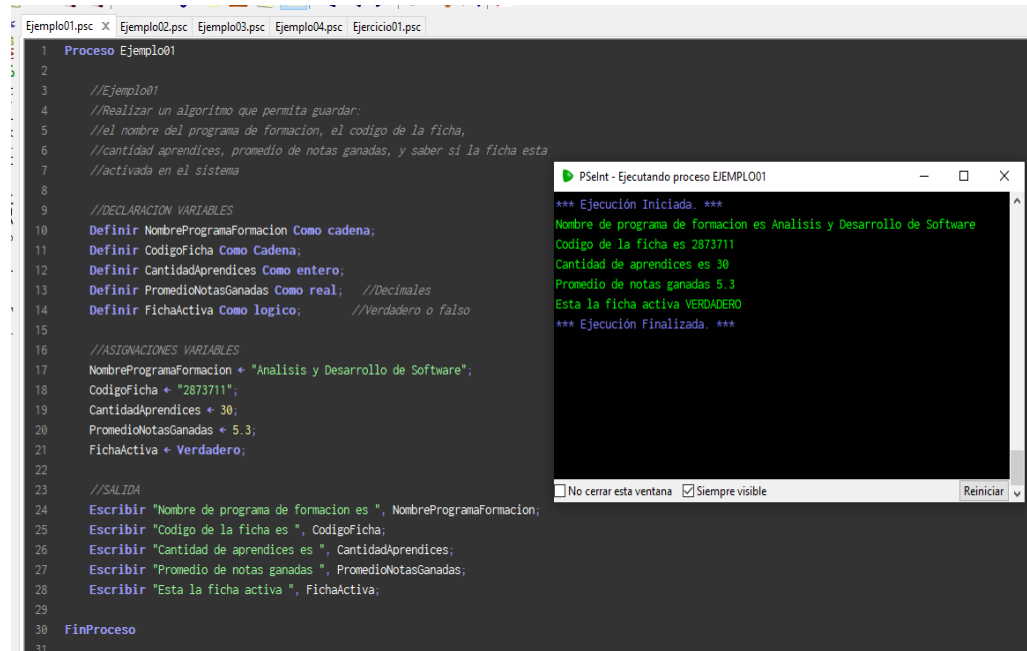


EVIDENCIAS Y EJEMPLOS DE EJERCICIOS ALGORITMOS POR COMPUTADORAS.

1. Ejemplo01

Realizar un algoritmo que permita guardar: el nombre del programa de formación, el código de la ficha, cantidad aprendices, promedio de notas ganadas, y saber si la ficha esta activada en el sistema.

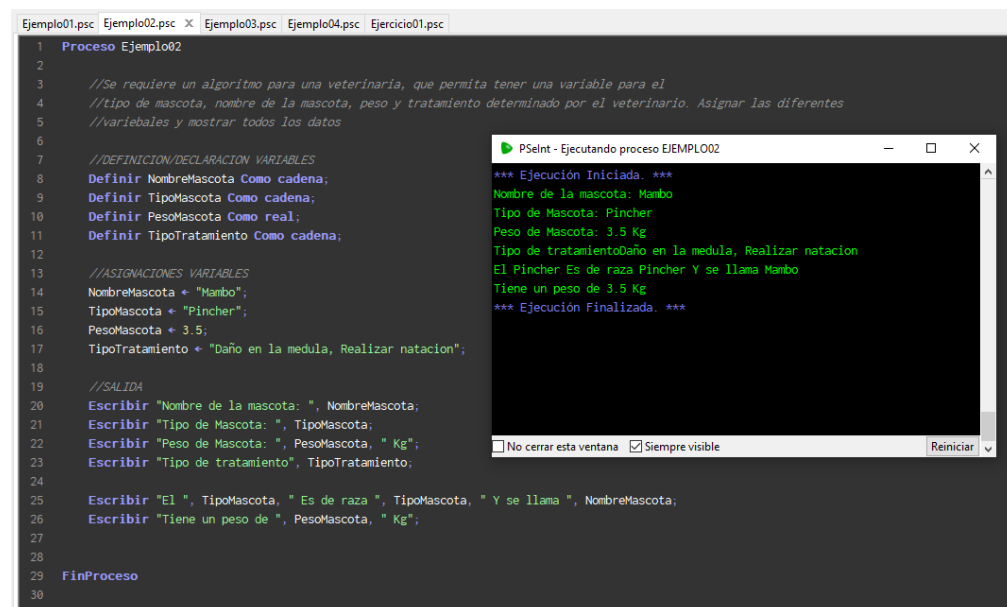


```
1 Proceso Ejemplo01
2
3 //Ejemplo01
4 //Realizar un algoritmo que permita guardar:
5 //el nombre del programa de formacion, el codigo de la ficha,
6 //cantidad aprendices, promedio de notas ganadas, y saber si la ficha esta
7 //activada en el sistema
8
9 //DECLARACION VARIABLES
10 Definir NombreProgramaFormacion Como cadena;
11 DefinirCodigoFicha Como Cadena;
12 Definir CantidadAprendices Como entero;
13 Definir PromedioNotasGanadas Como real; //Decimales
14 Definir FichaActiva Como logico; //Verdadero o falso
15
16 //ASIGNACIONES VARIABLES
17 NombreProgramaFormacion ← "Analisis y Desarrollo de Software";
18 CodigoFicha ← "2873711";
19 CantidadAprendices ← 30;
20 PromedioNotasGanadas ← 5.3;
21 FichaActiva ← Verdadero;
22
23 //SALIDA
24 Escribir "Nombre de programa de formacion es ", NombreProgramaFormacion;
25 Escribir "Codigo de la ficha es ", CodigoFicha;
26 Escribir "Cantidad de aprendices es ", CantidadAprendices;
27 Escribir "Promedio de notas ganadas ", PromedioNotasGanadas;
28 Escribir "Esta la ficha activa ", FichaActiva;
29
30 FinProceso
31
```

*** Ejecución Iniciada. ***
Nombre de programa de formacion es Analisis y Desarrollo de Software
Codigo de la ficha es 2873711
Cantidad de aprendices es 30
Promedio de notas ganadas 5.3
Esta la ficha activa VERDADERO
*** Ejecución Finalizada. ***

2. Ejemplo02

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. Asignar las diferentes variables y mostrar todos los datos.

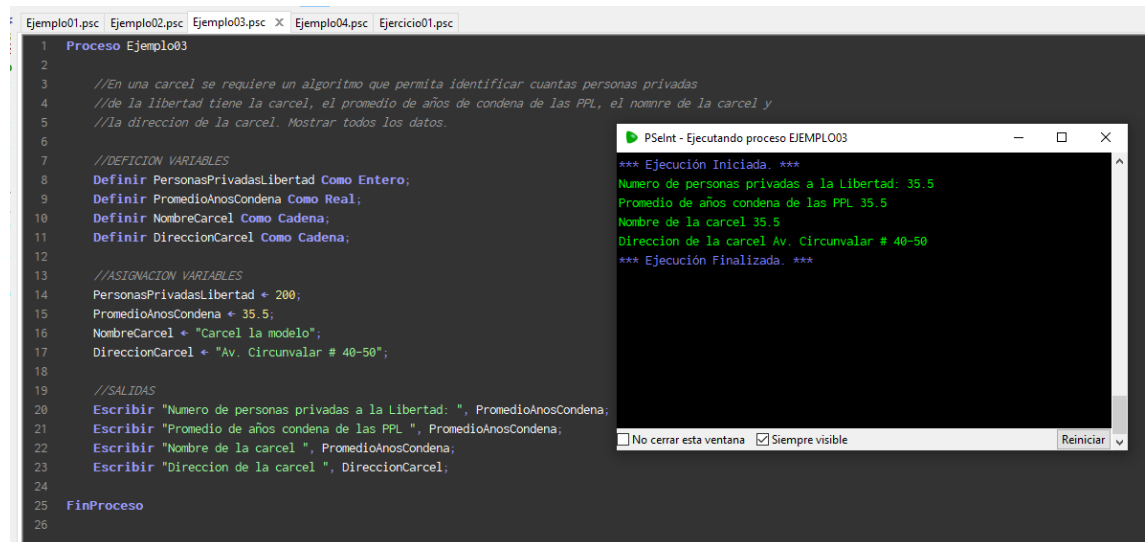


```
1 Proceso Ejemplo02
2
3 //Se requiere un algoritmo para una veterinaria, que permita tener una variable para el
4 //tipo de mascota, nombre de la mascota, peso y tratamiento determinado por el veterinario. Asignar las diferentes
5 //variables y mostrar todos los datos
6
7 //DEFINICION/DECLARACION VARIABLES
8 Definir NombreMascota Como cadena;
9 Definir TipoMascota Como cadena;
10 Definir PesoMascota Como real;
11 Definir TipoTratamiento Como cadena;
12
13 //ASIGNACIONES VARIABLES
14 NombreMascota ← "Manbo";
15 TipoMascota ← "Pincher";
16 PesoMascota ← 3.5;
17 TipoTratamiento ← "Daño en la medula, Realizar natacion";
18
19 //SALIDA
20 Escribir "Nombre de la mascota: ", NombreMascota;
21 Escribir "Tipo de Mascota: ", TipoMascota;
22 Escribir "Peso de Mascota: ", PesoMascota, " Kg";
23 Escribir "Tipo de tratamiento", TipoTratamiento;
24
25 Escribir "El ", TipoMascota, " Es de raza ", TipoMascota, " Y se llama ", NombreMascota;
26 Escribir "Tiene un peso de ", PesoMascota, " Kg";
27
28
29 FinProceso
30
```

*** Ejecución Iniciada. ***
Nombre de la mascota: Manbo
Tipo de Mascota: Pincher
Peso de Mascota: 3.5 Kg
Tipo de tratamiento Daño en la medula, Realizar natacion
El Pincher Es de raza Pincher Y se llama Manbo
Tiene un peso de 3.5 Kg
*** Ejecución Finalizada. ***

3. Ejemplo03

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 y la dirección de la cárcel. Mostrar todos los datos.

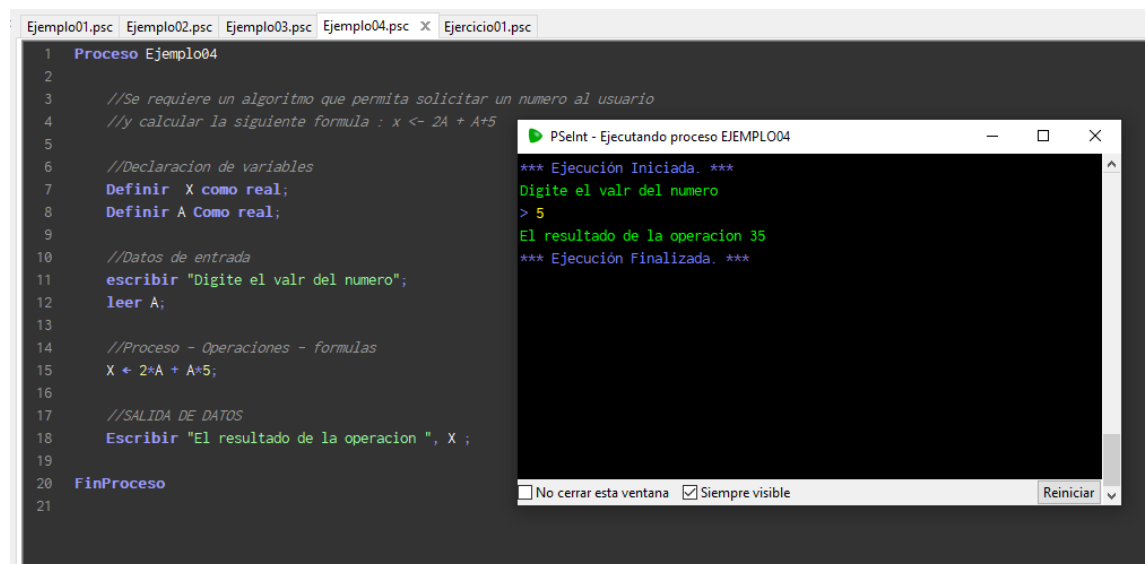


```
1 Proceso Ejemplo03
2
3 //En una cárcel se requiere un algoritmo que permita identificar cuantas personas privadas
4 //de la libertad tiene la cárcel, el promedio de años de condena de las PPL, el nombre de la cárcel y
5 //la dirección de la cárcel. Mostrar todos los datos.
6
7 //DEFINICION VARIABLES
8 Definir PersonasPrivadasLibertad Como Entero;
9 Definir PromedioAnosCondena Como Real;
10 Definir NombreCarcel Como Cadena;
11 Definir DireccionCarcel Como Cadena;
12
13 //ASIGNACION VARIABLES
14 PersonasPrivadasLibertad ← 200;
15 PromedioAnosCondena ← 35.5;
16 NombreCarcel ← "Carcel la modelo";
17 DireccionCarcel ← "Av. Circunvalar # 40-50";
18
19 //SALIDAS
20 Escribir "Numero de personas privadas a la Libertad: ", PromedioAnosCondena;
21 Escribir "Promedio de años condena de las PPL ", PromedioAnosCondena;
22 Escribir "Nombre de la carcel ", PromedioAnosCondena;
23 Escribir "Direccion de la carcel ", DireccionCarcel;
24
25 FinProceso
26
```

*** Ejecución Iniciada. ***
Numero de personas privadas a la Libertad: 35.5
Promedio de años condena de las PPL 35.5
Nombre de la carcel 35.5
Direccion de la carcel Av. Circunvalar # 40-50
*** Ejecución Finalizada. ***

4. Ejemplo04 DESDE AQUÍ 22/02

Se requiere un algoritmo que permita solicitar un numero al usuario y calcular la siguiente formula: $x \leftarrow 2A + A \cdot 5$

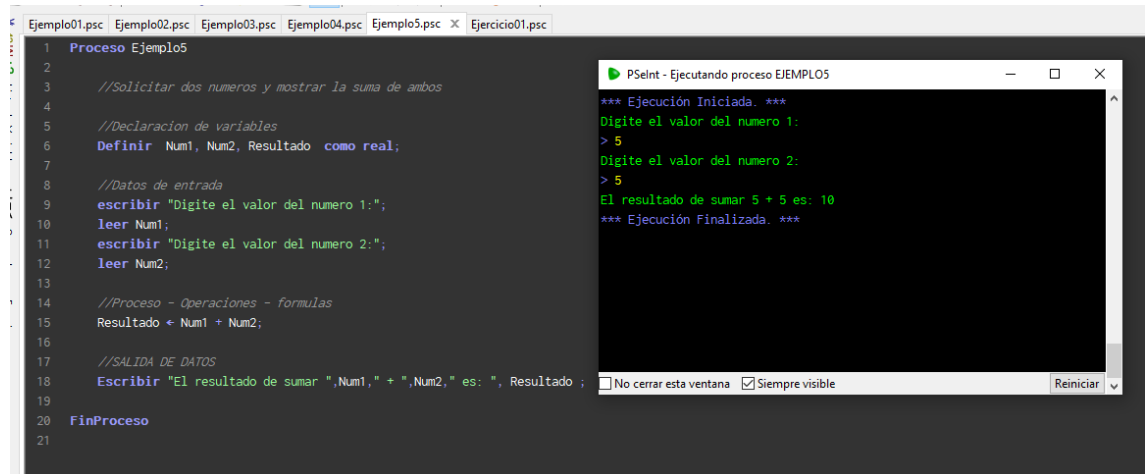


```
1 Proceso Ejemplo04
2
3 //Se requiere un algoritmo que permita solicitar un numero al usuario
4 //y calcular la siguiente formula :  $x \leftarrow 2A + A \cdot 5$ 
5
6 //Declaracion de variables
7 Definir X como real;
8 Definir A Como real;
9
10 //Datos de entrada
11 escribir "Digite el valr del numero";
12 leer A;
13
14 //Proceso - Operaciones - formulas
15  $X \leftarrow 2 \cdot A + A \cdot 5$ ;
16
17 //SALIDA DE DATOS
18 Escribir "El resultado de la operacion ", X ;
19
20 FinProceso
21
```

*** Ejecución Iniciada. ***
Digite el valr del numero
> 5
El resultado de la operacion 35
*** Ejecución Finalizada. ***

5. Ejemplo05

Solicitar dos números y mostrar la suma de ambos



```
1 Proceso Ejemplo05
2
3 //Solicitar dos numeros y mostrar la suma de ambos
4
5 //Declaracion de variables
6 Definir Num1, Num2, Resultado como real;
7
8 //Datos de entrada
9 escribir "Digite el valor del numero 1:";
10 leer Num1;
11 escribir "Digite el valor del numero 2:";
12 leer Num2;
13
14 //Proceso - Operaciones - formulas
15 Resultado ← Num1 + Num2;
16
17 //SALIDA DE DATOS
18 Escribir "El resultado de sumar ",Num1," + ",Num2," es: ", Resultado ;
19
20 FinProceso
21
```

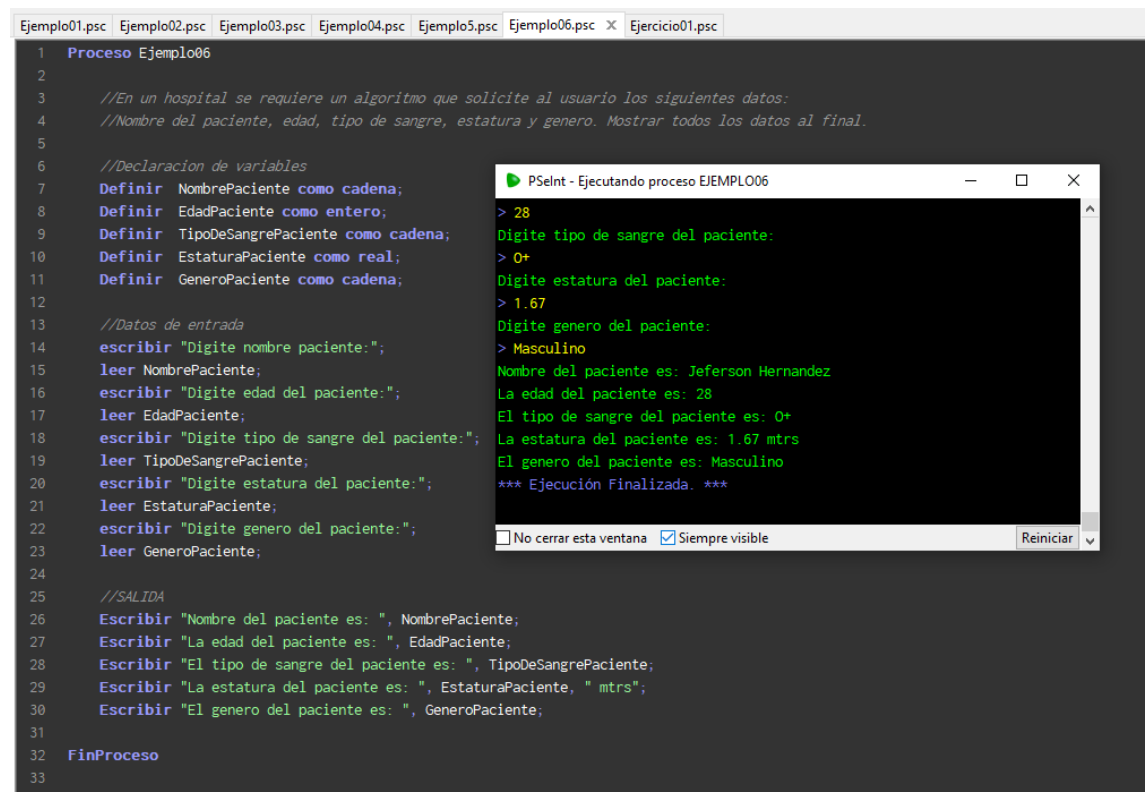
PSeInt - Ejecutando proceso EJEMPLO5

```
*** Ejecución Iniciada. ***
Digite el valor del numero 1:
> 5
Digite el valor del numero 2:
> 5
El resultado de sumar 5 + 5 es: 10
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

6. Ejemplo06

En un hospital se requiere 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.



```
1 Proceso Ejemplo06
2
3 //En un hospital se requiere un algoritmo que solicite al usuario los siguientes datos:
4 //Nombre del paciente, edad, tipo de sangre, estatura y genero. Mostrar todos los datos al final.
5
6 //Declaracion de variables
7 Definir NombrePaciente como cadena;
8 Definir EdadPaciente como entero;
9 Definir TipoDeSangrePaciente como cadena;
10 Definir EstaturaPaciente como real;
11 Definir GeneroPaciente como cadena;
12
13 //Datos de entrada
14 escribir "Digite nombre paciente:";
15 leer NombrePaciente;
16 escribir "Digite edad del paciente:";
17 leer EdadPaciente;
18 escribir "Digite tipo de sangre del paciente:";
19 leer TipoDeSangrePaciente;
20 escribir "Digite estatura del paciente:";
21 leer EstaturaPaciente;
22 escribir "Digite genero del paciente:";
23 leer GeneroPaciente;
24
25 //SALIDA
26 Escribir "Nombre del paciente es: ", NombrePaciente;
27 Escribir "La edad del paciente es: ", EdadPaciente;
28 Escribir "El tipo de sangre del paciente es: ", TipoDeSangrePaciente;
29 Escribir "La estatura del paciente es: ", EstaturaPaciente, " mtrs";
30 Escribir "El genero del paciente es: ", GeneroPaciente;
31
32 FinProceso
33
```

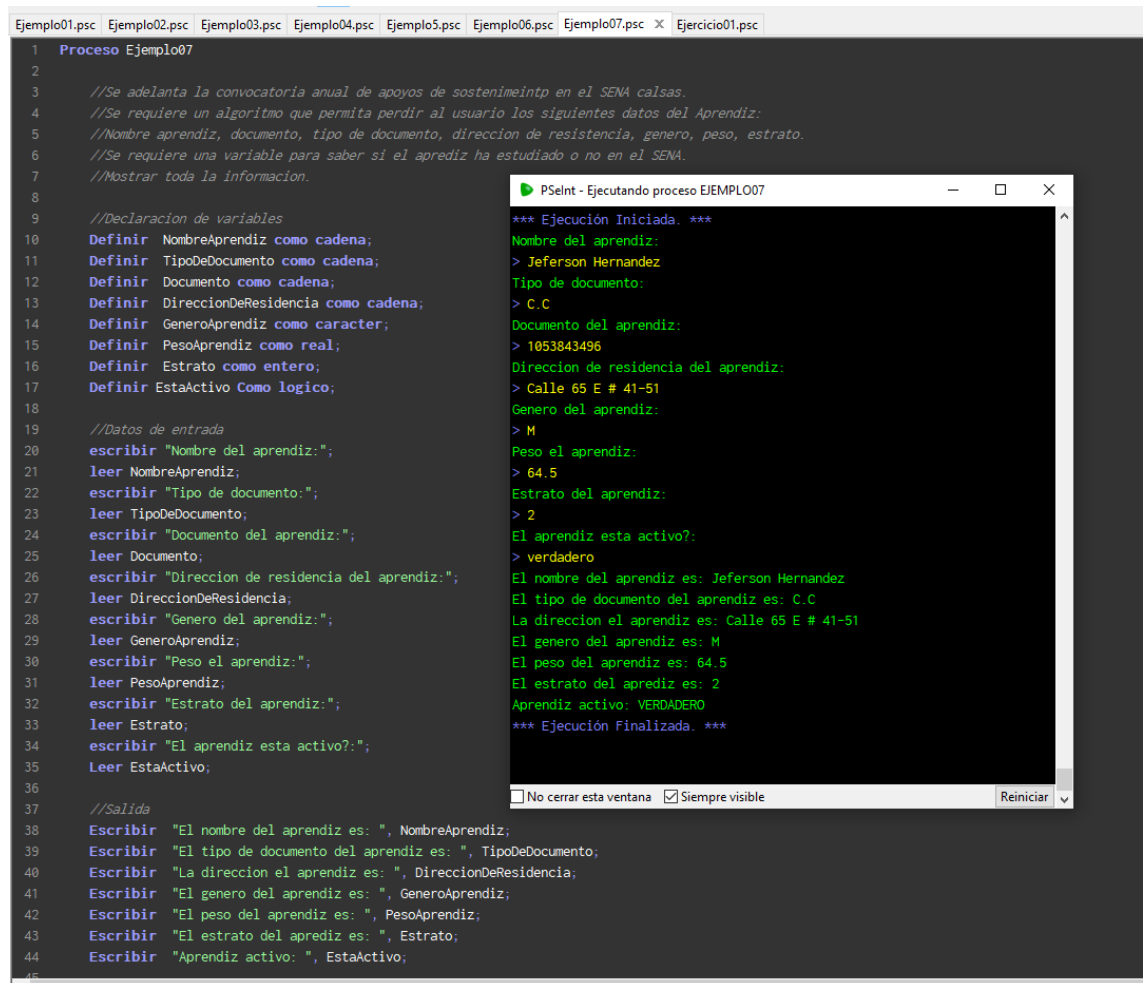
PSeInt - Ejecutando proceso EJEMPLO06

```
> 28
Digite tipo de sangre del paciente:
> 0+
Digite estatura del paciente:
> 1.67
Digite genero del paciente:
> Masculino
Nombre del paciente es: Jeferson Hernandez
La edad del paciente es: 28
El tipo de sangre del paciente es: 0+
La estatura del paciente es: 1.67 mtrs
El genero del paciente es: Masculino
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

7. Ejemplo07

En un hospital se requiere 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.

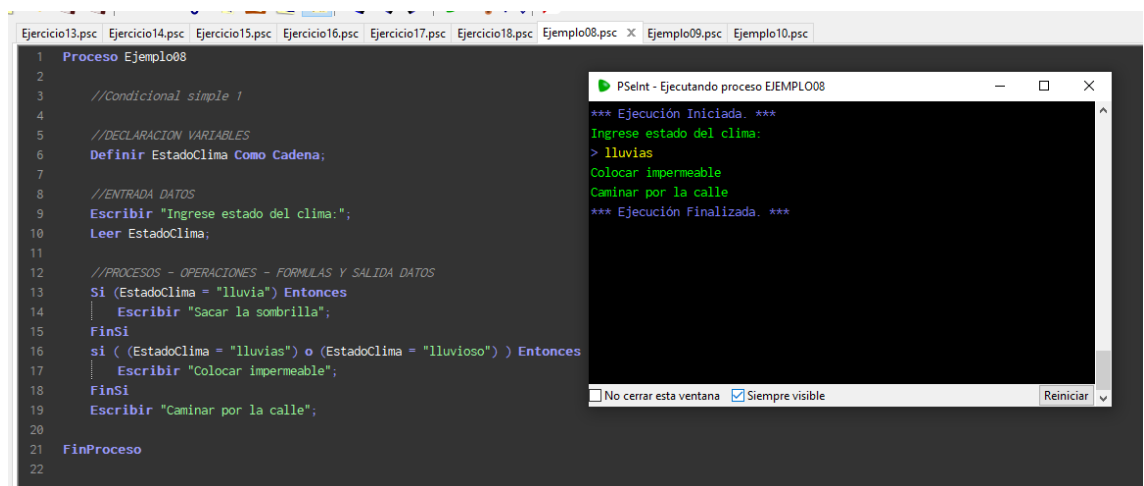


```
1 Proceso Ejemplo07
2
3 //Se adelanta la convocatoria anual de apoyos de sostenimeintp en el SENA calsas.
4 //Se requiere un algoritmo que permita pedir al usuario los siguientes datos del Aprendiz:
5 //Nombre aprendiz, documento, tipo de documento, direccion de residencia, genero, peso, estrato.
6 //Se requiere una variable para saber si el aprendiz ha estudiado o no en el SENA.
7 //Mostrar toda la informacion.
8
9 //Declaracion de variables
10 Definir NombreAprendiz como cadena;
11 Definir TipoDeDocumento como cadena;
12 Definir Documento como cadena;
13 Definir DireccionDeResidencia como cadena;
14 Definir GeneroAprendiz como caracter;
15 Definir PesoAprendiz como real;
16 Definir Estrato como entero;
17 Definir EstaActivo Como logico;
18
19 //Datos de entrada
20 escribir "Nombre del aprendiz:";
21 leer NombreAprendiz;
22 escribir "Tipo de documento:";
23 leer TipoDeDocumento;
24 escribir "Documento del aprendiz:";
25 leer Documento;
26 escribir "Direccion de residencia del aprendiz:";
27 leer DireccionDeResidencia;
28 escribir "Genero del aprendiz:";
29 leer GeneroAprendiz;
30 escribir "Peso el aprendiz:";
31 leer PesoAprendiz;
32 escribir "Estrato del aprendiz:";
33 leer Estrato;
34 escribir "El aprendiz esta activo?:";
35 Leer EstaActivo;
36
37 //Salida
38 Escribir "El nombre del aprendiz es: ", NombreAprendiz;
39 Escribir "El tipo de documento del aprendiz es: ", TipoDeDocumento;
40 Escribir "La direccion el aprendiz es: ", DireccionDeResidencia;
41 Escribir "El genero del aprendiz es: ", GeneroAprendiz;
42 Escribir "El peso del aprendiz es: ", PesoAprendiz;
43 Escribir "El estrato del aprendiz es: ", Estrato;
44 Escribir "Aprendiz activo: ", EstaActivo;
45
```

*** Ejecución Iniciada. ***
Nombre del aprendiz:
> Jefferson Hernandez
Tipo de documento:
> C.C
Documento del aprendiz:
> 1053843496
Direccion de residencia del aprendiz:
> Calle 65 E # 41-51
Genero del aprendiz:
> M
Peso el aprendiz:
> 64.5
Estrato del aprendiz:
> 2
El aprendiz esta activo?:
> verdadero
El nombre del aprendiz es: Jefferson Hernandez
El tipo de documento del aprendiz es: C.C
La direccion el aprendiz es: Calle 65 E # 41-51
El genero del aprendiz es: M
El peso del aprendiz es: 64.5
El estrato del aprendiz es: 2
Aprendiz activo: VERDADERO
*** Ejecución Finalizada. ***

8. Ejemplo08

Condicional simple 1

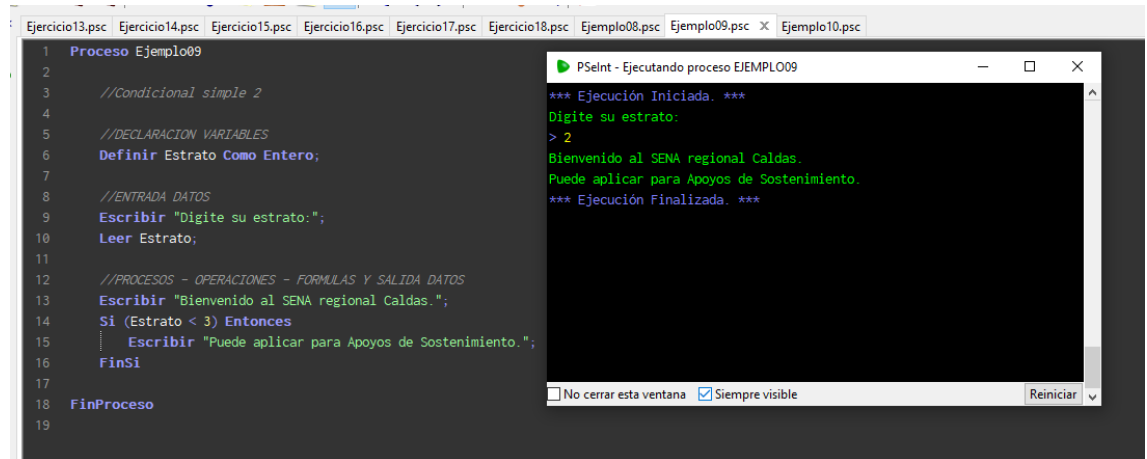


```
1 Proceso Ejemplo08
2
3 //Condicional simple 1
4
5 //DECLARACION VARIABLES
6 Definir EstadoClima Como Cadena;
7
8 //ENTRADA DATOS
9 Escribir "Ingrese estado del clima:";
10 Leer EstadoClima;
11
12 //PROCESOS - OPERACIONES - FORMULAS Y SALIDA DATOS
13 Si (EstadoClima = "lluvia") Entonces
14     Escribir "Sacar la sombrilla";
15 FinSi
16 Si ( (EstadoClima = "lluvias") o (EstadoClima = "llovioso") ) Entonces
17     Escribir "Colocar impermeable";
18 FinSi
19 Escribir "Caminar por la calle";
20
21 FinProceso
22
```

*** Ejecución Iniciada. ***
Ingrese estado del clima:
> lluvias
Colocar impermeable
Caminar por la calle
*** Ejecución Finalizada. ***

9. Ejemplo09

Condicional simple 2

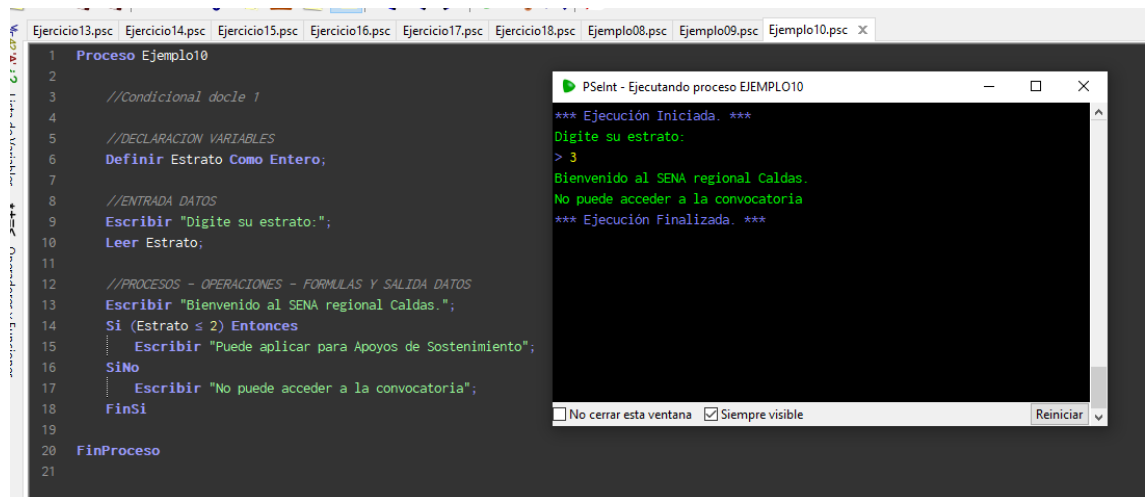


```
1 Proceso Ejemplo09
2
3 //Condicional simple 2
4
5 //DECLARACION VARIABLES
6 Definir Estrato Como Entero;
7
8 //ENTRADA DATOS
9 Escribir "Digite su estrato:";
10 Leer Estrato;
11
12 //PROCESOS - OPERACIONES - FORMULAS Y SALIDA DATOS
13 Escribir "Bienvenido al SENA regional Caldas.";
14 Si (Estrato < 3) Entonces
15     Escribir "Puede aplicar para Apoyos de Sostenimiento.";
16 FinSi
17
18 FinProceso
19
```

*** Ejecución Iniciada. ***
Digite su estrato:
> 2
Bienvenido al SENA regional Caldas.
Puede aplicar para Apoyos de Sostenimiento.
*** Ejecución Finalizada. ***

10. Ejemplo10

Condicional doble 1



```
1 Proceso Ejemplo10
2
3 //Condicional doble 1
4
5 //DECLARACION VARIABLES
6 Definir Estrato Como Entero;
7
8 //ENTRADA DATOS
9 Escribir "Digite su estrato:";
10 Leer Estrato;
11
12 //PROCESOS - OPERACIONES - FORMULAS Y SALIDA DATOS
13 Escribir "Bienvenido al SENA regional Caldas.";
14 Si (Estrato ≤ 2) Entonces
15     Escribir "Puede aplicar para Apoyos de Sostenimiento.";
16 SiNo
17     Escribir "No puede acceder a la convocatoria";
18 FinSi
19
20 FinProceso
21
```

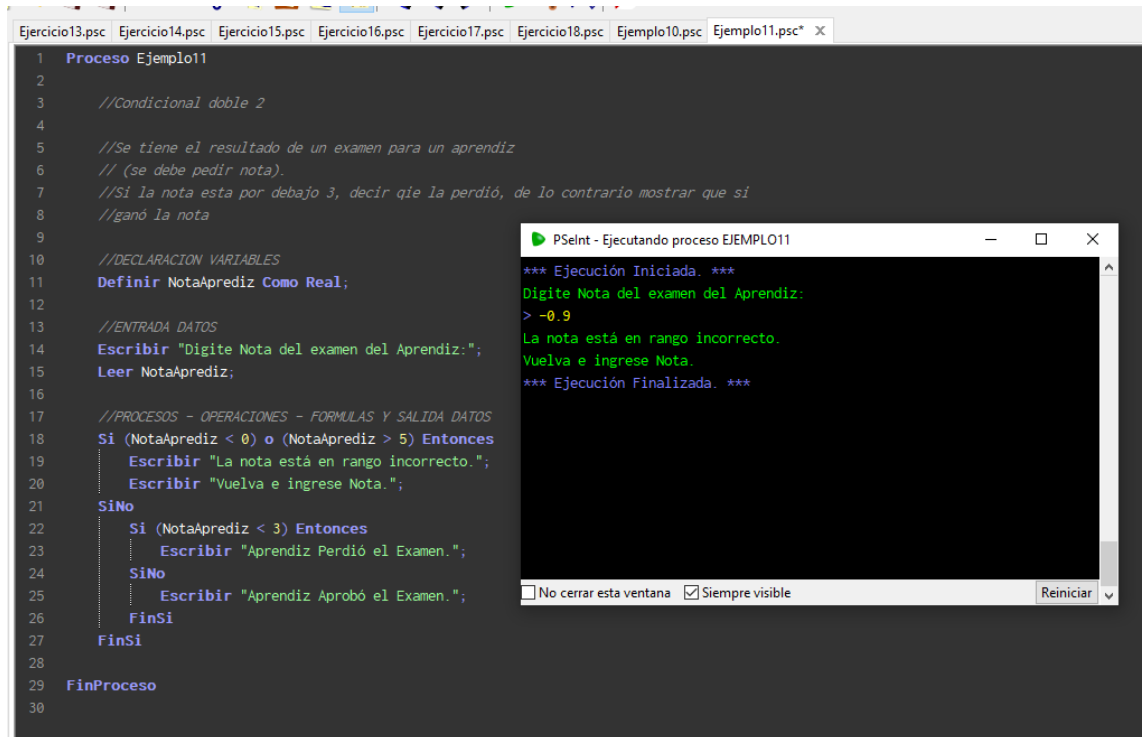
*** Ejecución Iniciada. ***
Digite su estrato:
> 3
Bienvenido al SENA regional Caldas.
No puede acceder a la convocatoria
*** Ejecución Finalizada. ***

11. Ejemplo11

Condicional doble 2. Condicional anidado.

Se tiene el resultado de un examen para un aprendiz (se debe pedir nota).

Si la nota está por debajo 3, decir que la perdió, de lo contrario mostrar que si ganó la nota



The screenshot displays the PSeInt IDE with a file named 'Ejemplo11.psc' open. The code is written in Pascal and implements a conditional logic for exam results. The execution window shows the program's output, including prompts for input and the resulting message based on the entered value.

```
1  Proceso Ejemplo11
2
3  //Condicional doble 2
4
5  //Se tiene el resultado de un examen para un aprendiz
6  // (se debe pedir nota).
7  //Si la nota esta por debajo 3, decir que la perdió, de lo contrario mostrar que si
8  //ganó la nota
9
10 //DECLARACION VARIABLES
11 Definir NotaAprendiz Como Real;
12
13 //ENTRADA DATOS
14 Escribir "Digite Nota del examen del Aprendiz:";
15 Leer NotaAprendiz;
16
17 //PROCESOS - OPERACIONES - FORMULAS Y SALIDA DATOS
18 Si (NotaAprendiz < 0) o (NotaAprendiz > 5) Entonces
19     Escribir "La nota está en rango incorrecto.";
20     Escribir "Vuelva e ingrese Nota.";
21 SiNo
22     Si (NotaAprendiz < 3) Entonces
23         Escribir "Aprendiz Perdió el Examen.";
24     SiNo
25         Escribir "Aprendiz Aprobó el Examen.";
26     FinSi
27 FinSi
28
29 FinProceso
30
```

Execution Output:

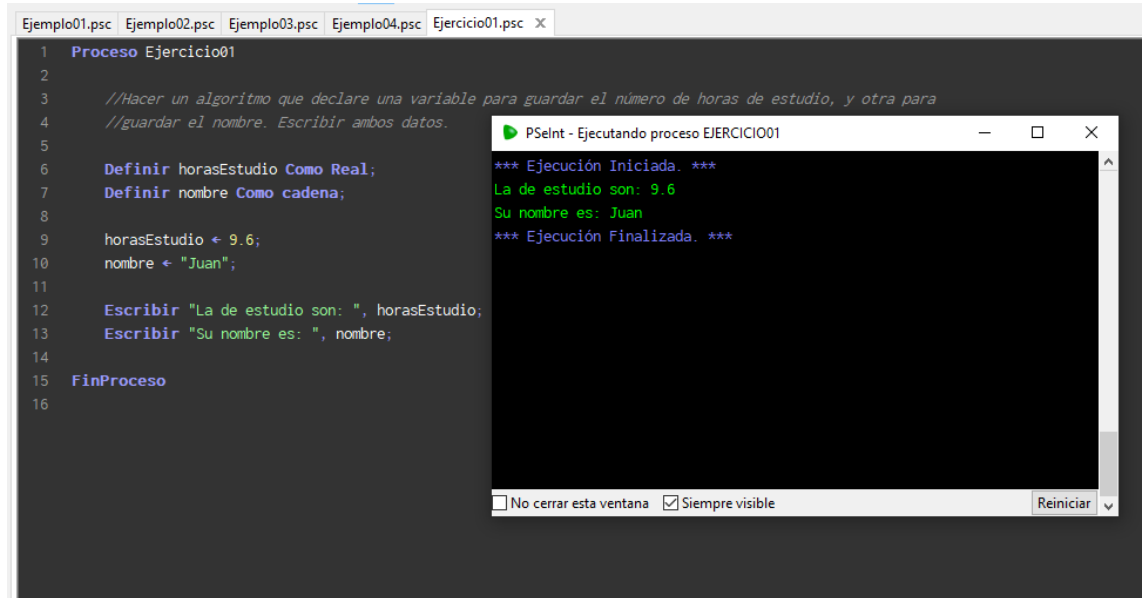
```
*** Ejecución Iniciada. ***
Digite Nota del examen del Aprendiz:
> -0.9
La nota está en rango incorrecto.
Vuelva e ingrese Nota.
*** Ejecución Finalizada. ***
```

Execution window controls: ☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

TALLER DE LA GUIA

1. Ejercicio01

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



```
1 Proceso Ejercicio01
2
3 //Hacer un algoritmo que declare una variable para guardar el número de horas de estudio, y otra para
4 //guardar el nombre. Escribir ambos datos.
5
6 Definir horasEstudio Como Real;
7 Definir nombre Como cadena;
8
9 horasEstudio ← 9.6;
10 nombre ← "Juan";
11
12 Escribir "La de estudio son: ", horasEstudio;
13 Escribir "Su nombre es: ", nombre;
14
15 FinProceso
16
```

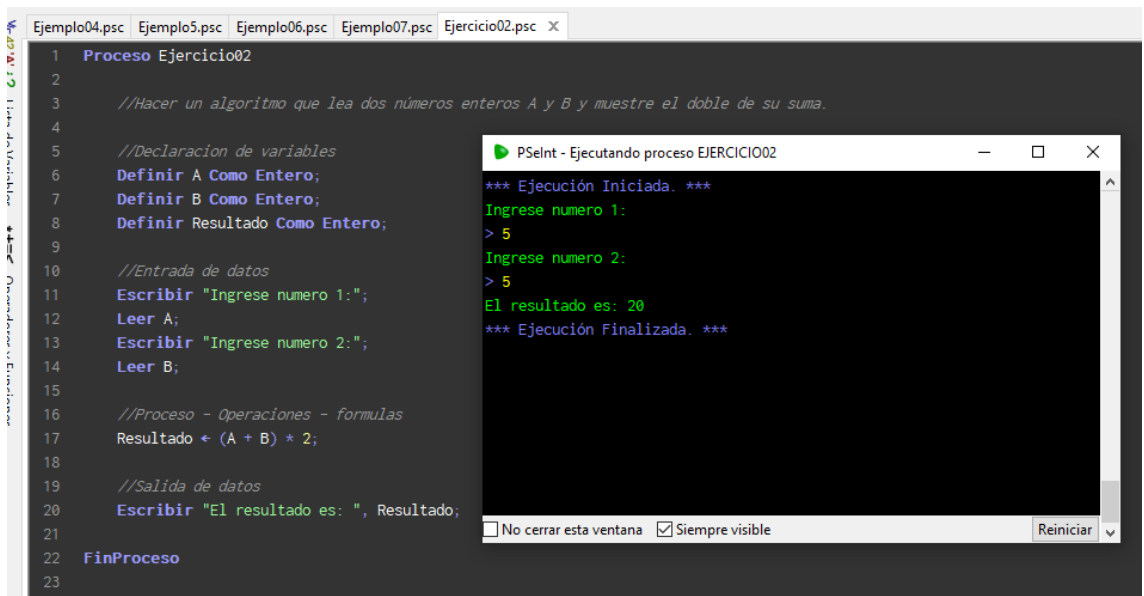
PSeInt - Ejecutando proceso EJERCICIO01

```
*** Ejecución Iniciada. ***
La de estudio son: 9.6
Su nombre es: Juan
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

2. Ejercicio02

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



```
1 Proceso Ejercicio02
2
3 //Hacer un algoritmo que lea dos números enteros A y B y muestre el doble de su suma.
4
5 //Declaracion de variables
6 Definir A Como Entero;
7 Definir B Como Entero;
8 Definir Resultado Como Entero;
9
10 //Entrada de datos
11 Escribir "Ingrese numero 1:";
12 Leer A;
13 Escribir "Ingrese numero 2:";
14 Leer B;
15
16 //Proceso - Operaciones - formulas
17 Resultado ← (A + B) * 2;
18
19 //Salida de datos
20 Escribir "El resultado es: ", Resultado;
21
22 FinProceso
23
```

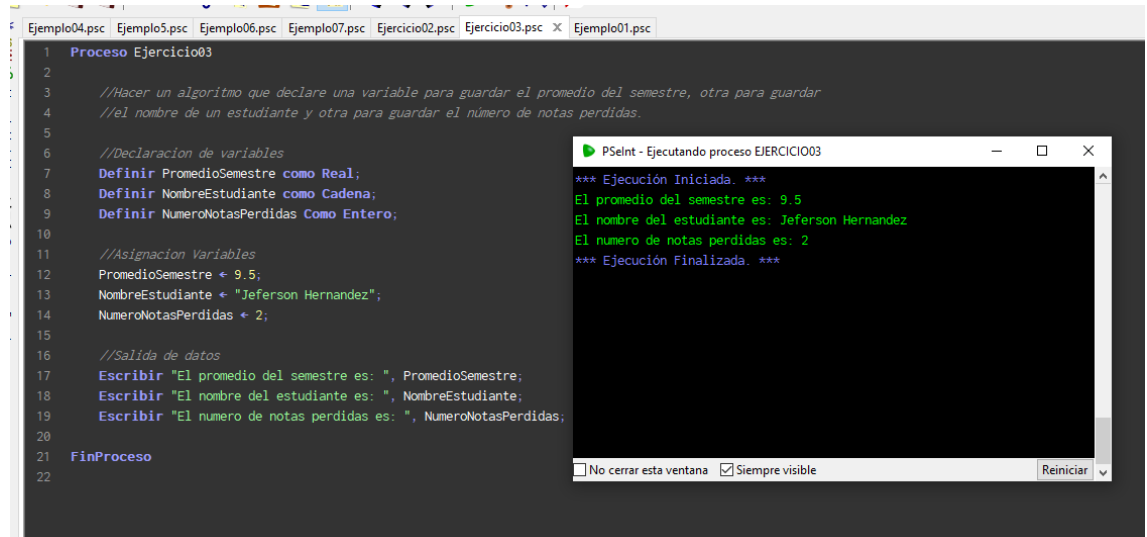
PSeInt - Ejecutando proceso EJERCICIO02

```
*** Ejecución Iniciada. ***
Ingrese numero 1:
> 5
Ingrese numero 2:
> 5
El resultado es: 20
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

3. Ejercicio03

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.

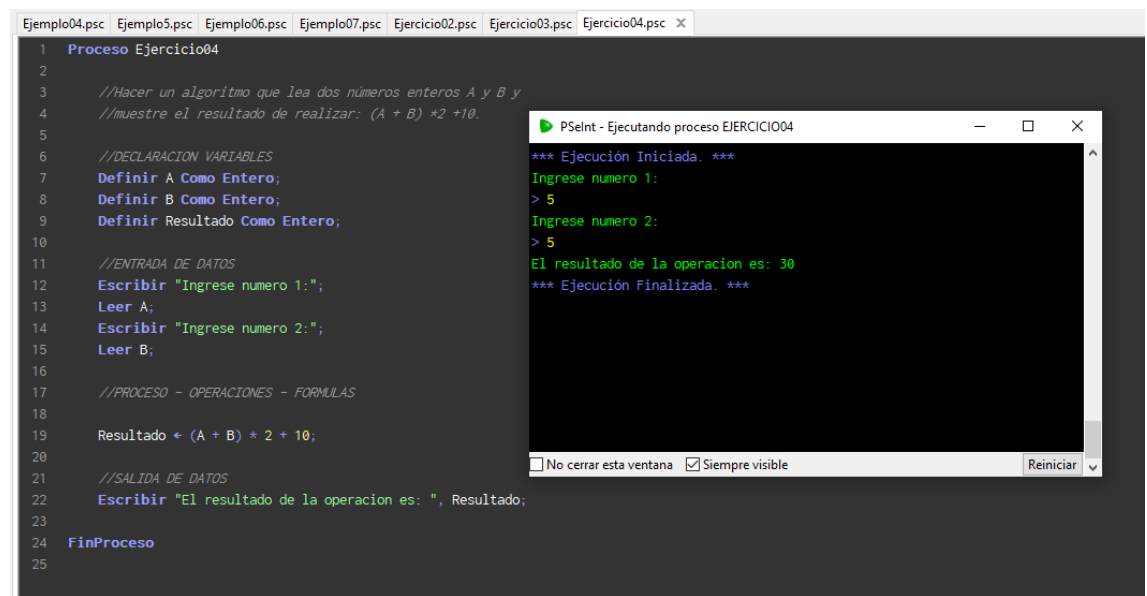


```
1 Proceso Ejercicio03
2
3 //Hacer un algoritmo que declare una variable para guardar el promedio del semestre, otra para guardar
4 //el nombre de un estudiante y otra para guardar el número de notas perdidas.
5
6 //Declaracion de variables
7 Definir PromedioSemestre como Real;
8 Definir NombreEstudiante como Cadena;
9 Definir NumeroNotasPerdidas Como Entero;
10
11 //Asignacion Variables
12 PromedioSemestre ← 9.5;
13 NombreEstudiante ← "Jeferson Hernandez";
14 NumeroNotasPerdidas ← 2;
15
16 //Salida de datos
17 Escribir "El promedio del semestre es: ", PromedioSemestre;
18 Escribir "El nombre del estudiante es: ", NombreEstudiante;
19 Escribir "El numero de notas perdidas es: ", NumeroNotasPerdidas;
20
21 FinProceso
22
```

*** Ejecución Iniciada. ***
El promedio del semestre es: 9.5
El nombre del estudiante es: Jeferson Hernandez
El numero de notas perdidas es: 2
*** Ejecución Finalizada. ***

4. Ejercicio04

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

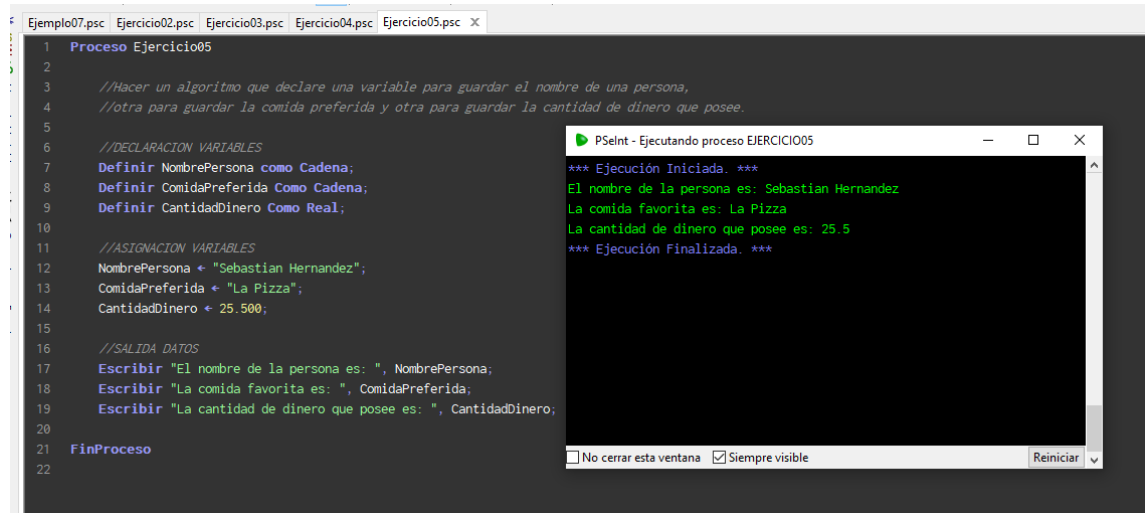


```
1 Proceso Ejercicio04
2
3 //Hacer un algoritmo que lea dos números enteros A y B y
4 //muestre el resultado de realizar: (A + B) * 2 + 10.
5
6 //DECLARACION VARIABLES
7 Definir A Como Entero;
8 Definir B Como Entero;
9 Definir Resultado Como Entero;
10
11 //ENTRADA DE DATOS
12 Escribir "Ingrese numero 1:";
13 Leer A;
14 Escribir "Ingrese numero 2:";
15 Leer B;
16
17 //PROCESO - OPERACIONES - FORMULAS
18
19 Resultado ← (A + B) * 2 + 10;
20
21 //SALIDA DE DATOS
22 Escribir "El resultado de la operacion es: ", Resultado;
23
24 FinProceso
25
```

*** Ejecución Iniciada. ***
Ingrese numero 1:
> 5
Ingrese numero 2:
> 5
El resultado de la operacion es: 30
*** Ejecución Finalizada. ***

5. Ejercicio05

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.



```
1 Proceso Ejercicio05
2
3 //Hacer un algoritmo que declare una variable para guardar el nombre de una persona,
4 //otra para guardar la comida preferida y otra para guardar la cantidad de dinero que posee.
5
6 //DECLARACION VARIABLES
7 Definir NombrePersona como Cadena;
8 Definir ComidaPreferida Como Cadena;
9 Definir CantidadDinero Como Real;
10
11 //ASIGNACION VARIABLES
12 NombrePersona ← "Sebastian Hernandez";
13 ComidaPreferida ← "La Pizza";
14 CantidadDinero ← 25.500;
15
16 //SALIDA DATOS
17 Escribir "El nombre de la persona es: ", NombrePersona;
18 Escribir "La comida favorita es: ", ComidaPreferida;
19 Escribir "La cantidad de dinero que posee es: ", CantidadDinero;
20
21 FinProceso
22
```

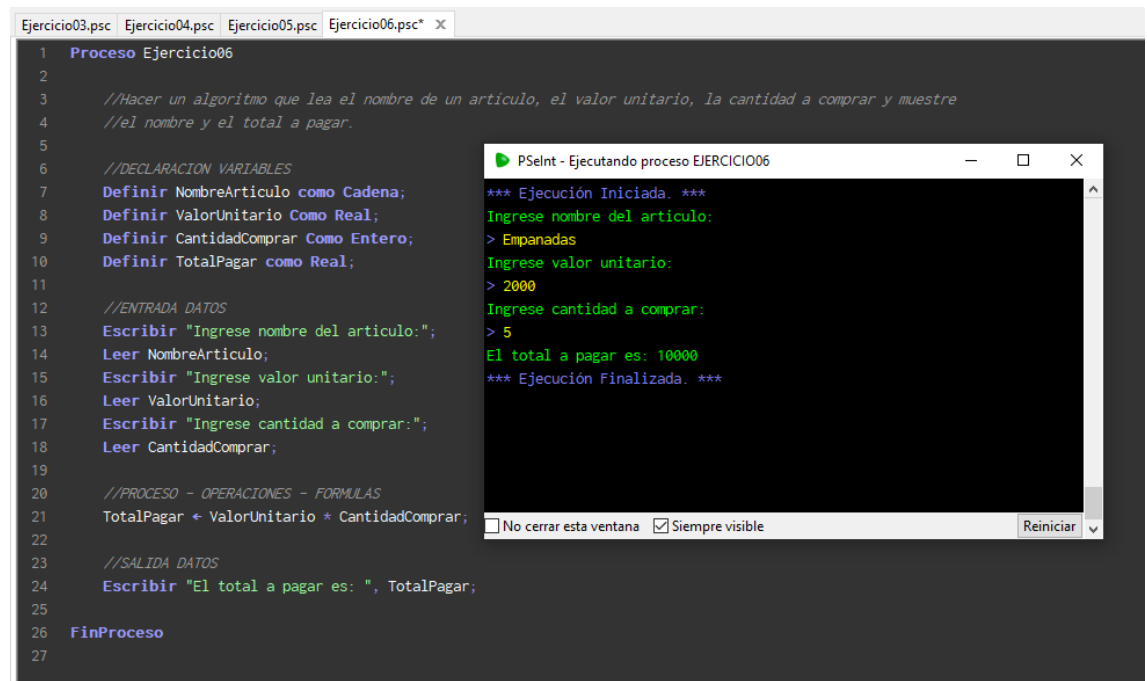
PSeInt - Ejecutando proceso EJERCICIO05

```
*** Ejecución Iniciada. ***
El nombre de la persona es: Sebastian Hernandez
La comida favorita es: La Pizza
La cantidad de dinero que posee es: 25.5
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

6. Ejercicio06

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 Ejercicio06
2
3 //Hacer un algoritmo que lea el nombre de un artículo, el valor unitario, la cantidad a comprar y muestre
4 //el nombre y el total a pagar.
5
6 //DECLARACION VARIABLES
7 Definir NombreArticulo como Cadena;
8 Definir ValorUnitario Como Real;
9 Definir CantidadComprar Como Entero;
10 Definir TotalPagar como Real;
11
12 //ENTRADA DATOS
13 Escribir "Ingrese nombre del articulo:";
14 Leer NombreArticulo;
15 Escribir "Ingrese valor unitario:";
16 Leer ValorUnitario;
17 Escribir "Ingrese cantidad a comprar:";
18 Leer CantidadComprar;
19
20 //PROCESO - OPERACIONES - FORMULAS
21 TotalPagar ← ValorUnitario * CantidadComprar;
22
23 //SALIDA DATOS
24 Escribir "El total a pagar es: ", TotalPagar;
25
26 FinProceso
27
```

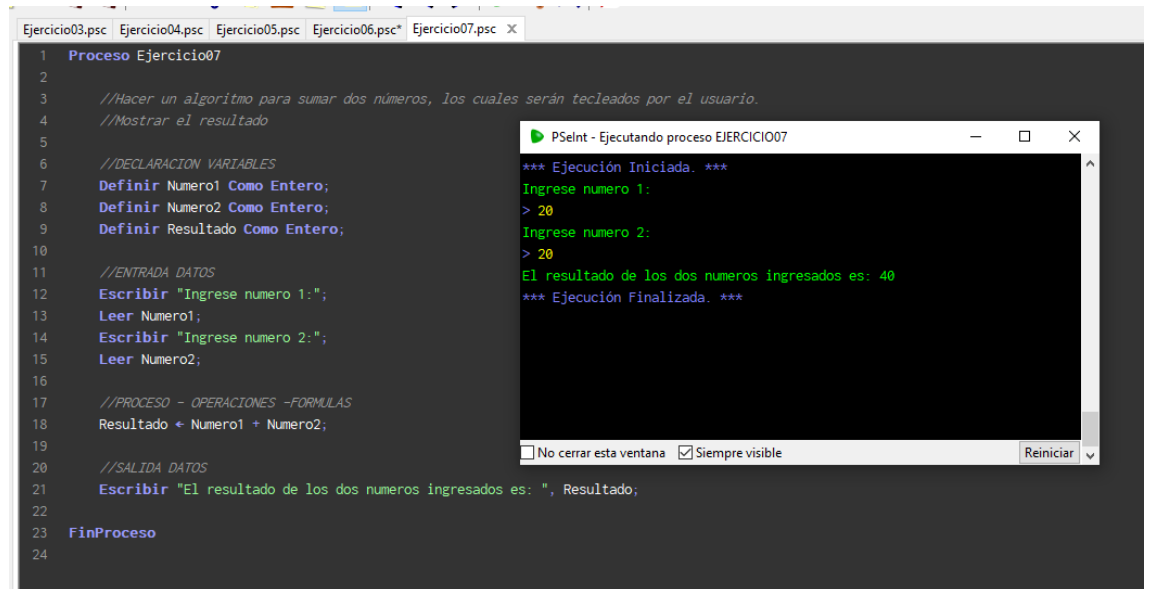
PSeInt - Ejecutando proceso EJERCICIO06

```
*** Ejecución Iniciada. ***
Ingrese nombre del articulo:
> Empanadas
Ingrese valor unitario:
> 2000
Ingrese cantidad a comprar:
> 5
El total a pagar es: 10000
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

7. Ejercicio07

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



The screenshot shows the PSeInt IDE with the code for Ejercicio07. The code is as follows:

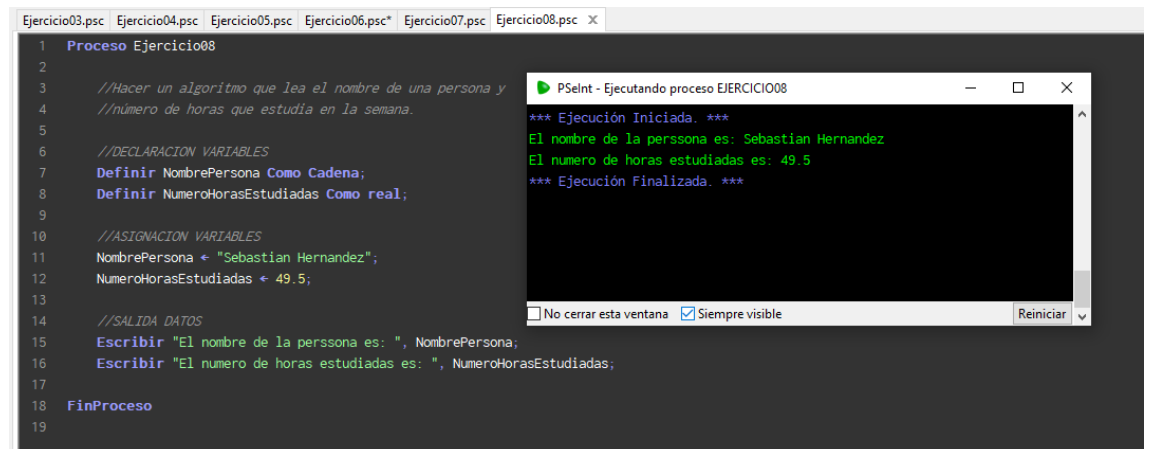
```
1 Proceso Ejercicio07
2
3 //Hacer un algoritmo para sumar dos números, los cuales serán tecleados por el usuario.
4 //Mostrar el resultado
5
6 //DECLARACION VARIABLES
7 Definir Numero1 Como Entero;
8 Definir Numero2 Como Entero;
9 Definir Resultado Como Entero;
10
11 //ENTRADA DATOS
12 Escribir "Ingrese numero 1:";
13 Leer Numero1;
14 Escribir "Ingrese numero 2:";
15 Leer Numero2;
16
17 //PROCESO - OPERACIONES - FORMULAS
18 Resultado ← Numero1 + Numero2;
19
20 //SALIDA DATOS
21 Escribir "El resultado de los dos numeros ingresados es: ", Resultado;
22
23 FinProceso
24
```

The execution output window shows the following text:

```
*** Ejecución Iniciada. ***
Ingrese numero 1:
> 20
Ingrese numero 2:
> 20
El resultado de los dos numeros ingresados es: 40
*** Ejecución Finalizada. ***
```

8. Ejercicio08

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 the code for Ejercicio08. The code is as follows:

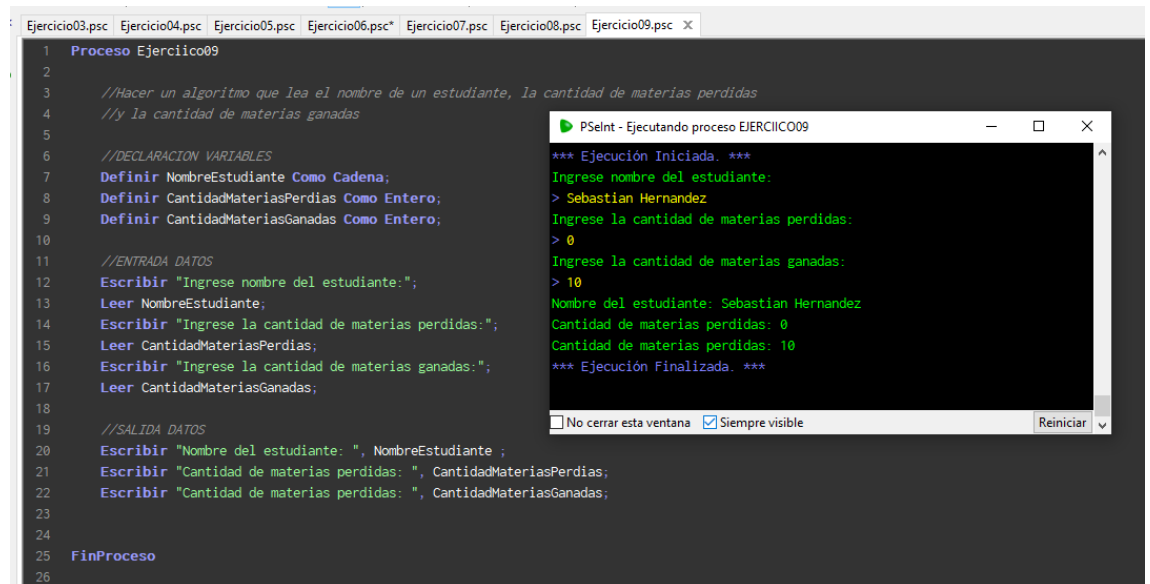
```
1 Proceso Ejercicio08
2
3 //Hacer un algoritmo que lea el nombre de una persona y
4 //número de horas que estudia en la semana.
5
6 //DECLARACION VARIABLES
7 Definir NombrePersona Como Cadena;
8 Definir NumeroHorasEstudiadas Como real;
9
10 //ASIGNACION VARIABLES
11 NombrePersona ← "Sebastian Hernandez";
12 NumeroHorasEstudiadas ← 49.5;
13
14 //SALIDA DATOS
15 Escribir "El nombre de la perssona es: ", NombrePersona;
16 Escribir "El numero de horas estudiadas es: ", NumeroHorasEstudiadas;
17
18 FinProceso
19
```

The execution output window shows the following text:

```
*** Ejecución Iniciada. ***
El nombre de la perssona es: Sebastian Hernandez
El numero de horas estudiadas es: 49.5
*** Ejecución Finalizada. ***
```

9. Ejercicio09

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



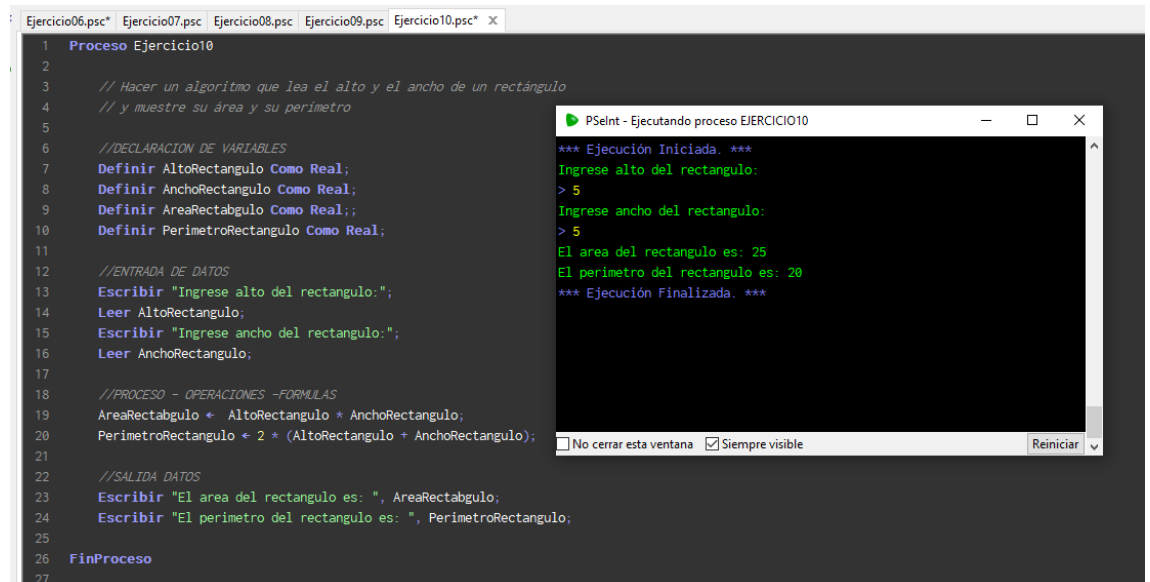
The screenshot shows a Pascal IDE with a file named 'Ejercicio09.psc' open. The code defines a process 'Proceso Ejercicio09' that reads a student's name, the number of failed subjects, and the number of passed subjects, then displays the results. A terminal window titled 'PSeInt - Ejecutando proceso EJERCICIO09' shows the execution with the input 'Sebastian Hernandez' and '10' for failed subjects, resulting in '0' for passed subjects.

```
1 Proceso Ejercicio09
2
3 //Hacer un algoritmo que lea el nombre de un estudiante, la cantidad de materias perdidas
4 //y la cantidad de materias ganadas
5
6 //DECLARACION VARIABLES
7 Definir NombreEstudiante Como Cadena;
8 Definir CantidadMateriasPerdias Como Entero;
9 Definir CantidadMateriasGanadas Como Entero;
10
11 //ENTRADA DATOS
12 Escribir "Ingrese nombre del estudiante:";
13 Leer NombreEstudiante;
14 Escribir "Ingrese la cantidad de materias perdidas:";
15 Leer CantidadMateriasPerdias;
16 Escribir "Ingrese la cantidad de materias ganadas:";
17 Leer CantidadMateriasGanadas;
18
19 //SALIDA DATOS
20 Escribir "Nombre del estudiante: ", NombreEstudiante ;
21 Escribir "Cantidad de materias perdidas: ", CantidadMateriasPerdias;
22 Escribir "Cantidad de materias ganadas: ", CantidadMateriasGanadas;
23
24
25 FinProceso
26
```

*** Ejecución Iniciada. ***
Ingrese nombre del estudiante:
> Sebastian Hernandez
Ingrese la cantidad de materias perdidas:
> 10
Ingrese la cantidad de materias ganadas:
> 0
Nombre del estudiante: Sebastian Hernandez
Cantidad de materias perdidas: 10
Cantidad de materias ganadas: 0
*** Ejecución Finalizada. ***

10. Ejercicio10

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



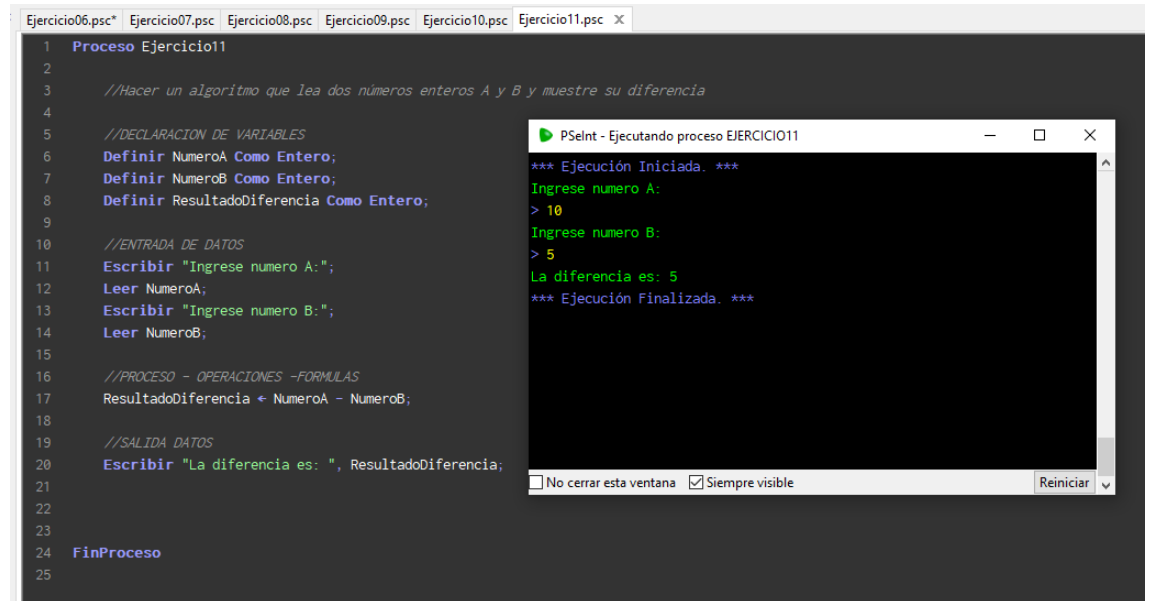
The screenshot shows a Pascal IDE with a file named 'Ejercicio10.psc' open. The code defines a process 'Proceso Ejercicio10' that reads the height and width of a rectangle, calculates its area and perimeter, and displays the results. A terminal window titled 'PSeInt - Ejecutando proceso EJERCICIO10' shows the execution with inputs '5' for height and '5' for width, resulting in an area of '25' and a perimeter of '20'.

```
1 Proceso Ejercicio10
2
3 // Hacer un algoritmo que lea el alto y el ancho de un rectángulo
4 // y muestre su área y su perímetro
5
6 //DECLARACION DE VARIABLES
7 Definir AltoRectangulo Como Real;
8 Definir AnchoRectangulo Como Real;
9 Definir AreaRectangulo Como Real;;
10 Definir PerimetroRectangulo Como Real;
11
12 //ENTRADA DE DATOS
13 Escribir "Ingrese alto del rectangulo:";
14 Leer AltoRectangulo;
15 Escribir "Ingrese ancho del rectangulo:";
16 Leer AnchoRectangulo;
17
18 //PROCESO - OPERACIONES -FORMULAS
19 AreaRectangulo ← AltoRectangulo * AnchoRectangulo;
20 PerimetroRectangulo ← 2 * (AltoRectangulo + AnchoRectangulo);
21
22 //SALIDA DATOS
23 Escribir "El area del rectangulo es: ", AreaRectangulo;
24 Escribir "El perimetro del rectangulo es: ", PerimetroRectangulo;
25
26 FinProceso
27
```

*** Ejecución Iniciada. ***
Ingrese alto del rectangulo:
> 5
Ingrese ancho del rectangulo:
> 5
El area del rectangulo es: 25
El perimetro del rectangulo es: 20
*** Ejecución Finalizada. ***

11. Ejercicio11

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

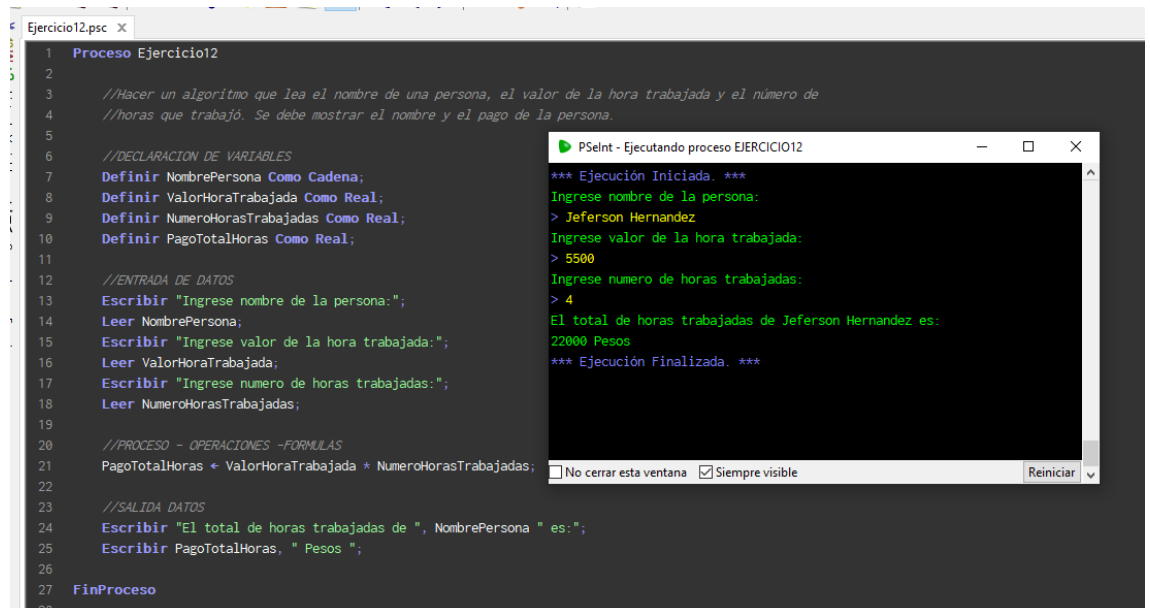


```
1 Proceso Ejercicio11
2
3 //Hacer un algoritmo que lea dos numeros enteros A y B y muestre su diferencia
4
5 //DECLARACION DE VARIABLES
6 Definir NumeroA Como Entero;
7 Definir NumeroB Como Entero;
8 Definir ResultadoDiferencia Como Entero;
9
10 //ENTRADA DE DATOS
11 Escribir "Ingrese numero A:";
12 Leer NumeroA;
13 Escribir "Ingrese numero B:";
14 Leer NumeroB;
15
16 //PROCESO - OPERACIONES -FORMULAS
17 ResultadoDiferencia ← NumeroA - NumeroB;
18
19 //SALIDA DATOS
20 Escribir "La diferencia es: ", ResultadoDiferencia;
21
22
23 FinProceso
24
25
```

*** Ejecución Iniciada. ***
Ingrese numero A:
> 10
Ingrese numero B:
> 5
La diferencia es: 5
*** Ejecución Finalizada. ***

12. Ejercicio12

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.

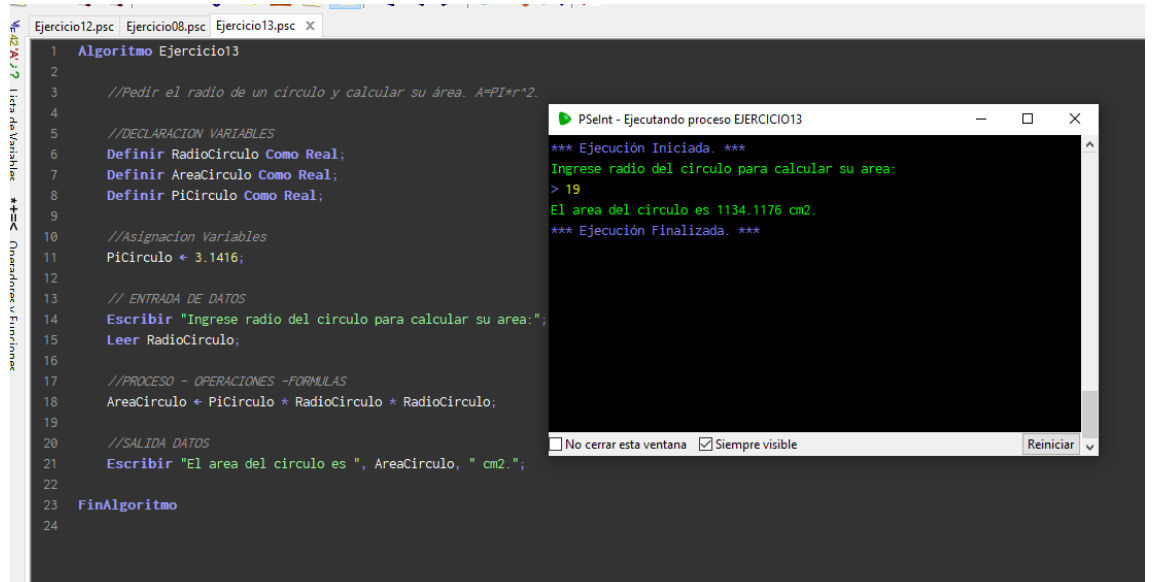


```
1 Proceso Ejercicio12
2
3 //Hacer un algoritmo que lea el nombre de una persona, el valor de la hora trabajada y el número de
4 //horas que trabajó. Se debe mostrar el nombre y el pago de la persona.
5
6 //DECLARACION DE VARIABLES
7 Definir NombrePersona Como Cadena;
8 Definir ValorHoraTrabajada Como Real;
9 Definir NumeroHorasTrabajadas Como Real;
10 Definir PagoTotalHoras Como Real;
11
12 //ENTRADA DE DATOS
13 Escribir "Ingrese nombre de la persona:";
14 Leer NombrePersona;
15 Escribir "Ingrese valor de la hora trabajada:";
16 Leer ValorHoraTrabajada;
17 Escribir "Ingrese numero de horas trabajadas:";
18 Leer NumeroHorasTrabajadas;
19
20 //PROCESO - OPERACIONES -FORMULAS
21 PagoTotalHoras ← ValorHoraTrabajada * NumeroHorasTrabajadas;
22
23 //SALIDA DATOS
24 Escribir "El total de horas trabajadas de ", NombrePersona " es:";
25 Escribir PagoTotalHoras, " Pesos ";
26
27 FinProceso
28
```

*** Ejecución Iniciada. ***
Ingrese nombre de la persona:
> Jeferson Hernandez
Ingrese valor de la hora trabajada:
> 5500
Ingrese numero de horas trabajadas:
> 4
El total de horas trabajadas de Jeferson Hernandez es:
22000 Pesos
*** Ejecución Finalizada. ***

13. Ejercicio13

Pedir el radio de un círculo y calcular su área. $A=PI*r^2$.

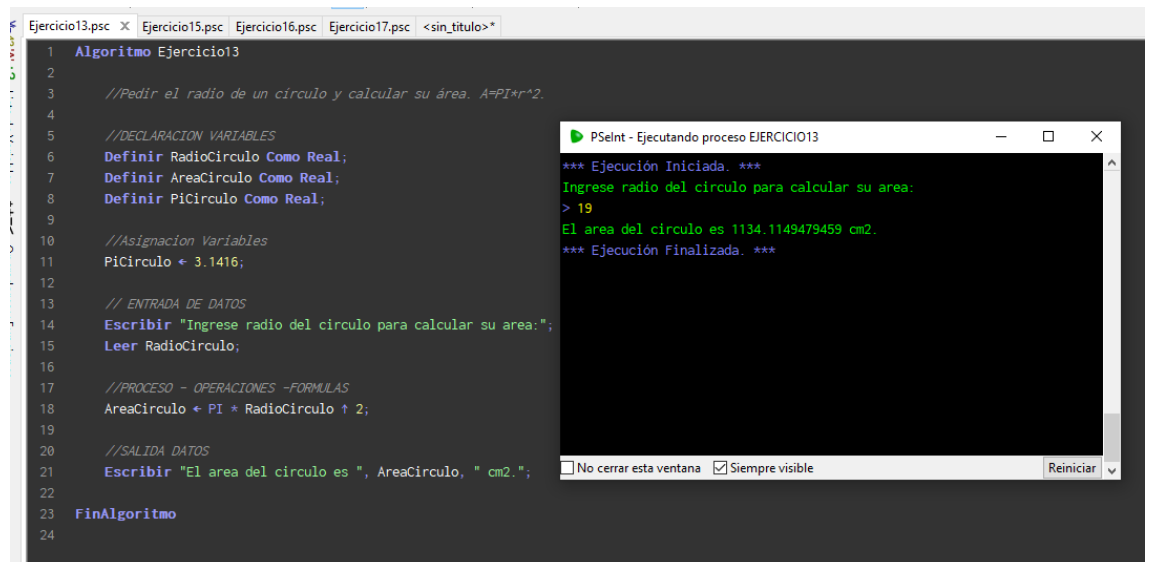


```
1  Algoritmo Ejercicio13
2
3  //Pedir el radio de un círculo y calcular su área. A=PI*r^2.
4
5  //DECLARACION VARIABLES
6  Definir RadioCirculo Como Real;
7  Definir AreaCirculo Como Real;
8  Definir PiCirculo Como Real;
9
10 //Asignacion Variables
11 PiCirculo ← 3.1416;
12
13 // ENTRADA DE DATOS
14 Escribir "Ingrese radio del círculo para calcular su area:";
15 Leer RadioCirculo;
16
17 //PROCESO - OPERACIONES -FORMULAS
18 AreaCirculo ← PiCirculo * RadioCirculo * RadioCirculo;
19
20 //SALIDA DATOS
21 Escribir "El area del círculo es ", AreaCirculo, " cm2.";
22
23 FinAlgoritmo
24
```

PSeInt - Ejecutando proceso EJERCICIO13

```
*** Ejecución Iniciada. ***
Ingrese radio del círculo para calcular su area:
> 19
El area del círculo es 1134.1176 cm2.
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar



```
1  Algoritmo Ejercicio13
2
3  //Pedir el radio de un círculo y calcular su área. A=PI*r^2.
4
5  //DECLARACION VARIABLES
6  Definir RadioCirculo Como Real;
7  Definir AreaCirculo Como Real;
8  Definir PiCirculo Como Real;
9
10 //Asignacion Variables
11 PiCirculo ← 3.1416;
12
13 // ENTRADA DE DATOS
14 Escribir "Ingrese radio del círculo para calcular su area:";
15 Leer RadioCirculo;
16
17 //PROCESO - OPERACIONES -FORMULAS
18 AreaCirculo ← PI * RadioCirculo ↑ 2;
19
20 //SALIDA DATOS
21 Escribir "El area del círculo es ", AreaCirculo, " cm2.";
22
23 FinAlgoritmo
24
```

PSeInt - Ejecutando proceso EJERCICIO13

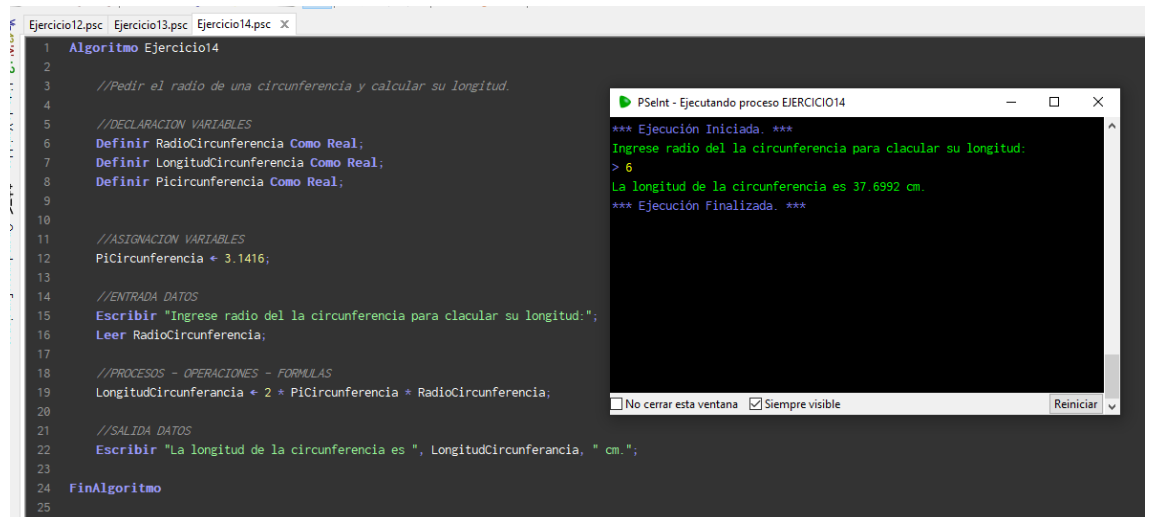
```
*** Ejecución Iniciada. ***
Ingrese radio del círculo para calcular su area:
> 19
El area del círculo es 1134.1149479459 cm2.
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

<https://www.mundodeportivo.com/uncomo/educacion/articulo/como-sacar-el-area-de-un-circulo-37105.html>

14. Ejercicio14

Pedir el radio de una circunferencia y calcular su longitud.



The screenshot shows a Pascal program named 'Ejercicio14.psc' and its execution in PSeInt. The program prompts the user to enter the radius of a circle and calculates its circumference. The execution window shows the user inputting '6' and the program outputting the circumference as '37.6992 cm'.

```
1 Algoritmo Ejercicio14
2
3 //Pedir el radio de una circunferencia y calcular su longitud.
4
5 //DECLARACION VARIABLES
6 Definir RadioCircunferencia Como Real;
7 Definir LongitudCircunferencia Como Real;
8 Definir PiCircunferencia Como Real;
9
10
11 //ASIGNACION VARIABLES
12 PiCircunferencia ← 3.1416;
13
14 //ENTRADA DATOS
15 Escribir "Ingrese radio del la circunferencia para clacular su longitud:";
16 Leer RadioCircunferencia;
17
18 //PROCESOS - OPERACIONES - FORMULAS
19 LongitudCircunferencia ← 2 * PiCircunferencia * RadioCircunferencia;
20
21 //SALIDA DATOS
22 Escribir "La longitud de la circunferencia es ", LongitudCircunferencia, " cm.";
23
24 FinAlgoritmo
25
```

PSeInt - Ejecutando proceso EJERCICIO14

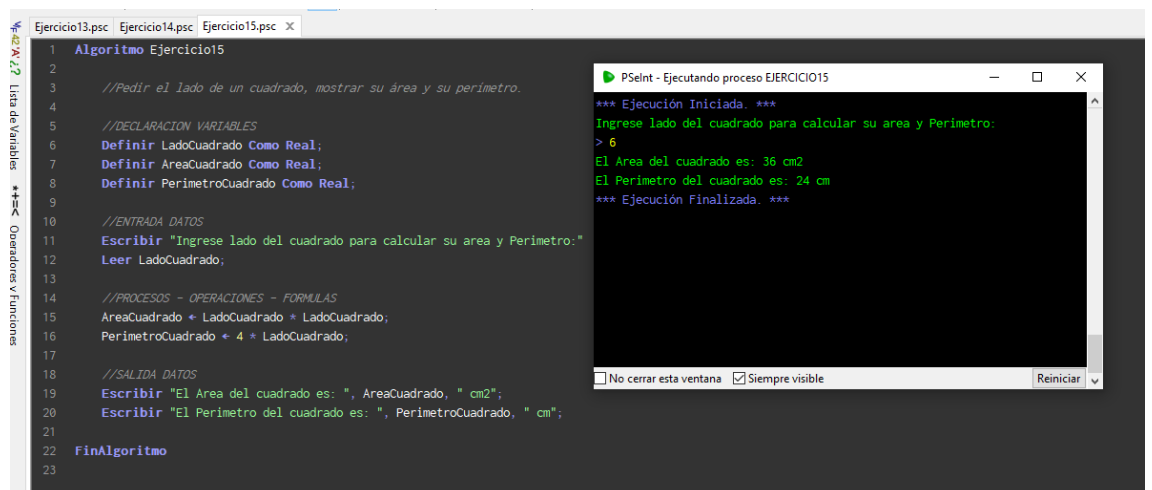
```
*** Ejecución Iniciada. ***
Ingrese radio del la circunferencia para clacular su longitud:
> 6
La longitud de la circunferencia es 37.6992 cm.
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

<https://www.mundodeportivo.com/uncomo/educacion/articulo/como-calculer-el-perimetro-de-una-circunferencia-40007.html>

15. Ejercicio15

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



The screenshot shows a Pascal program named 'Ejercicio15.psc' and its execution in PSeInt. The program prompts the user to enter the side length of a square and calculates its area and perimeter. The execution window shows the user inputting '6' and the program outputting the area as '36 cm2' and the perimeter as '24 cm'.

```
1 Algoritmo Ejercicio15
2
3 //Pedir el lado de un cuadrado, mostrar su área y su perimetro.
4
5 //DECLARACION VARIABLES
6 Definir LadoCuadrado Como Real;
7 Definir AreaCuadrado Como Real;
8 Definir PerimetroCuadrado Como Real;
9
10
11 //ENTRADA DATOS
12 Escribir "Ingrese lado del cuadrado para calcular su area y Perimetro:";
13 Leer LadoCuadrado;
14
15 //PROCESOS - OPERACIONES - FORMULAS
16 AreaCuadrado ← LadoCuadrado * LadoCuadrado;
17 PerimetroCuadrado ← 4 * LadoCuadrado;
18
19 //SALIDA DATOS
20 Escribir "El Area del cuadrado es: ", AreaCuadrado, " cm2";
21 Escribir "El Perimetro del cuadrado es: ", PerimetroCuadrado, " cm";
22
23 FinAlgoritmo
24
```

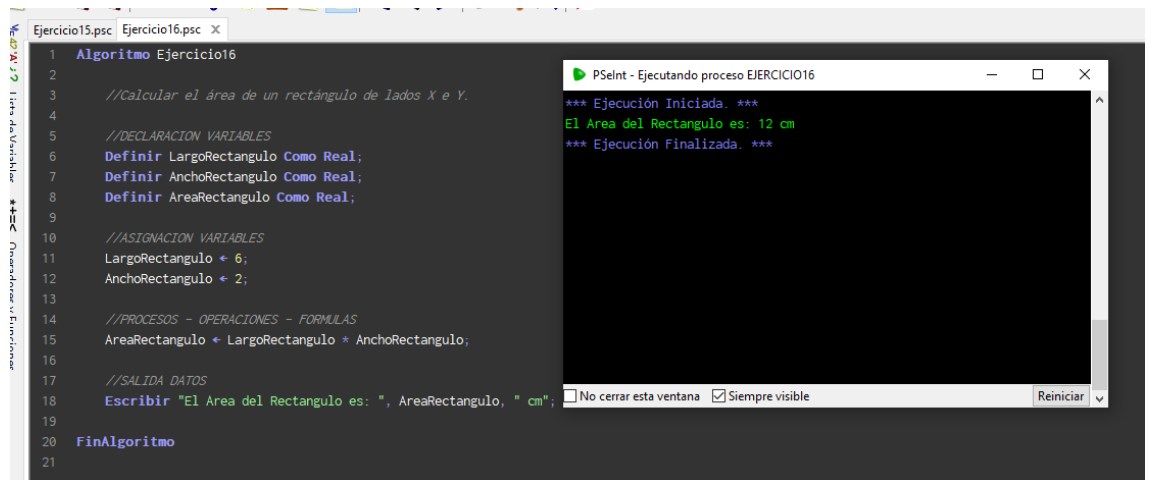
PSeInt - Ejecutando proceso EJERCICIO15

```
*** Ejecución Iniciada. ***
Ingrese lado del cuadrado para calcular su area y Perimetro:
> 6
El Area del cuadrado es: 36 cm2
El Perimetro del cuadrado es: 24 cm
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

16. Ejercicio16

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



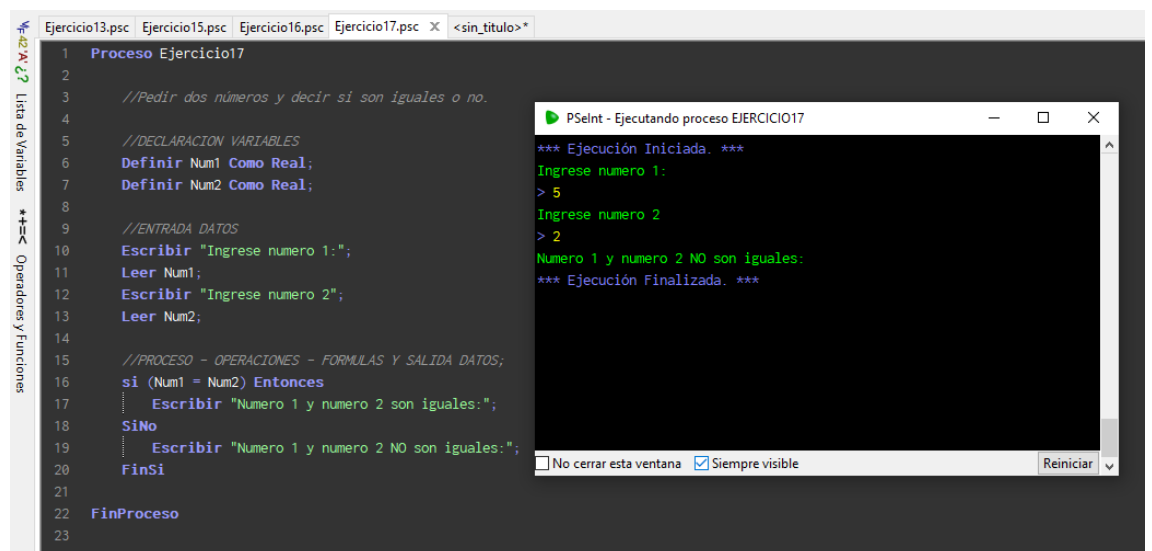
The screenshot shows a Pascal IDE with two windows. The main window displays the code for 'Algoritmo Ejercicio16'. The code calculates the area of a rectangle with length 6 and width 2, resulting in an area of 12 cm. The output window, titled 'PSeInt - Ejecutando proceso EJERCICIO16', shows the execution steps: '*** Ejecución Iniciada. ***', 'El Area del Rectangulo es: 12 cm', and '*** Ejecución Finalizada. ***'. The IDE interface includes a sidebar with 'Lista de Variables' and 'Operadores y Funciones'.

```
1 Algoritmo Ejercicio16
2
3 //Calcular el área de un rectángulo de lados X e Y.
4
5 //DECLARACION VARIABLES
6 Definir LargoRectangulo Como Real;
7 Definir AnchoRectangulo Como Real;
8 Definir AreaRectangulo Como Real;
9
10 //ASIGNACION VARIABLES
11 LargoRectangulo ← 6;
12 AnchoRectangulo ← 2;
13
14 //PROCESOS - OPERACIONES - FORMULAS
15 AreaRectangulo ← LargoRectangulo * AnchoRectangulo;
16
17 //SALIDA DATOS
18 Escribir "El Area del Rectangulo es: ", AreaRectangulo, " cm";
19
20 FinAlgoritmo
21
```

*** Ejecución Iniciada. ***
El Area del Rectangulo es: 12 cm
*** Ejecución Finalizada. ***

17. Ejercicio17

Pedir dos números y decir si son iguales o no.



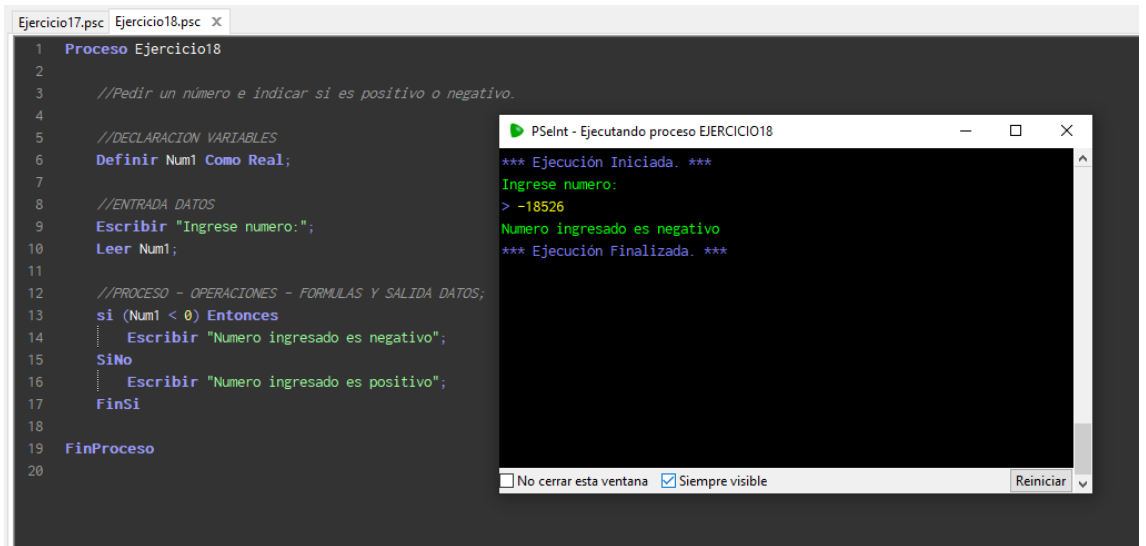
The screenshot shows a Pascal IDE with two windows. The main window displays the code for 'Proceso Ejercicio17'. The code prompts the user to enter two numbers, 5 and 2, and then checks if they are equal. Since they are not equal, it outputs 'Numero 1 y numero 2 NO son iguales:'. The output window, titled 'PSeInt - Ejecutando proceso EJERCICIO17', shows the execution steps: '*** Ejecución Iniciada. ***', 'Ingrese numero 1:', '> 5', 'Ingrese numero 2', '> 2', 'Numero 1 y numero 2 NO son iguales:', and '*** Ejecución Finalizada. ***'. The IDE interface includes a sidebar with 'Lista de Variables' and 'Operadores y Funciones'.

```
1 Proceso Ejercicio17
2
3 //Pedir dos números y decir si son iguales o no.
4
5 //DECLARACION VARIABLES
6 Definir Num1 Como Real;
7 Definir Num2 Como Real;
8
9 //ENTRADA DATOS
10 Escribir "Ingrese numero 1:";
11 Leer Num1;
12 Escribir "Ingrese numero 2:";
13 Leer Num2;
14
15 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
16 si (Num1 = Num2) Entonces
17     Escribir "Numero 1 y numero 2 son iguales:";
18 SiNo
19     Escribir "Numero 1 y numero 2 NO son iguales:";
20 FinSi
21
22 FinProceso
23
```

*** Ejecución Iniciada. ***
Ingrese numero 1:
> 5
Ingrese numero 2
> 2
Numero 1 y numero 2 NO son iguales:
*** Ejecución Finalizada. ***

18. Ejercicio18

Pedir un número e indicar si es positivo o negativo.



The screenshot shows the PSeInt IDE with two windows. The main window displays the code for 'Proceso Ejercicio18'. The code includes comments for input, variable declaration, data entry, and a conditional process to check if a number is positive or negative. The execution window shows the output of the program, including the prompt 'Ingrese numero:', the input '-18526', and the resulting message 'Numero ingresado es negativo'.

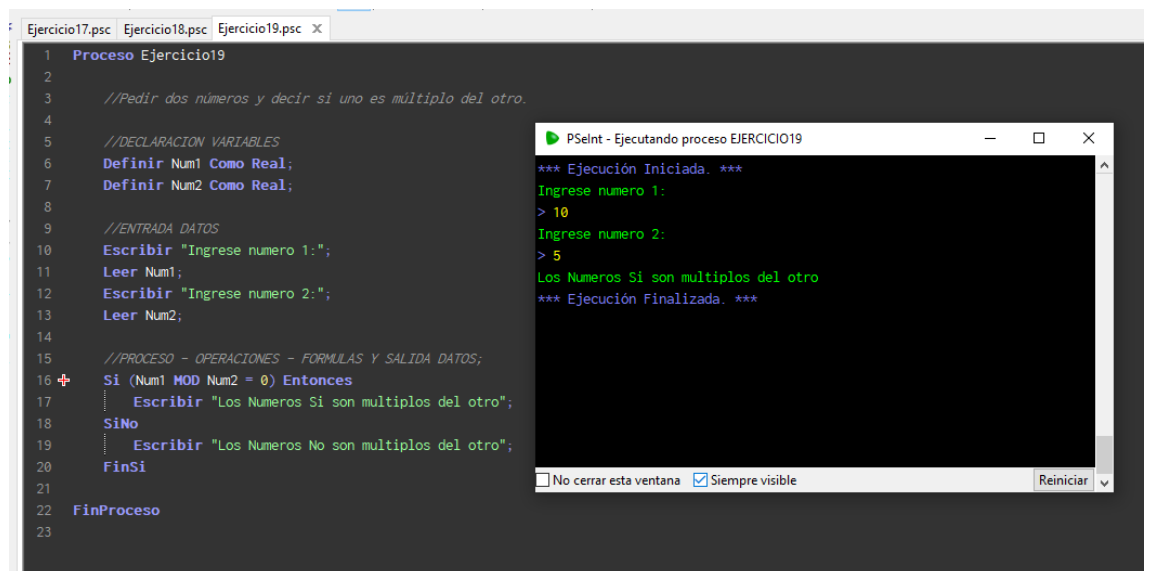
```
1 Proceso Ejercicio18
2
3 //Pedir un número e indicar si es positivo o negativo.
4
5 //DECLARACION VARIABLES
6 Definir Num1 Como Real;
7
8 //ENTRADA DATOS
9 Escribir "Ingrese numero:";
10 Leer Num1;
11
12 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
13 si (Num1 < 0) Entonces
14     Escribir "Numero ingresado es negativo";
15 SiNo
16     Escribir "Numero ingresado es positivo";
17 FinSi
18
19 FinProceso
20
```

Execution Output:

```
*** Ejecución Iniciada. ***
Ingrese numero:
> -18526
Numero ingresado es negativo
*** Ejecución Finalizada. ***
```

19. Ejercicio19

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



The screenshot shows the PSeInt IDE with two windows. The main window displays the code for 'Proceso Ejercicio19'. The code includes comments for input, variable declaration, data entry, and a conditional process to check if one number is a multiple of the other using the modulo operator. The execution window shows the output of the program, including prompts for two numbers, the inputs '10' and '5', and the resulting message 'Los Numeros Si son multiplos del otro'.

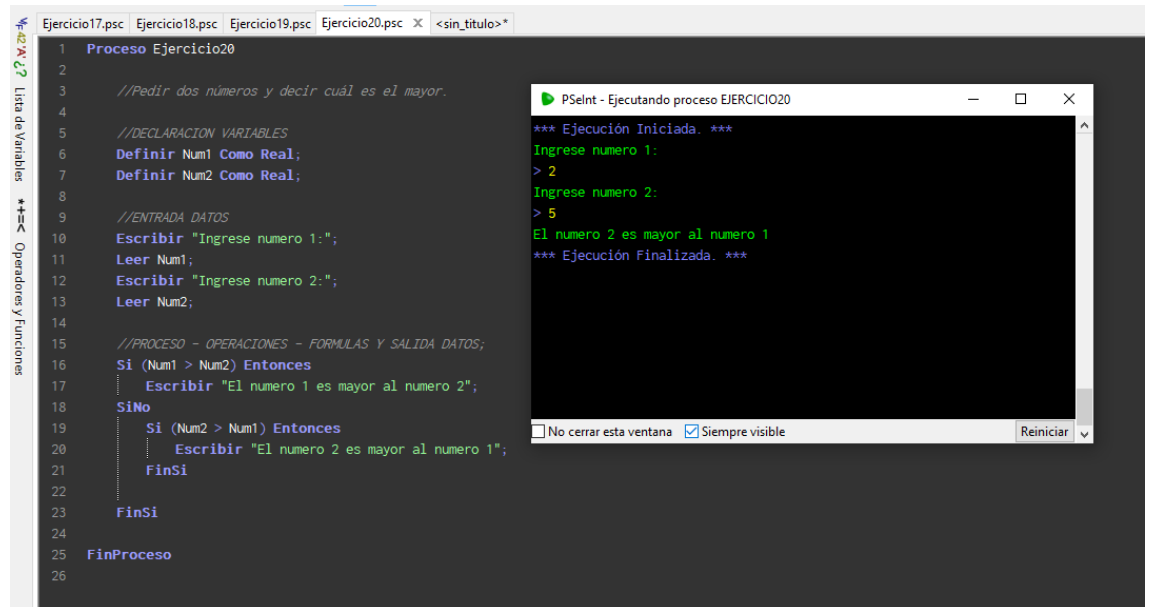
```
1 Proceso Ejercicio19
2
3 //Pedir dos números y decir si uno es múltiplo del otro.
4
5 //DECLARACION VARIABLES
6 Definir Num1 Como Real;
7 Definir Num2 Como Real;
8
9 //ENTRADA DATOS
10 Escribir "Ingrese numero 1:";
11 Leer Num1;
12 Escribir "Ingrese numero 2:";
13 Leer Num2;
14
15 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
16 Si (Num1 MOD Num2 = 0) Entonces
17     Escribir "Los Numeros Si son multiplos del otro";
18 SiNo
19     Escribir "Los Numeros No son multiplos del otro";
20 FinSi
21
22 FinProceso
23
```

Execution Output:

```
*** Ejecución Iniciada. ***
Ingrese numero 1:
> 10
Ingrese numero 2:
> 5
Los Numeros Si son multiplos del otro
*** Ejecución Finalizada. ***
```


20. Ejercicio20

Pedir dos números y decir cuál es el mayor.

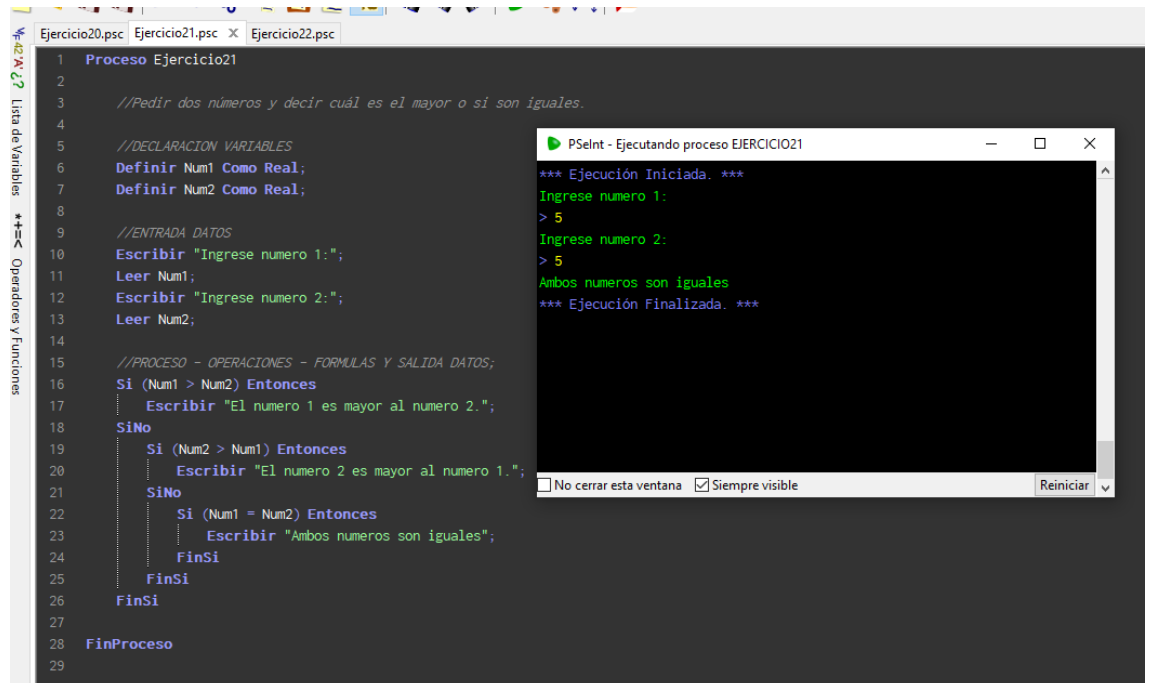


```
1 Proceso Ejercicio20
2
3 //Pedir dos números y decir cuál es el mayor.
4
5 //DECLARACION VARIABLES
6 Definir Num1 Como Real;
7 Definir Num2 Como Real;
8
9 //ENTRADA DATOS
10 Escribir "Ingrese numero 1:";
11 Leer Num1;
12 Escribir "Ingrese numero 2:";
13 Leer Num2;
14
15 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
16 Si (Num1 > Num2) Entonces
17     Escribir "El numero 1 es mayor al numero 2";
18 SiNo
19     Si (Num2 > Num1) Entonces
20         Escribir "El numero 2 es mayor al numero 1";
21     FinSi
22 FinSi
23
24 FinProceso
25
26
```

*** Ejecución Iniciada. ***
Ingrese numero 1:
> 2
Ingrese numero 2:
> 5
El numero 2 es mayor al numero 1
*** Ejecución Finalizada. ***

21. Ejercicio21

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

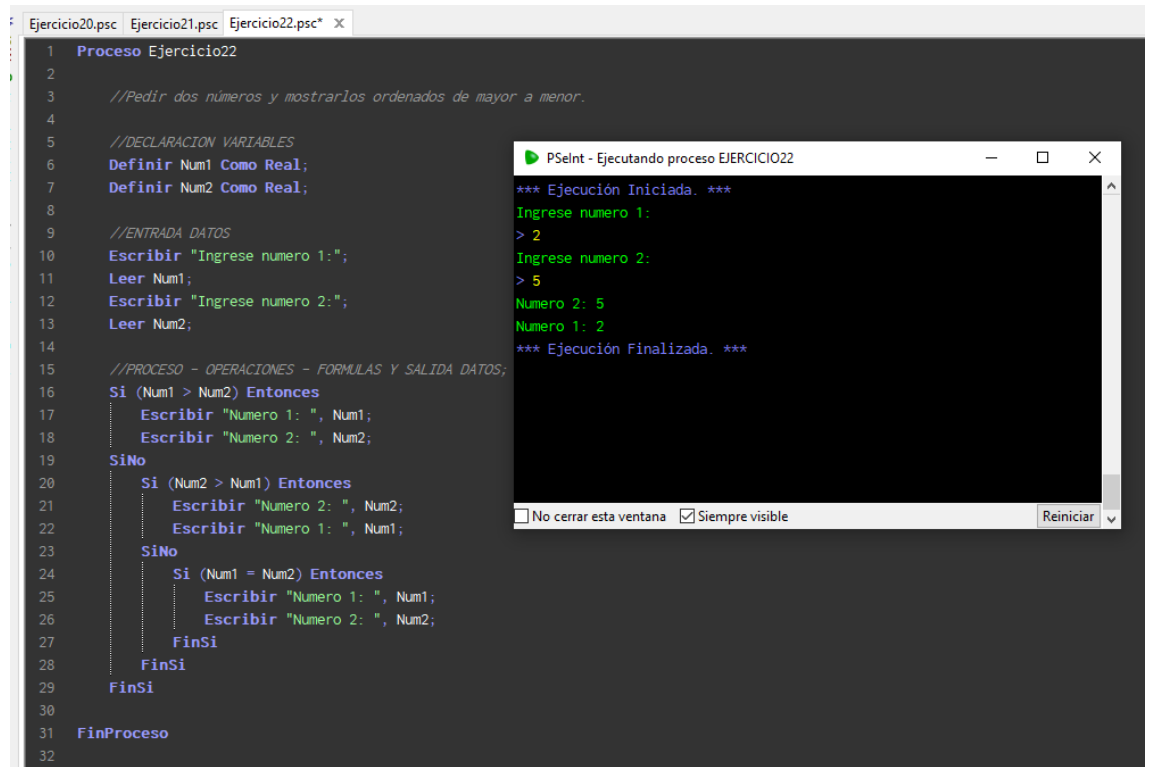


```
1 Proceso Ejercicio21
2
3 //Pedir dos números y decir cuál es el mayor o si son iguales.
4
5 //DECLARACION VARIABLES
6 Definir Num1 Como Real;
7 Definir Num2 Como Real;
8
9 //ENTRADA DATOS
10 Escribir "Ingrese numero 1:";
11 Leer Num1;
12 Escribir "Ingrese numero 2:";
13 Leer Num2;
14
15 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
16 Si (Num1 > Num2) Entonces
17     Escribir "El numero 1 es mayor al numero 2.";
18 SiNo
19     Si (Num2 > Num1) Entonces
20         Escribir "El numero 2 es mayor al numero 1.";
21     SiNo
22         Si (Num1 = Num2) Entonces
23             Escribir "Ambos numeros son iguales";
24         FinSi
25     FinSi
26 FinSi
27
28 FinProceso
29
```

*** Ejecución Iniciada. ***
Ingrese numero 1:
> 5
Ingrese numero 2:
> 5
Ambos numeros son iguales
*** Ejecución Finalizada. ***

22. Ejercicio22

Pedir dos números y mostrarlos ordenados de mayor a menor.



```
1 Proceso Ejercicio22
2
3 //Pedir dos números y mostrarlos ordenados de mayor a menor.
4
5 //DECLARACION VARIABLES
6 Definir Num1 Como Real;
7 Definir Num2 Como Real;
8
9 //ENTRADA DATOS
10 Escribir "Ingrese numero 1:";
11 Leer Num1;
12 Escribir "Ingrese numero 2:";
13 Leer Num2;
14
15 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
16 Si (Num1 > Num2) Entonces
17     Escribir "Numero 1: ", Num1;
18     Escribir "Numero 2: ", Num2;
19 SiNo
20     Si (Num2 > Num1) Entonces
21         Escribir "Numero 2: ", Num2;
22         Escribir "Numero 1: ", Num1;
23     SiNo
24         Si (Num1 = Num2) Entonces
25             Escribir "Numero 1: ", Num1;
26             Escribir "Numero 2: ", Num2;
27         FinSi
28     FinSi
29 FinSi
30
31 FinProceso
32
```

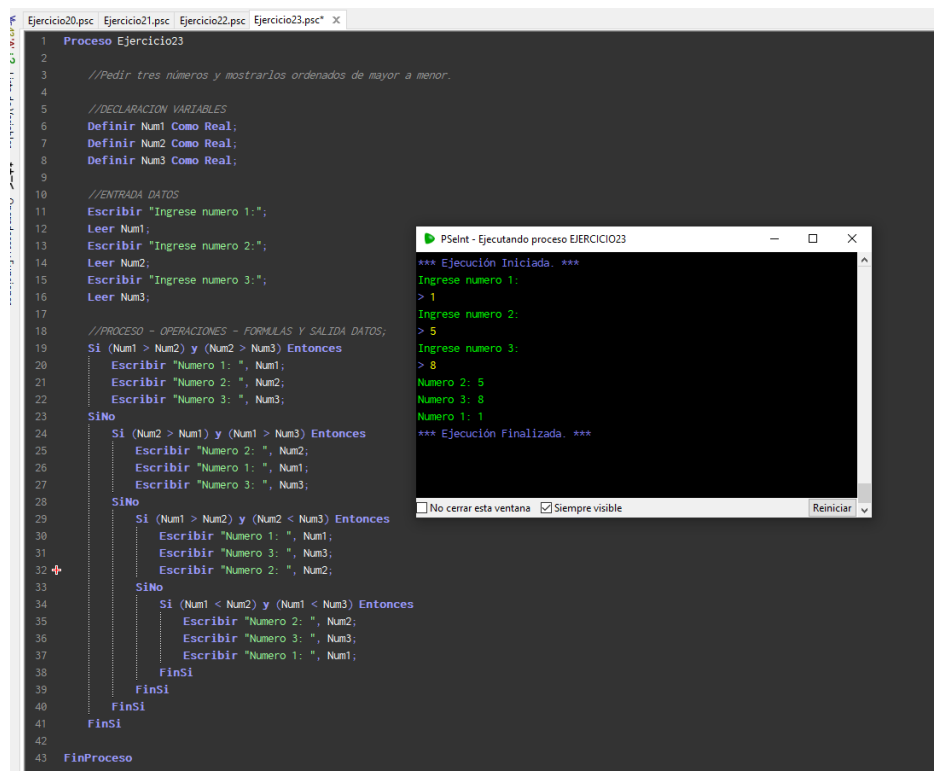
PSeInt - Ejecutando proceso EJERCICIO22

```
*** Ejecución Iniciada. ***
Ingrese numero 1:
> 2
Ingrese numero 2:
> 5
Numero 2: 5
Numero 1: 2
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

23. Ejercicio23

Pedir tres números y mostrarlos ordenados de mayor a menor.



```
1 Proceso Ejercicio23
2
3 //Pedir tres números y mostrarlos ordenados de mayor a menor.
4
5 //DECLARACION VARIABLES
6 Definir Num1 Como Real;
7 Definir Num2 Como Real;
8 Definir Num3 Como Real;
9
10 //ENTRADA DATOS
11 Escribir "Ingrese numero 1:";
12 Leer Num1;
13 Escribir "Ingrese numero 2:";
14 Leer Num2;
15 Escribir "Ingrese numero 3:";
16 Leer Num3;
17
18 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
19 Si (Num1 > Num2) y (Num2 > Num3) Entonces
20     Escribir "Numero 1: ", Num1;
21     Escribir "Numero 2: ", Num2;
22     Escribir "Numero 3: ", Num3;
23 SiNo
24     Si (Num2 > Num1) y (Num1 > Num3) Entonces
25         Escribir "Numero 2: ", Num2;
26         Escribir "Numero 1: ", Num1;
27         Escribir "Numero 3: ", Num3;
28     SiNo
29         Si (Num1 > Num2) y (Num2 < Num3) Entonces
30             Escribir "Numero 1: ", Num1;
31             Escribir "Numero 3: ", Num3;
32             Escribir "Numero 2: ", Num2;
33         SiNo
34             Si (Num1 < Num2) y (Num1 < Num3) Entonces
35                 Escribir "Numero 2: ", Num2;
36                 Escribir "Numero 3: ", Num3;
37                 Escribir "Numero 1: ", Num1;
38             FinSi
39         FinSi
40     FinSi
41 FinSi
42
43 FinProceso
44
```

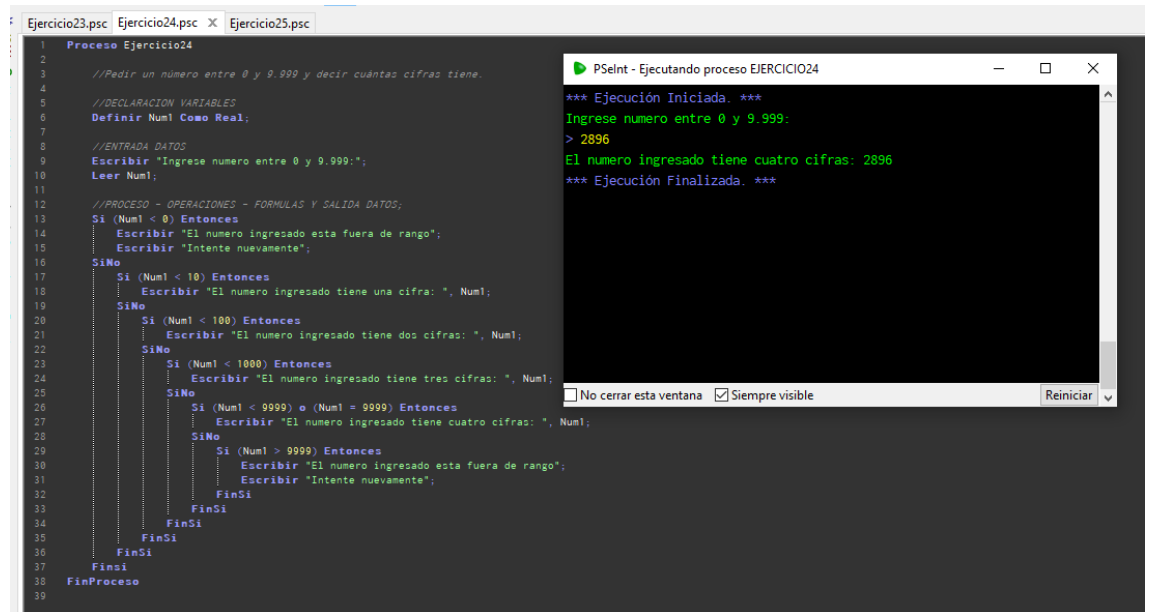
PSeInt - Ejecutando proceso EJERCICIO23

```
*** Ejecución Iniciada. ***
Ingrese numero 1:
> 1
Ingrese numero 2:
> 5
Ingrese numero 3:
> 8
Numero 2: 5
Numero 3: 8
Numero 1: 1
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

24. Ejercicio24

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

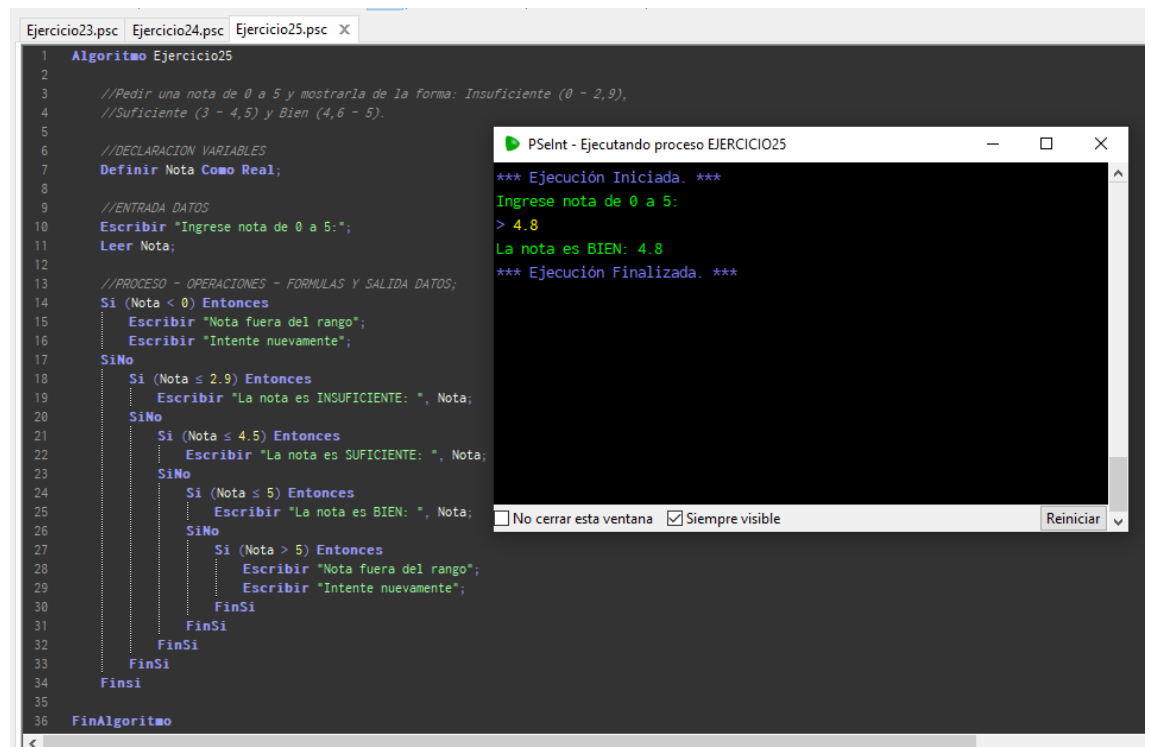


```
1 Proceso Ejercicio24
2
3 //Pedir un número entre 0 y 9.999 y decir cuántas cifras tiene.
4
5 //DECLARACION VARIABLES
6 Definir Num1 Como Real;
7
8 //ENTRADA DATOS
9 Escribir "Ingrese numero entre 0 y 9.999:";
10 Leer Num1;
11
12 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
13 Si (Num1 < 0) Entonces
14     Escribir "El numero ingresado esta fuera de rango";
15     Escribir "Intente nuevamente";
16 SiNo
17     Si (Num1 < 10) Entonces
18         Escribir "El numero ingresado tiene una cifra: ", Num1;
19     SiNo
20         Si (Num1 < 100) Entonces
21             Escribir "El numero ingresado tiene dos cifras: ", Num1;
22         SiNo
23             Si (Num1 < 1000) Entonces
24                 Escribir "El numero ingresado tiene tres cifras: ", Num1;
25             SiNo
26                 Si (Num1 < 9999) o (Num1 = 9999) Entonces
27                     Escribir "El numero ingresado tiene cuatro cifras: ", Num1;
28                 SiNo
29                     Si (Num1 > 9999) Entonces
30                         Escribir "El numero ingresado esta fuera de rango";
31                         Escribir "Intente nuevamente";
32                     FinSi
33                 FinSi
34             FinSi
35         FinSi
36     FinSi
37 Finsi
38 FinProceso
39
```

*** Ejecución Iniciada. ***
Ingrese numero entre 0 y 9.999:
> 2896
El numero ingresado tiene cuatro cifras: 2896
*** Ejecución Finalizada. ***

25. Ejercicio25

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
3 //Pedir una nota de 0 a 5 y mostrarla de la forma: Insuficiente (0 – 2,9),
4 //Suficiente (3 – 4,5) y Bien (4,6 – 5).
5
6 //DECLARACION VARIABLES
7 Definir Nota Como Real;
8
9 //ENTRADA DATOS
10 Escribir "Ingrese nota de 0 a 5:";
11 Leer Nota;
12
13 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
14 Si (Nota < 0) Entonces
15     Escribir "Nota fuera del rango";
16     Escribir "Intente nuevamente";
17 SiNo
18     Si (Nota ≤ 2.9) Entonces
19         Escribir "La nota es INSUFICIENTE: ", Nota;
20     SiNo
21         Si (Nota ≤ 4.5) Entonces
22             Escribir "La nota es SUFICIENTE: ", Nota;
23         SiNo
24             Si (Nota ≤ 5) Entonces
25                 Escribir "La nota es BIEN: ", Nota;
26             SiNo
27                 Si (Nota > 5) Entonces
28                     Escribir "Nota fuera del rango";
29                     Escribir "Intente nuevamente";
30                 FinSi
31             FinSi
32         FinSi
33     FinSi
34 Finsi
35 FinAlgoritmo
36
```

*** Ejecución Iniciada. ***
Ingrese nota de 0 a 5:
> 4.8
La nota es BIEN: 4.8
*** Ejecución Finalizada. ***

26. Ejercicio26

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

The screenshot shows a Pascal IDE with a file named 'Ejercicio26.psc' open. The code is an algorithm that takes an integer input between 0 and 10 and outputs its name in Spanish (cero, uno, dos, etc.). It uses a series of 'Si' (if) statements to handle each possible value. A 'PSeInt' window titled 'Ejecutando proceso EJERCICIO26' shows the execution results: 'Ingreso numero entre 0 a 10: > 10' and 'La nota es DIEZ: 10'.

```
1 Algoritmo Ejercicio26
2 //Pedir una nota numérica entera entre 0 y 10.
3 //y mostrar dicha nota de la forma: cero, uno, dos, tres...
4
5 //DECLARACION VARIABLES
6 Definir Nota Como Entero;
7
8 //ENTRADA DATOS
9 Escribir "Ingrese numero entre 0 a 10:";
10 Leer Nota;
11
12 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
13 Si (Nota = 0) Entonces
14 | Escribir "La nota es CERO: ", Nota;
15 SiNo
16 Si (Nota = 1) Entonces
17 | Escribir "La nota es UNO: ", Nota;
18 SiNo
19 Si (Nota = 2) Entonces
20 | Escribir "La nota es DOS: ", Nota;
21 SiNo
22 Si (Nota = 3) Entonces
23 | Escribir "La nota es TRES: ", Nota;
24 SiNo
25 Si (Nota = 4) Entonces
26 | Escribir "La nota es CUATRO: ", Nota;
27 SiNo
28 Si (Nota = 5) Entonces
29 | Escribir "La nota es CINCO: ", Nota;
30 SiNo
31 Si (Nota = 6) Entonces
32 | Escribir "La nota es SEIS: ", Nota;
33 SiNo
34 Si (Nota = 7) Entonces
35 | Escribir "La nota es SIETE: ", Nota;
36 SiNo
37 Si (Nota = 8) Entonces
38 | Escribir "La nota es OCHO: ", Nota;
39 SiNo
40 Si (Nota = 9) Entonces
41 | Escribir "La nota es NUEVE: ", Nota;
42 SiNo
43 Si (Nota = 10) Entonces
44 | Escribir "La nota es DIEZ: ", Nota;
45 Si (Nota > 10) Entonces
46 | Escribir "Nota fuera del rango";
47 | Escribir "Intente nuevamente";
48 FinSi
49 FinSi
50 FinSi
51 FinSi
52 FinSi
53 FinSi
54 FinSi
55 FinSi
56 FinSi
57 FinSi
58 FinSi
59 FinSi
60 FinSi
61 FinSi
62 FinSi
63 FinSi
64 FinAlgoritmo
```

PSeInt - Ejecutando proceso EJERCICIO26

```
*** Ejecución Iniciada. ***
Ingreso numero entre 0 a 10:
> 10
La nota es DIEZ: 10
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

27. Ejercicio27

Pedir un número y decir si es par o impar.

The screenshot shows a Pascal IDE with a file named 'Ejercicio27.psc' open. The code is an algorithm that takes an integer input and checks if it is even or odd using the modulo operator. A 'PSeInt' window titled 'Ejecutando proceso EJERCICIO27' shows the execution results: 'Ingreso numero: > 5' and 'El numero es IMPAR: 5'.

```
1 Algoritmo Ejercicio27
2
3 // Pedir un número y decir si es par o impar.
4
5 //DECLARACION VARIABLES
6 Definir Num1 Como Entero;
7
8 //ENTRADA DATOS
9 Escribir "Ingreso numero:";
10 Leer Num1;
11
12 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
13 Si (Num1 MOD 2 = 0) Entonces
14 | Escribir "El numero es PAR: ", Num1;
15 SiNo
16 | Escribir "El numero es IMPAR: ", Num1;
17 FinSi
18
19 FinAlgoritmo
20
```

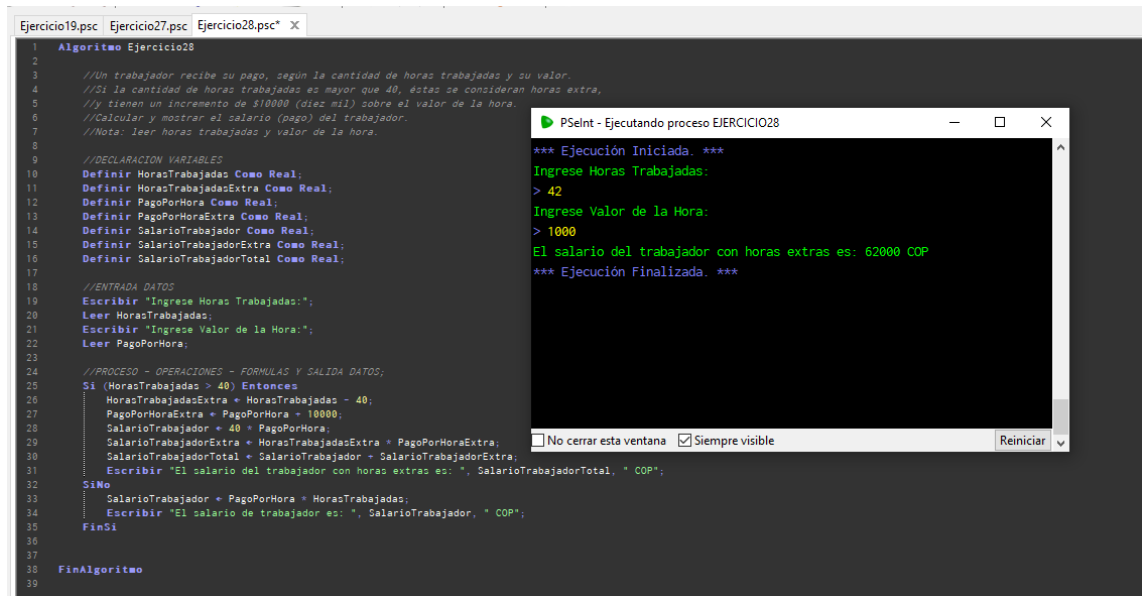
PSeInt - Ejecutando proceso EJERCICIO27

```
*** Ejecución Iniciada. ***
Ingreso numero:
> 5
El numero es IMPAR: 5
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar

28. Ejercicio28

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.



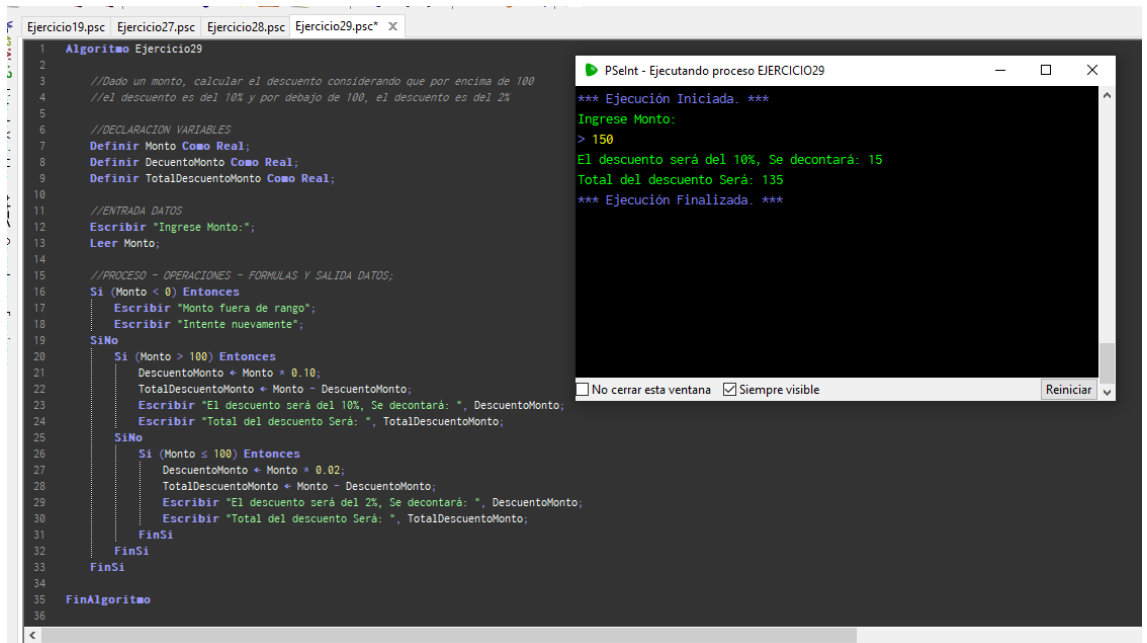
```
1 Algoritmo Ejercicio28
2
3 //Un trabajador recibe su pago, según la cantidad de horas trabajadas y su valor.
4 //Si la cantidad de horas trabajadas es mayor que 40, éstas se consideran horas extra,
5 // tienen un incremento de $10000 (diez mil) sobre el valor de la hora.
6 //Calcular y mostrar el salario (pago) del trabajador.
7 //Nota: leer horas trabajadas y valor de la hora.
8
9 //DECLARACION VARIABLES
10 Definir HorasTrabajadas Como Real;
11 Definir HorasTrabajadasExtra Como Real;
12 Definir PagoPorHora Como Real;
13 Definir PagoPorHoraExtra Como Real;
14 Definir SalarioTrabajador Como Real;
15 Definir SalarioTrabajadorExtra Como Real;
16 Definir SalarioTrabajadorTotal Como Real;
17
18 //ENTRADA DATOS
19 Escribir "Ingrese Horas Trabajadas:";
20 Leer HorasTrabajadas;
21 Escribir "Ingrese Valor de la Hora:";
22 Leer PagoPorHora;
23
24 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
25 Si (HorasTrabajadas > 40) Entonces
26   HorasTrabajadasExtra ← HorasTrabajadas - 40;
27   PagoPorHoraExtra ← PagoPorHora + 10000;
28   SalarioTrabajador ← 40 * PagoPorHora;
29   SalarioTrabajadorExtra ← HorasTrabajadasExtra * PagoPorHoraExtra;
30   SalarioTrabajadorTotal ← SalarioTrabajador + SalarioTrabajadorExtra;
31   Escribir "El salario del trabajador con horas extras es: ", SalarioTrabajadorTotal, " COP";
32 SiNo
33   SalarioTrabajador ← PagoPorHora * HorasTrabajadas;
34   Escribir "El salario de trabajador es: ", SalarioTrabajador, " COP";
35 FinSi
36
37 FinAlgoritmo
38
39
```

Execution output in PSeInt:

```
*** Ejecución Iniciada. ***
Ingrese Horas Trabajadas:
> 42
Ingrese Valor de la Hora:
> 1000
El salario del trabajador con horas extras es: 62000 COP
*** Ejecución Finalizada. ***
```

29. Ejercicio29

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%



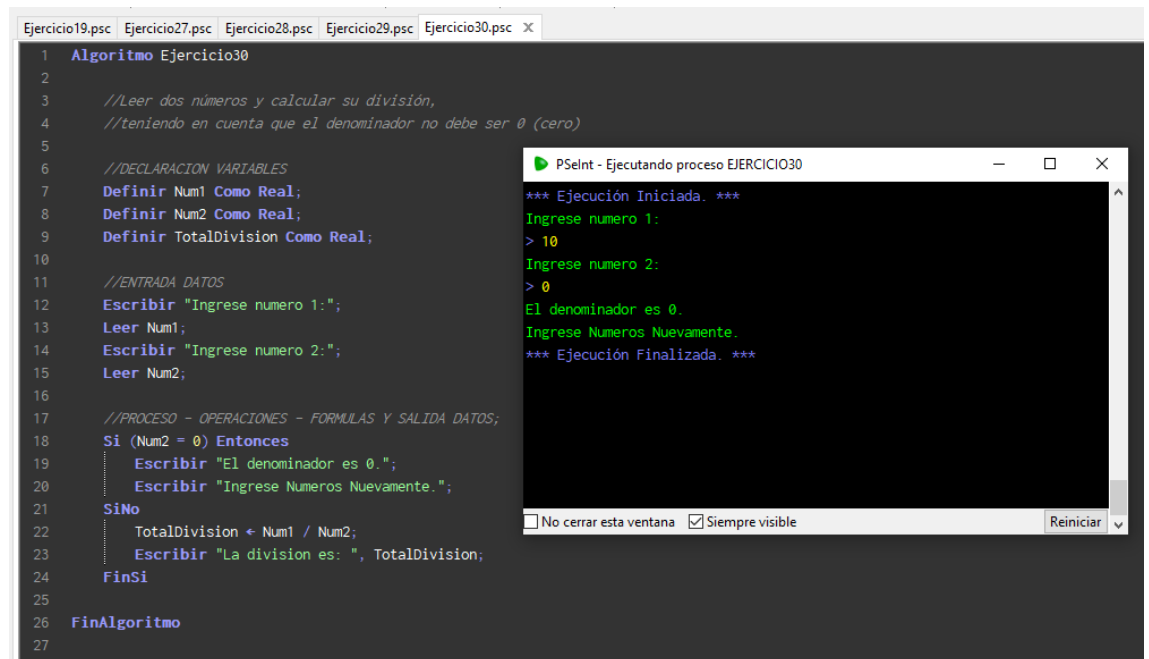
```
1 Algoritmo Ejercicio29
2
3 //Dado un monto, calcular el descuento considerando que por encima de 100
4 //el descuento es del 10% y por debajo de 100, el descuento es del 2%
5
6 //DECLARACION VARIABLES
7 Definir Monto Como Real;
8 Definir DescuentoMonto Como Real;
9 Definir TotalDescuentoMonto Como Real;
10
11 //ENTRADA DATOS
12 Escribir "Ingrese Monto:";
13 Leer Monto;
14
15 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
16 Si (Monto < 0) Entonces
17   Escribir "Monto fuera de rango";
18   Escribir "Intente nuevamente";
19 SiNo
20   Si (Monto > 100) Entonces
21     DescuentoMonto ← Monto * 0.10;
22     TotalDescuentoMonto ← Monto - DescuentoMonto;
23     Escribir "El descuento será del 10%, Se decontará: ", DescuentoMonto;
24     Escribir "Total del descuento Será: ", TotalDescuentoMonto;
25   SiNo
26     Si (Monto ≤ 100) Entonces
27       DescuentoMonto ← Monto * 0.02;
28       TotalDescuentoMonto ← Monto - DescuentoMonto;
29       Escribir "El descuento será del 2%, Se decontará: ", DescuentoMonto;
30       Escribir "Total del descuento Será: ", TotalDescuentoMonto;
31     FinSi
32   FinSi
33 FinSi
34
35 FinAlgoritmo
36
37
```

Execution output in PSeInt:

```
*** Ejecución Iniciada. ***
Ingrese Monto:
> 150
El descuento será del 10%, Se decontará: 15
Total del descuento Será: 135
*** Ejecución Finalizada. ***
```

30. Ejercicio30

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



```
1  Algoritmo Ejercicio30
2
3  //Leer dos números y calcular su división,
4  //teniendo en cuenta que el denominador no debe ser 0 (cero)
5
6  //DECLARACION VARIABLES
7  Definir Num1 Como Real;
8  Definir Num2 Como Real;
9  Definir TotalDivision Como Real;
10
11 //ENTRADA DATOS
12 Escribir "Ingrese numero 1: ";
13 Leer Num1;
14 Escribir "Ingrese numero 2: ";
15 Leer Num2;
16
17 //PROCESO - OPERACIONES - FORMULAS Y SALIDA DATOS;
18 Si (Num2 = 0) Entonces
19     Escribir "El denominador es 0.";
20     Escribir "Ingrese Numeros Nuevamente.";
21 SiNo
22     TotalDivision ← Num1 / Num2;
23     Escribir "La division es: ", TotalDivision;
24 FinSi
25
26 FinAlgoritmo
27
```

PSeInt - Ejecutando proceso EJERCICIO30

```
*** Ejecución Iniciada. ***
Ingrese numero 1:
> 10
Ingrese numero 2:
> 0
El denominador es 0.
Ingrese Numeros Nuevamente.
*** Ejecución Finalizada. ***
```

☐ No cerrar esta ventana ☒ Siempre visible Reiniciar