Proceso ejercicio6
2 // Definir variables
3 Definir nombre_articulo Como Caracter;
4 Definir cantidad_comprar Como Entero;
5 Definir valor_unitario, total_pagar Como Real;
6
7 // Pedir al usuario los datos
8 Escribir "¿Cuál es el nombre del artículo? ";
9 Leer nombre_articulo;
10 Escribir "¿Cuál es el valor del artículo? ";
11 Leer valor_unitario;
12 Escribir "¿Cuanto es la cantidad que desea comprar? ";
13 Leer cantidad_comprar;
14 // En memoria guardamos el artículo
15 Escribir "El nombre del artículo es: ", nombre_articulo;
16 // Hacemos operación
17 total_pagar = valor_unitario * cantidad_comprar;
18 // Escribimos el total en consola
19 Escribir "El total a pagar es: ", total_pagar;
20 FinProceso
21
```

La ejecución paso a paso ha finalizado.

- 7) Hacer un algoritmo para sumar dos números, los cuales serán tecleados por el usuario. Mostrar el resultado



```
1 Proceso ejercicio7
2 // Definimos las variables
3 Definir num1, num2, resultado Como Real;
4 // Pedimos valores al usuario
5 Escribir "Ingresa el primer número";
6 Leer num1;
7 Escribir "Ingresa el segundo número";
8 Leer num2;
9 // Hacemos la operación
10 resultado = num1 + num2;
11 // Escribimos el resultado en consola
12 Escribir "El resultado de sumar ambos números es: ", resultado;
13 FinProceso
14
```

La ejecución paso a paso ha finalizado.

- 8) Hacer un algoritmo que lea el nombre de una persona y número de horas que estudia en la semana.

The screenshot shows the PSeInt IDE with a file named 'ejercicio8.psc'. The code defines variables for a person's name and weekly study hours, prompts the user for input, and then displays the results. The execution window shows the following output:

```
*** Ejecución Iniciada. ***
Ingresa tu nombre:
> Miguel
Ingresa el total de horas que estudia en la semana:
> 48
Su nombre es: Miguel y estudia a la semana 48 horas
*** Ejecución Finalizada. ***
```

The status bar at the bottom indicates 'La ejecución paso a paso ha finalizado.'

- 9) Hacer un algoritmo que lea el nombre de un estudiante, la cantidad de materias perdidas y la cantidad de materias ganadas.

The screenshot shows the PSeInt IDE with a file named 'ejercicio9.psc'. The code defines variables for a student's name, lost subjects, and gained subjects, prompts the user for input, and then displays the results. The execution window shows the following output:

```
*** Ejecución Iniciada. ***
Ingresa el nombre del estudiante:
> Miguel
El estudiante Miguel tiene 1 materia perdida y 2 materias ganadas
*** Ejecución Finalizada. ***
```

The status bar at the bottom indicates 'La ejecución paso a paso ha finalizado.'

- 10) Hacer un algoritmo que lea el alto y el ancho de un rectángulo y muestre su área y su perímetro.

The screenshot shows the PSeInt IDE with a file named 'ejercicio10.psc'. The code defines variables for height, width, area, and perimeter, prompts the user for input, and calculates the results. The execution window shows the program running with inputs 4 and 6, resulting in an area of 24 and a perimeter of 20. The 'Paso a paso' (Step by step) window is open on the right, showing the execution flow.

```
1 Proceso ejercicio10
2 // Definimos las variables
3 Definir alto, ancho, resultado_area, resultado_perimetro como Real;
4 // Ingresamos los valores por consola
5 Escribir "Ingresa el alto del rectangulo";
6 Leer alto;
7 Escribir "Ingresa el ancho del rectangulo";
8 Leer ancho;
9 // Hacemos la operacion
10 resultado_area = alto * ancho;
11 resultado_perimetro = (alto + ancho) * 2;
12 // Imprimos el resultado
13 Escribir "El area del rectangulo es ", resultado_area, " y el perimetro del rectangulo es ", resultado_perimetro;
14 FinProceso
```

*** Ejecución Inicializada. ***
Ingresa el alto del rectangulo
> 4
Ingresa el ancho del rectangulo
> 6
El area del rectangulo es 24 y el perimetro del rectangulo es 20
*** Ejecución Finalizada. ***

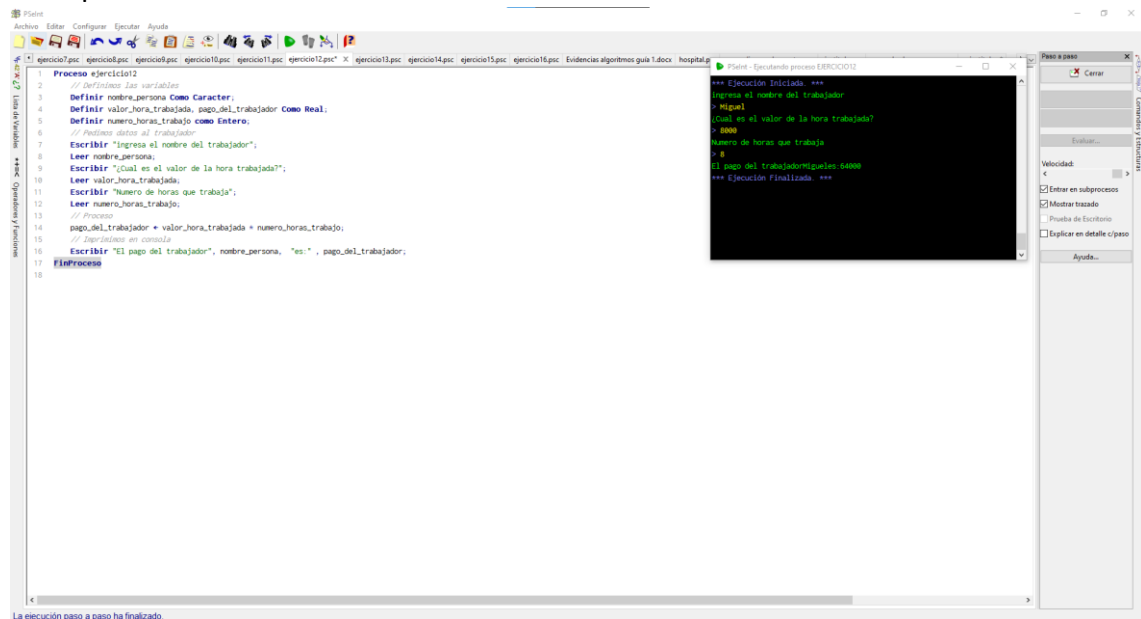
- 11) Hacer un algoritmo que lea dos números enteros A y B y muestre su diferencia.

The screenshot shows the PSeInt IDE with a file named 'ejercicio11.psc'. The code defines variables for two numbers and their difference, prompts the user for input, and calculates the difference. The execution window shows the program running with inputs 6 and 4, resulting in a difference of 2. The 'Paso a paso' (Step by step) window is open on the right, showing the execution flow.

```
1 Proceso ejercicio11
2 // Definimos variables
3 Definir A, B, diferencia Como Entero;
4 // Pedimos valores al usuario
5 Escribir "Ingresa el primer numero";
6 Leer A;
7 Escribir "Ingresa el segundo numero";
8 Leer B;
9 // Hacemos la operacion
10 diferencia = A - B;
11 // Imprimos en consola
12 Escribir "La diferencia entre ambos numeros es: ", diferencia;
13 FinProceso
```

*** Ejecución Inicializada. ***
Ingresa el primer numero
> 6
Ingresa el segundo numero
> 4
La diferencia entre ambos numeros es: 2
*** Ejecución Finalizada. ***

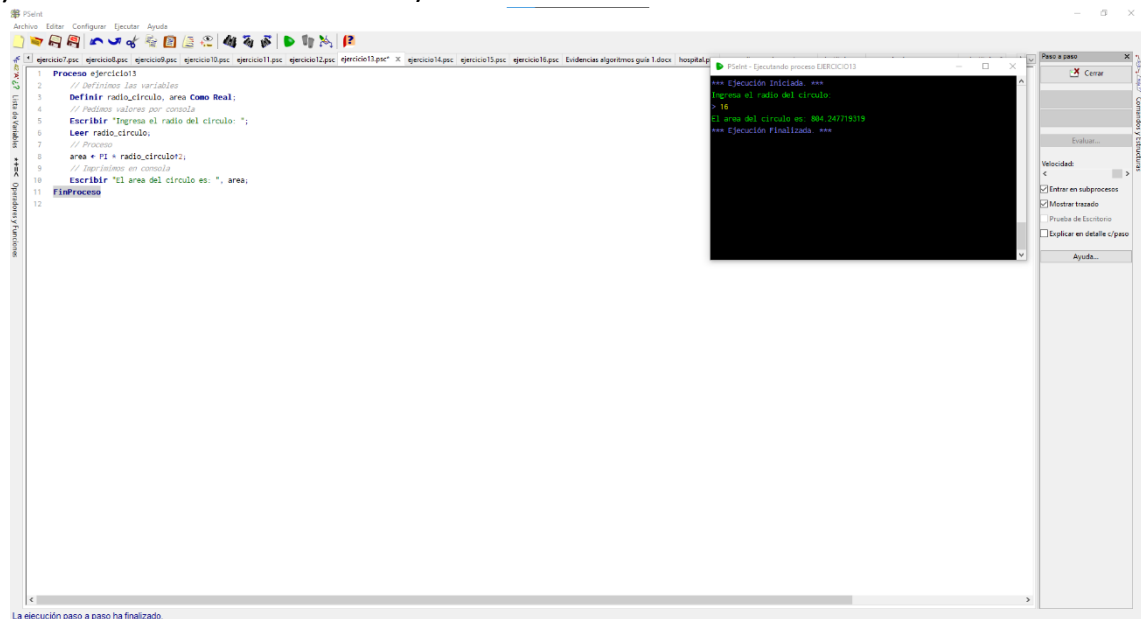
- 12) Hacer un algoritmo que lea el nombre de una persona, el valor de la hora trabajada y el número de horas que trabajó. Se debe mostrar el nombre y el pago de la persona.



```
1 Proceso ejercicio12
2 // Definimos las variables
3 Definir nombre_persona Como Caracter;
4 Definir valor_hora_trabajada, pago_del_trabajador Como Real;
5 Definir numero_horas_trabajo Como Entero;
6 // Pedimos datos al trabajador
7 Escribir "Ingresa el nombre del trabajador";
8 Leer nombre_persona;
9 Escribir "¿Cuál es el valor de la hora trabajada?";
10 Leer valor_hora_trabajada;
11 Escribir "Número de horas que trabaja";
12 Leer numero_horas_trabajo;
13 // Proceso
14 pago_del_trabajador ← valor_hora_trabajada * numero_horas_trabajo;
15 // Imprimos en consola
16 Escribir "El pago del trabajador", nombre_persona, " es: ", pago_del_trabajador;
17 FinProceso
```

La ejecución paso a paso ha finalizado.

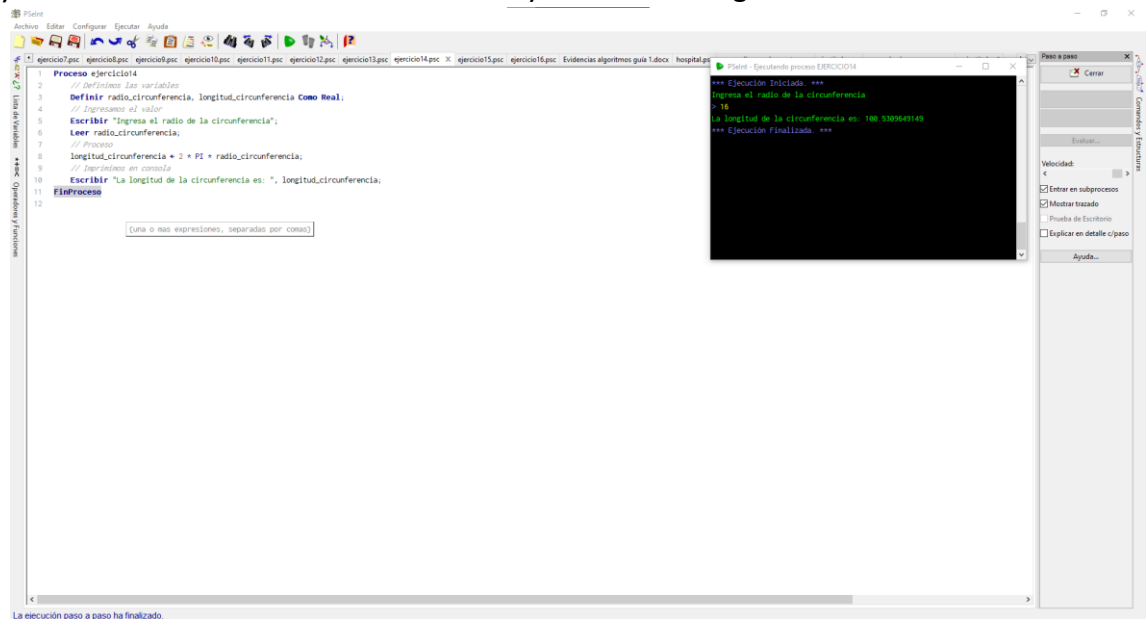
- 13) Pedir el radio de un círculo y calcular su área. $A = \pi * r^2$.



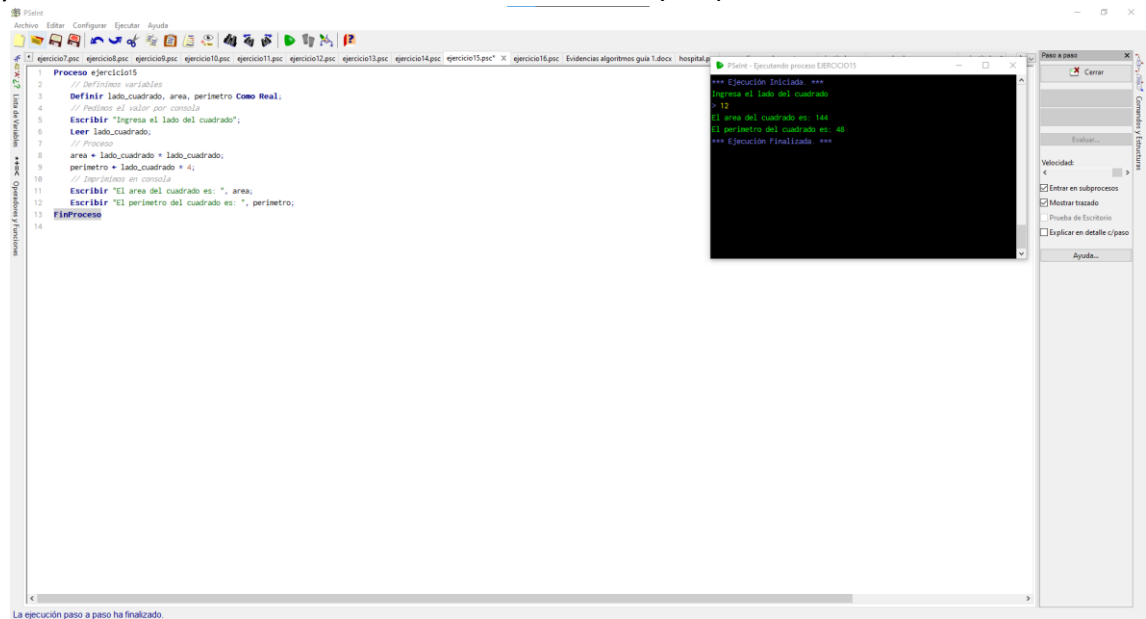
```
1 Proceso ejercicio13
2 // Definimos las variables
3 Definir radio_circulo, area Como Real;
4 // Pedimos valores por consola
5 Escribir "Ingresa el radio del círculo ";
6 Leer radio_circulo;
7 // Proceso
8 area ← Pi * radio_circulo^2;
9 // Imprimos en consola
10 Escribir "El área del círculo es: ", area;
11 FinProceso
```

La ejecución paso a paso ha finalizado.

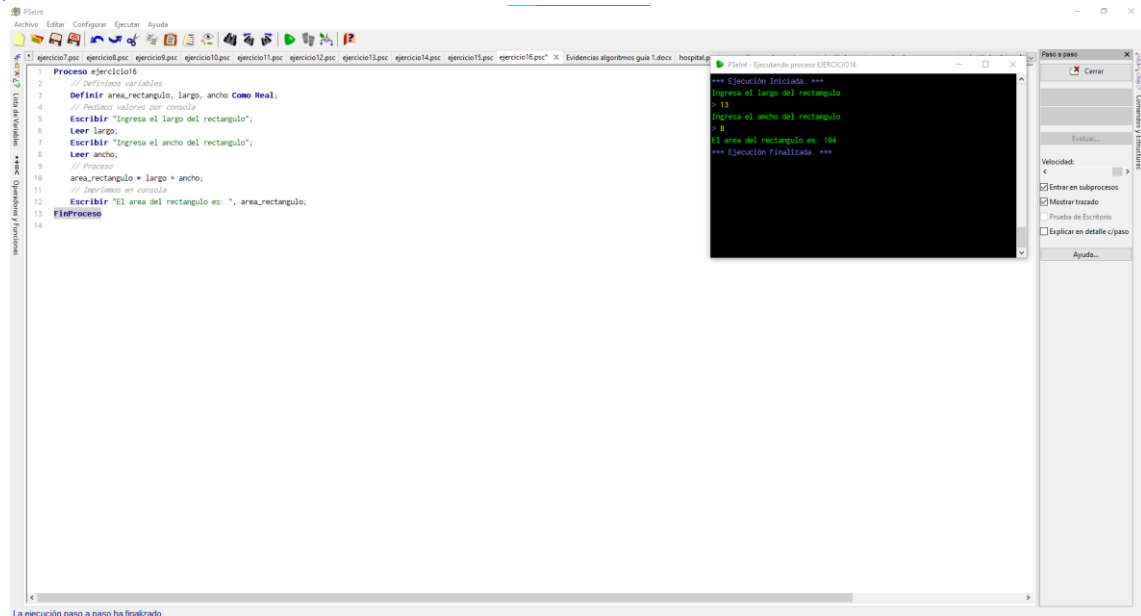
14) Pedir el radio de una circunferencia y calcular su longitud



15) Pedir el lado de un cuadrado, mostrar su área y su perímetro.



16) Calcular el área de un rectángulo de lados X e Y.



The screenshot shows the PSeInt IDE with a file named 'ejercicio16.pse'. The code is as follows:

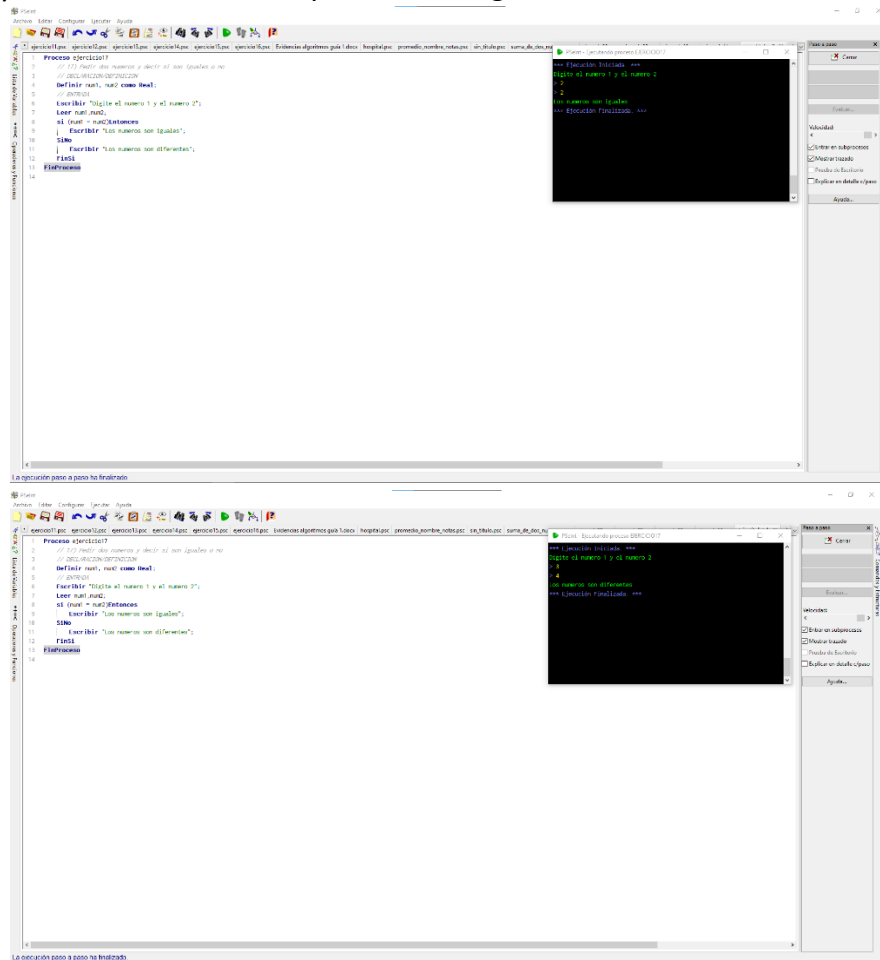
```
1 Proceso ejercicio16
2 // Definir variables
3 Definir area_rectangulo, largo, ancho Como Real;
4 // Pedir valores por consola
5 Escribir "Ingresa el largo del rectangulo";
6 Leer largo;
7 Escribir "Ingresa el ancho del rectangulo";
8 Leer ancho;
9 // Proceso
10 area_rectangulo = largo * ancho;
11 // Escribir en consola
12 Escribir "El area del rectangulo es: ", area_rectangulo;
13 FinProceso
14
```

The execution window shows the following output:

```
*** Ejecución Iniciada ***
Ingresa el largo del rectangulo
> 13
Ingresa el ancho del rectangulo
> 3
El area del rectangulo es: 39
*** Ejecución Finalizada ***
```

The status bar at the bottom indicates 'La ejecución paso a paso ha finalizado.'

17) Pedir dos números y decir si son iguales o no.



The first screenshot shows the PSeInt IDE with a file named 'ejercicio17.pse'. The code is as follows:

```
1 Proceso ejercicio17
2 // Pedir dos numeros y decir si son iguales o no
3 // Declarar variables
4 Definir num1, num2 Como Real;
5 // Proceso
6 Escribir "Ingresa el numero 1 y el numero 2";
7 Leer num1;
8 Leer num2;
9 Si num1 = num2 Entonces
10 | Escribir "Los numeros son iguales";
11 | FinSi
12 Escribir "Los numeros son diferentes";
13 FinProceso
14
```

The execution window shows the following output:

```
*** Ejecución Iniciada ***
Ingresa el numero 1 y el numero 2
> 1
Ingresa el numero 2
> 2
Los numeros son diferentes
*** Ejecución Finalizada ***
```

The status bar at the bottom indicates 'La ejecución paso a paso ha finalizado.'

The second screenshot shows the same PSeInt IDE with the same file, but the code is slightly different:

```
1 Proceso ejercicio17
2 // Pedir dos numeros y decir si son iguales o no
3 // Declarar variables
4 Definir num1, num2 Como Real;
5 // Proceso
6 Escribir "Ingresa el numero 1 y el numero 2";
7 Leer num1;
8 Leer num2;
9 Si num1 = num2 Entonces
10 | Escribir "Los numeros son iguales";
11 | FinSi
12 Escribir "Los numeros son diferentes";
13 FinProceso
14
```

The execution window shows the following output:

```
*** Ejecución Iniciada ***
Ingresa el numero 1 y el numero 2
> 1
Ingresa el numero 2
> 1
Los numeros son iguales
*** Ejecución Finalizada ***
```

The status bar at the bottom indicates 'La ejecución paso a paso ha finalizado.'

18) Pedir un número e indicar si es positivo o negativo.

The first screenshot shows the IDE with a Java file named 'ejercicio18.java'. The code is as follows:

```
1 Proceso ejercicio18
2 // Definición de variables
3 Definir num como Real;
4 // Leer número
5 Escribir "Ingresa el número";
6 Leer num;
7 // Proceso = null
8 Si (num > 0) Entonces
9     Escribir "El número es positivo";
10 sino
11     Escribir "El número es negativo";
12 FinSi
13 FinProceso
```

The second screenshot shows the same IDE with the program executed. The output window displays the following text:

```
*** Ejecución Inicializada ***
Ingresa el número
5
El número es positivo
*** Ejecución Finalizada ***
```

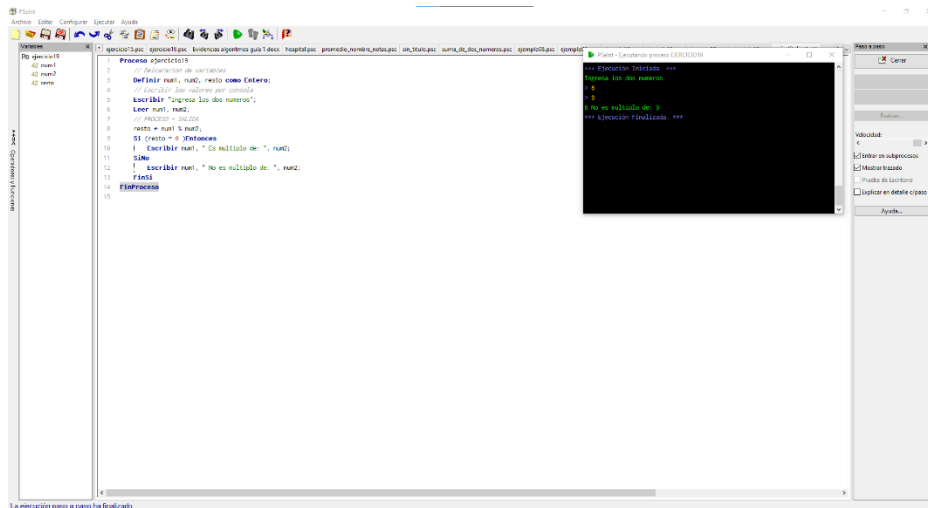
19) Pedir dos números y decir si uno es múltiplo del otro.

The screenshot shows the IDE with a Java file named 'ejercicio19.java'. The code is as follows:

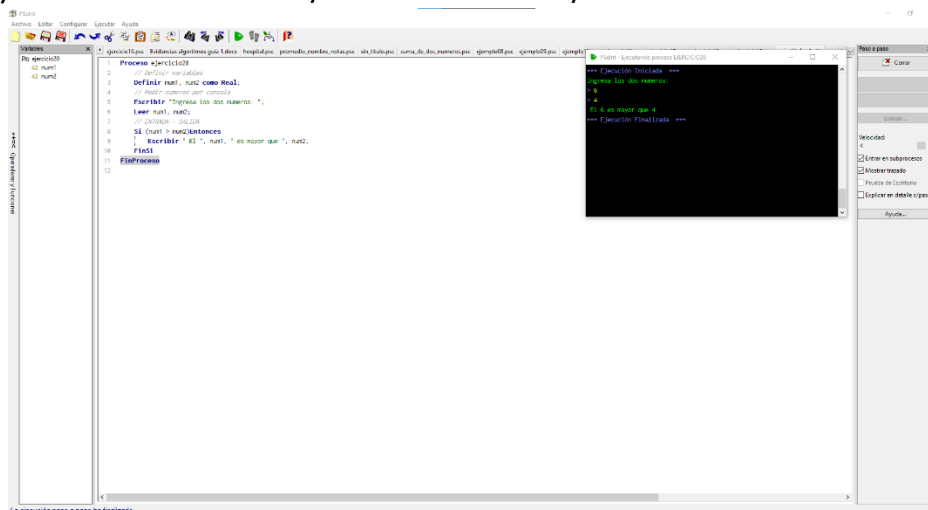
```
1 Proceso ejercicio19
2 // Definición de variables
3 Definir num1, num2, resto como Entero;
4 // Leer los valores por consola
5 Escribir "Ingresa los dos números";
6 Leer num1, num2;
7 // Proceso = null
8 resto = num1 % num2;
9 Si (resto = 0) Entonces
10     Escribir num1, " es múltiplo de ", num2;
11 sino
12     Escribir num1, " No es múltiplo de ", num2;
13 FinSi
14 FinProceso
```

The output window displays the following text:

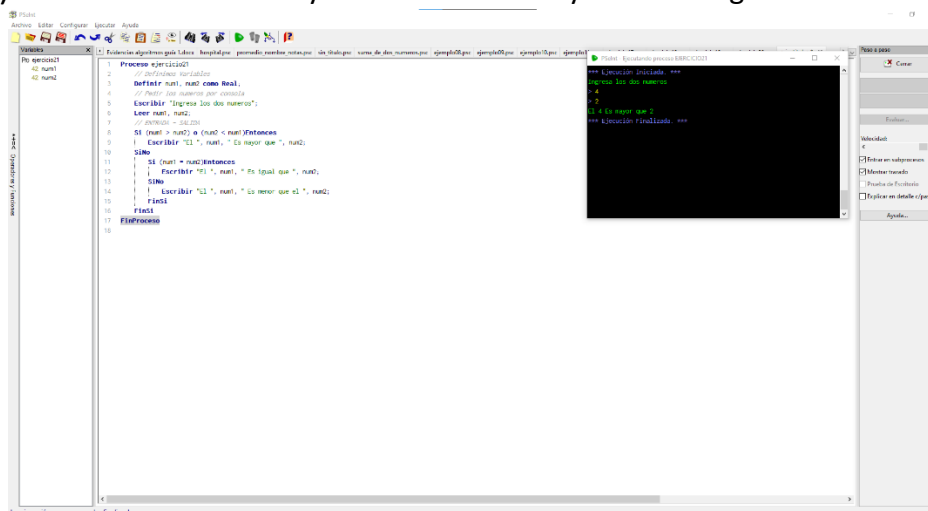
```
*** Ejecución Inicializada ***
Ingresa los dos números
5
2
5 es múltiplo de 2
*** Ejecución Finalizada ***
```

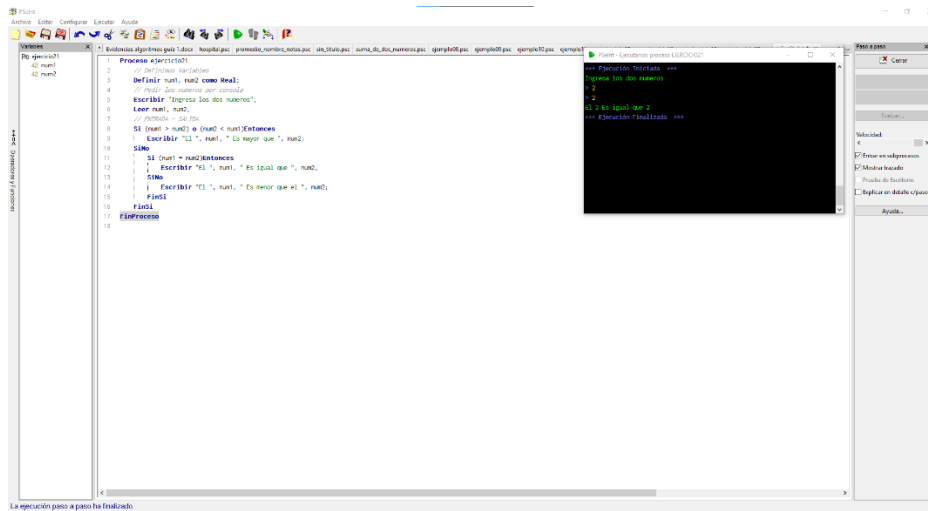


20) Pedir dos números y decir cuál es el mayor

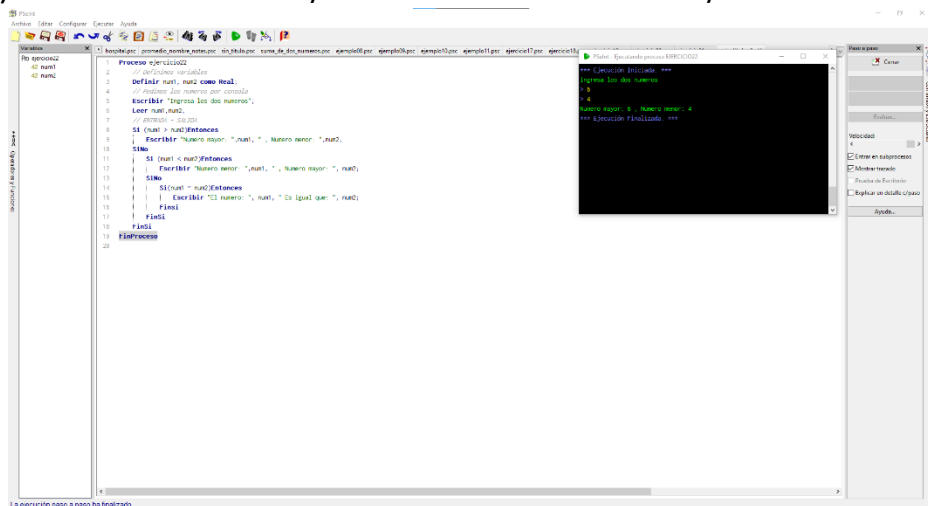


21) Pedir dos números y decir cuál es el mayor o si son iguales.

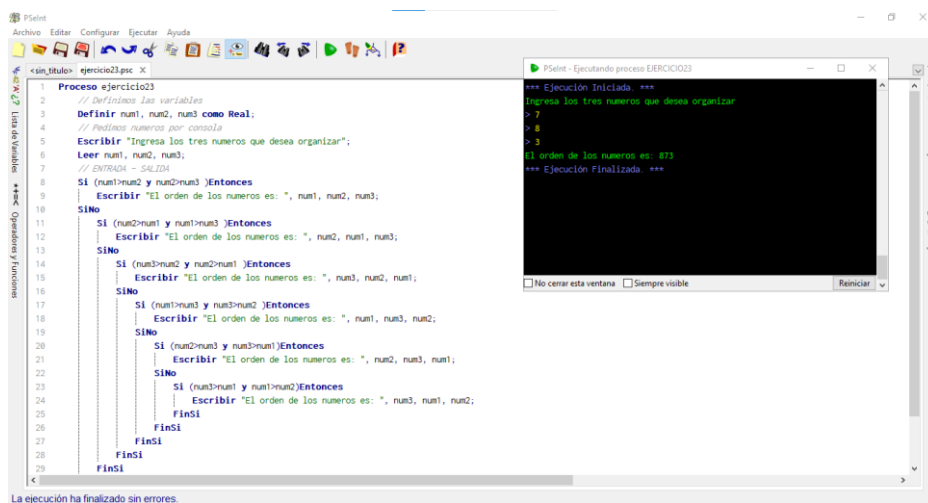




22) Pedir dos números y mostrarlos ordenados de mayor a menor.



23) Pedir tres números y mostrarlos ordenados de mayor a menor.



24) Pedir un número entre 0 y 9.999 y decir cuántas cifras tiene.

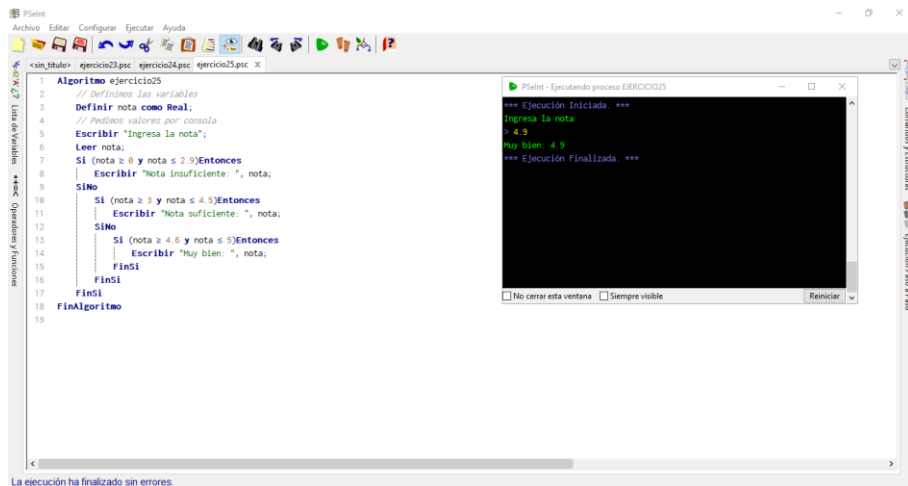
```
1 Algoritmo ejercicio24
2 // Definimos variables
3 Definir num1, cifras como Entero;
4 // Pedimos número por consola
5 Escribir "Ingrese un número entre 0 y 9.999: ";
6 Leer num1;
7 // ENTRADA - SALIDA
8 Si num1 ≥ 0 y num1 ≤ 9 Entonces
9     cifras ← 1;
10 FinSi
11
12 Si num1 ≥ 10 y num1 ≤ 99 Entonces
13     cifras ← 2;
14 FinSi
15
16 Si num1 ≥ 100 y num1 ≤ 999 Entonces
17     cifras ← 3;
18 FinSi
19
20 Si num1 ≥ 1000 y num1 ≤ 9999 Entonces
21     cifras ← 4;
22 FinSi
23
24 Escribir "El número ", num1, " tiene ", cifras, " cifras.";
25 Finalgoritmo
26
```

La ejecución ha finalizado sin errores.

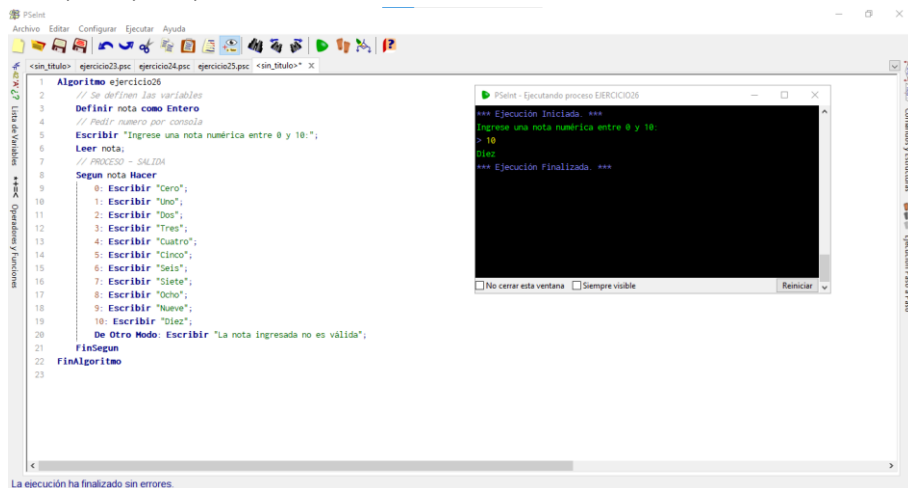
25) Pedir una nota de 0 a 5 y mostrarla de la forma: Insuficiente (0 – 2,9), Suficiente (3 – 4,5) y Bien (4,6 – 5).

```
1 Algoritmo ejercicio25
2 // Definimos las variables
3 Definir nota como Real;
4 // Pedimos valores por consola
5 Escribir "Ingresa la nota";
6 Leer nota;
7 Si (nota ≥ 0 y nota ≤ 2.9)Entonces
8     Escribir "Nota insuficiente: ", nota;
9 Sino
10     Si (nota ≥ 3 y nota ≤ 4.5)Entonces
11         Escribir "Nota suficiente: ", nota;
12     Sino
13         Si (nota ≥ 4.6 y nota ≤ 5)Entonces
14             Escribir "Muy bien: ", nota;
15         FinSi
16     FinSi
17 FinSi
18 Finalgoritmo
19
```

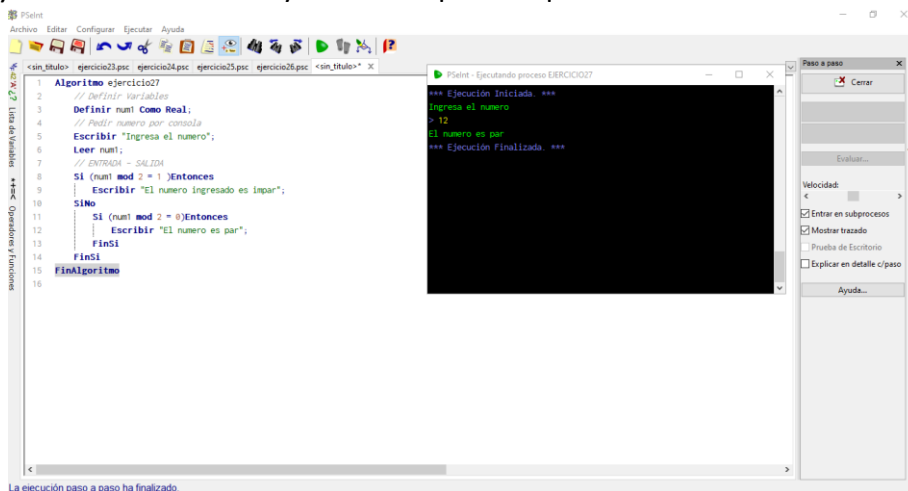
La ejecución ha finalizado sin errores.

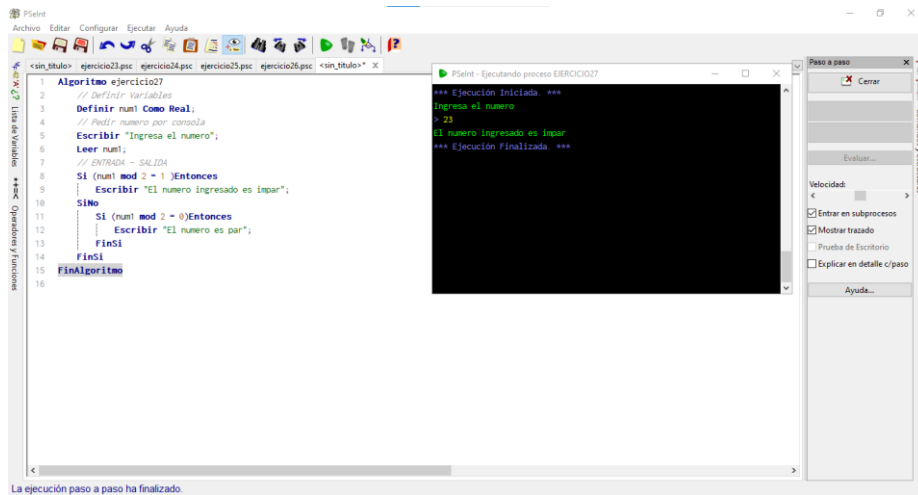


26) Pedir una nota numérica entera entre 0 y 10, y mostrar dicha nota de la forma: cero, uno, dos, tres...

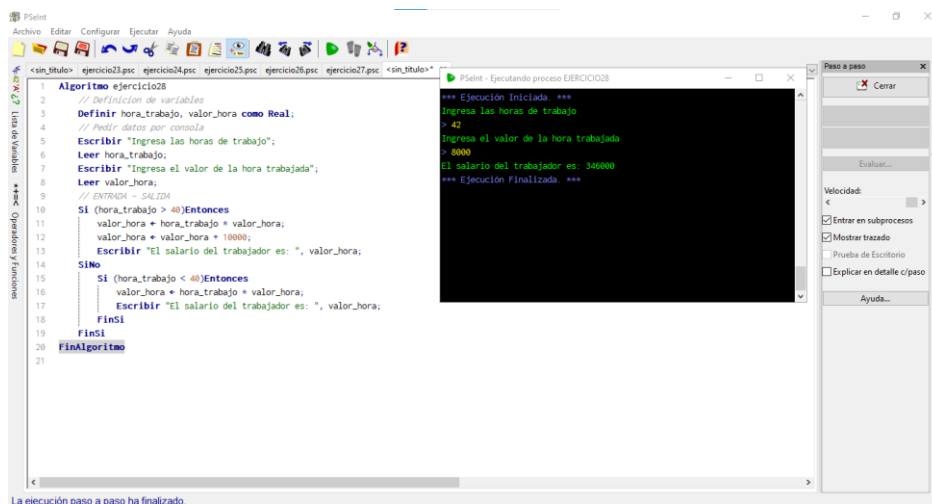


27) Pedir un número y decir si es par o impar.

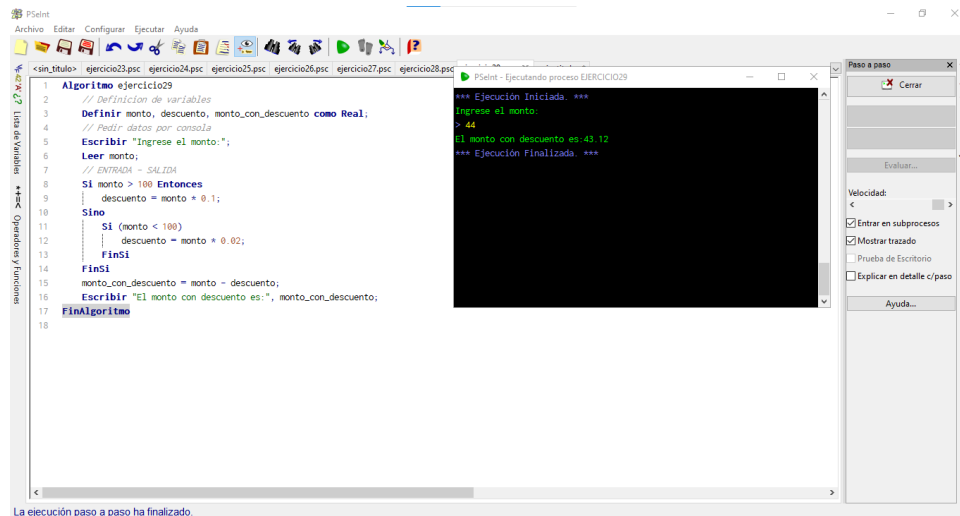




- 28) Un trabajador recibe su pago, según la cantidad de horas trabajadas y su valor. Si la cantidad de horas trabajadas es mayor que 40, éstas se consideran horas extra, y tienen un incremento de \$10000 (diez mil) sobre el valor de la hora. Calcular y mostrar el salario (pago) del trabajador. Nota: leer horas trabajadas y valor de la hora.

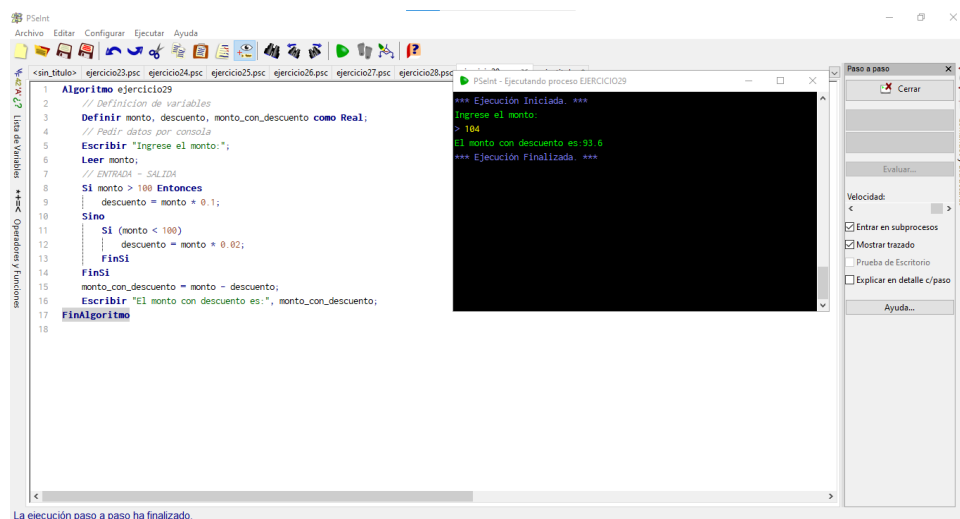


- 29) Dado un monto, calcular el descuento considerando que por encima de 100 el descuento es del 10% y por debajo de 100, el descuento es del 2%.



```
1 Algoritmo ejercicio29
2 // Definición de variables
3 Definir monto, descuento, monto_con_descuento como Real;
4 // Pedir datos por consola
5 Escribir "Ingrese el monto:";
6 Leer monto;
7 // ENTRADA - SALIDA
8 Si monto > 100 Entonces
9     descuento = monto * 0.1;
10 Sino
11     Si (monto < 100)
12         descuento = monto * 0.02;
13     FinSi
14 FinSi
15 monto_con_descuento = monto - descuento;
16 Escribir "El monto con descuento es:", monto_con_descuento;
17 FinAlgoritmo
```

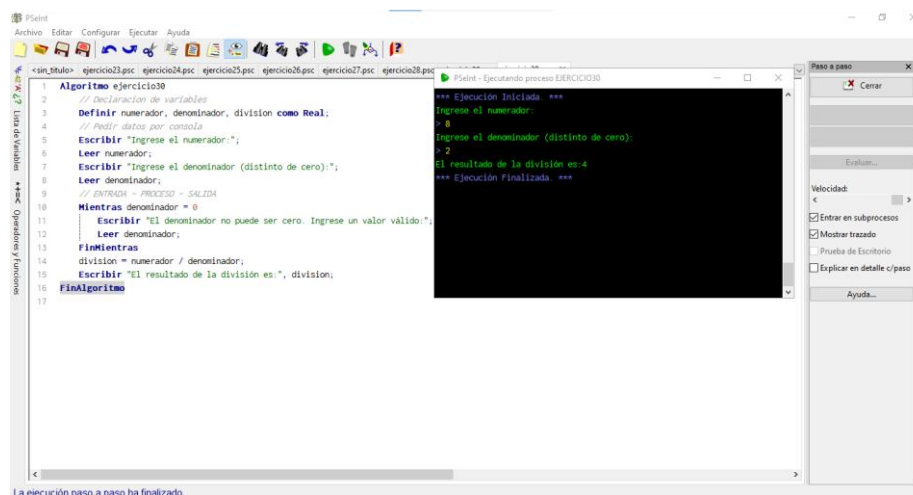
La ejecución paso a paso ha finalizado.



```
1 Algoritmo ejercicio29
2 // Definición de variables
3 Definir monto, descuento, monto_con_descuento como Real;
4 // Pedir datos por consola
5 Escribir "Ingrese el monto:";
6 Leer monto;
7 // ENTRADA - SALIDA
8 Si monto > 100 Entonces
9     descuento = monto * 0.1;
10 Sino
11     Si (monto < 100)
12         descuento = monto * 0.02;
13     FinSi
14 FinSi
15 monto_con_descuento = monto - descuento;
16 Escribir "El monto con descuento es:", monto_con_descuento;
17 FinAlgoritmo
```

La ejecución paso a paso ha finalizado.

- 30) Leer dos números y calcular su división, teniendo en cuenta que el denominador no debe ser 0 (cero).



```
1 Algoritmo ejercicio30
2 // Declaración de variables
3 Definir numerador, denominador, division como Real;
4 // Pedir datos por consola
5 Escribir "Ingrese el numerador:";
6 Leer numerador;
7 Escribir "Ingrese el denominador (distinto de cero):";
8 Leer denominador;
9 // ENTRADA - PROCESO - SALIDA
10 Mientras denominador = 0
11     Escribir "El denominador no puede ser cero. Ingrese un valor válido:";
12     Leer denominador;
13 FinMientras
14 division = numerador / denominador;
15 Escribir "El resultado de la división es:", division;
16 FinAlgoritmo
```

La ejecución paso a paso ha finalizado.