



Universidad Autónoma De Tamaulipas

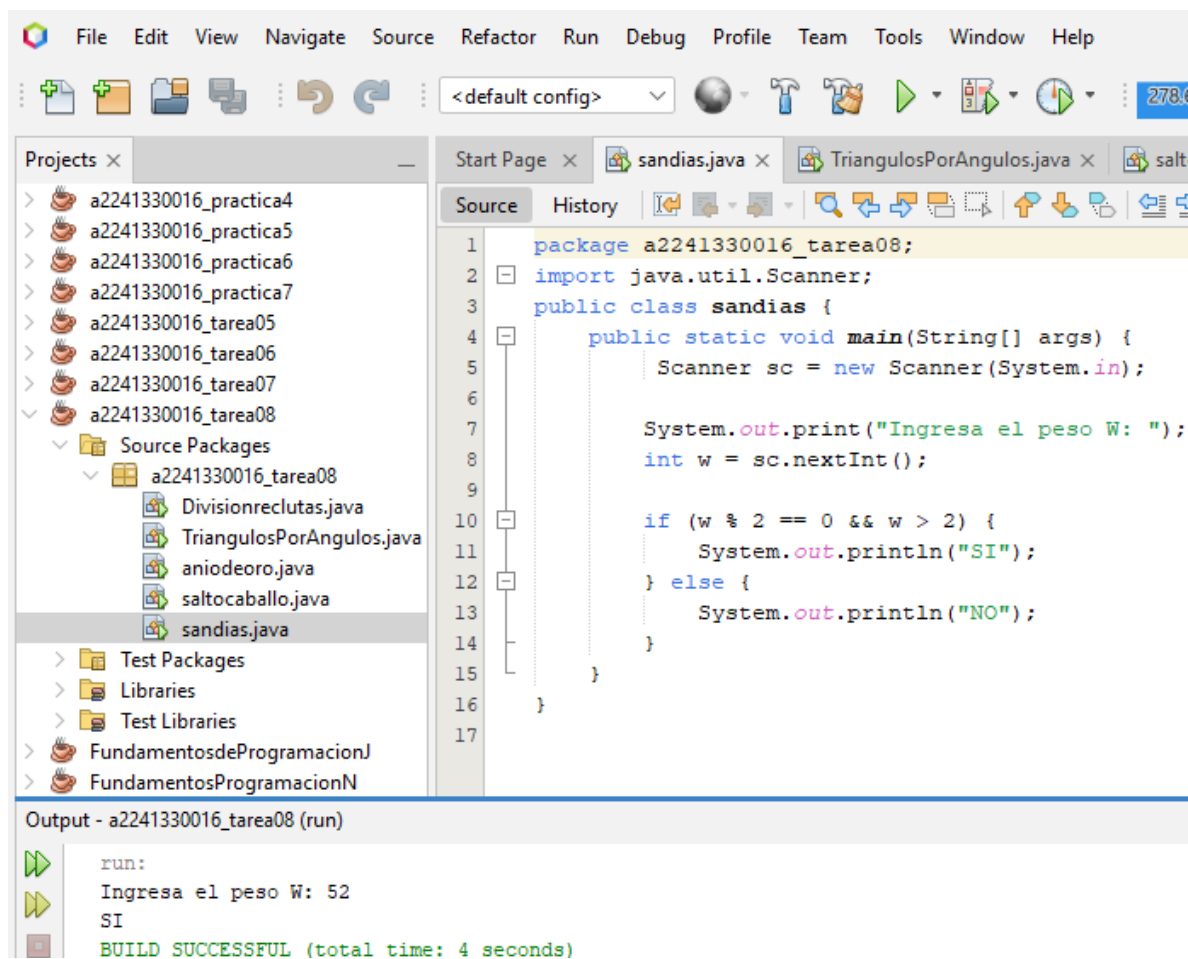
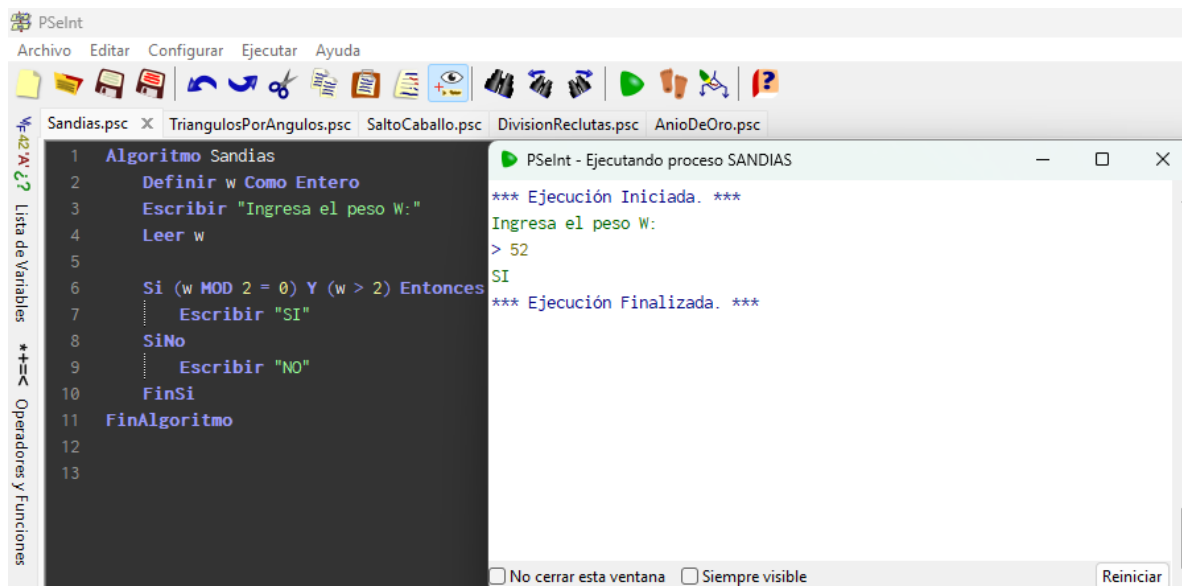
Fundamentos de Programación

1-N

Tarea 08

Molina Meneses Diego

Problema 1 sandias:



Problema 2 TriangulosPorAngulos:

The screenshot shows the PSeInt IDE with a pseudocode algorithm for classifying triangles based on their angles. The algorithm prompts the user to enter three angles (A, B, C) and then checks if they form a right triangle (one angle is 90), an obtuse triangle (one angle is greater than 90), or an acute triangle (all angles are less than 90). If none of these conditions are met, it declares the angles as invalid.

```
1 Algoritmo TriangulosPorAngulos
2 Definir A, B, C Como Entero
3 Escribir "Ingresa A, B, C:"
4 Leer A, B, C
5
6 Si (A > 0) Y (B > 0) Y (C > 0) Y (A + B + C = 180) Entonces
7     Si (A = 90) O (B = 90) O (C = 90) Entonces
8         Escribir "Rectangulo"
9     SiNo
10        Si (A > 90) O (B > 90) O (C > 90) Entonces
11            Escribir "Obtusangulo"
12        SiNo
13            Escribir "Acutangulo"
14        FinSi
15    FinSi
16 SiNo
17     Escribir "Angulos Invalidos"
18 FinSi
19 FinAlgoritmo
```

The execution window shows the following output:

```
*** Ejecución Iniciada. ***
Ingresa A, B, C:
> 90
> 60
> 30
Rectangulo
*** Ejecución Finalizada. ***
```

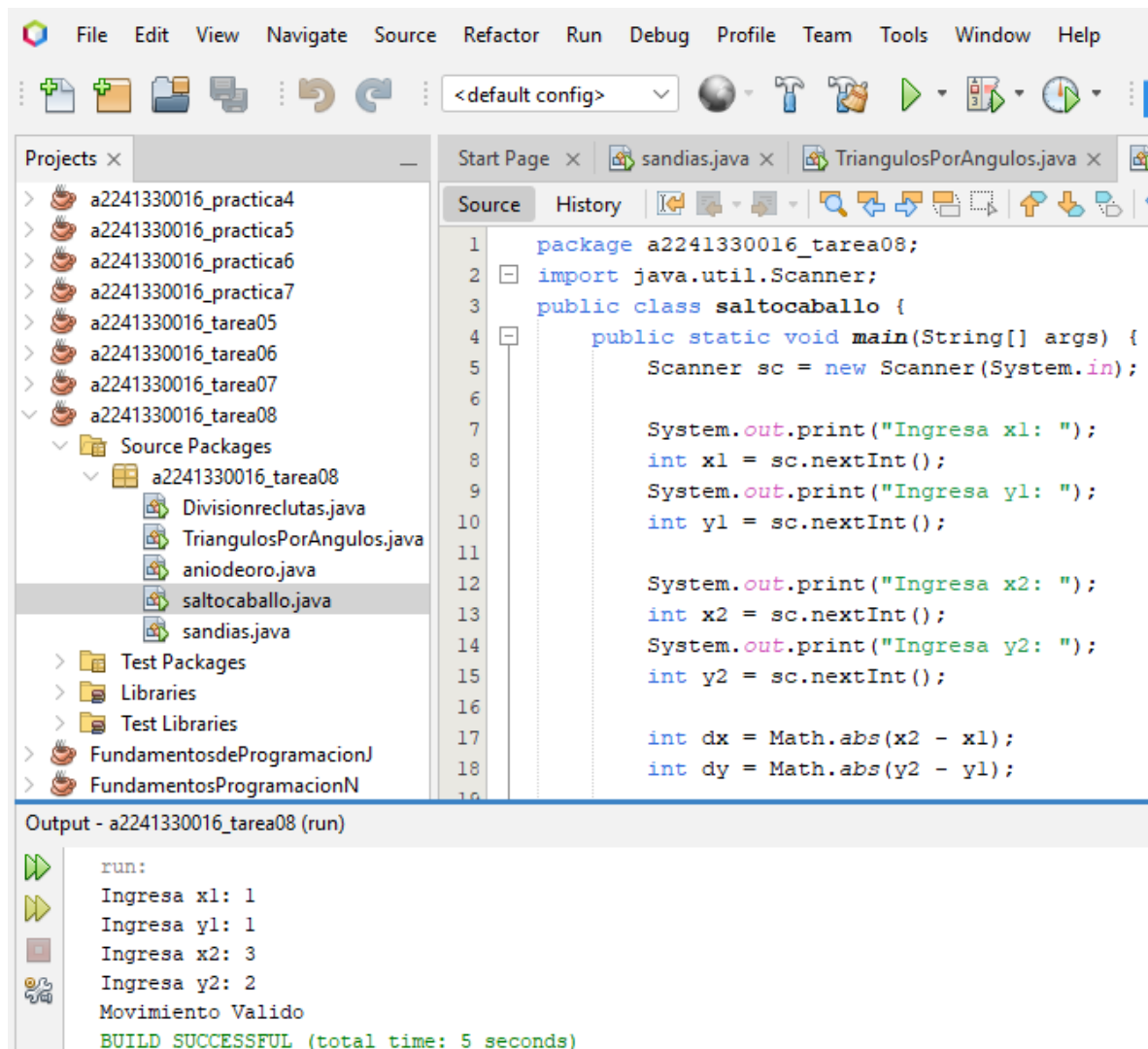
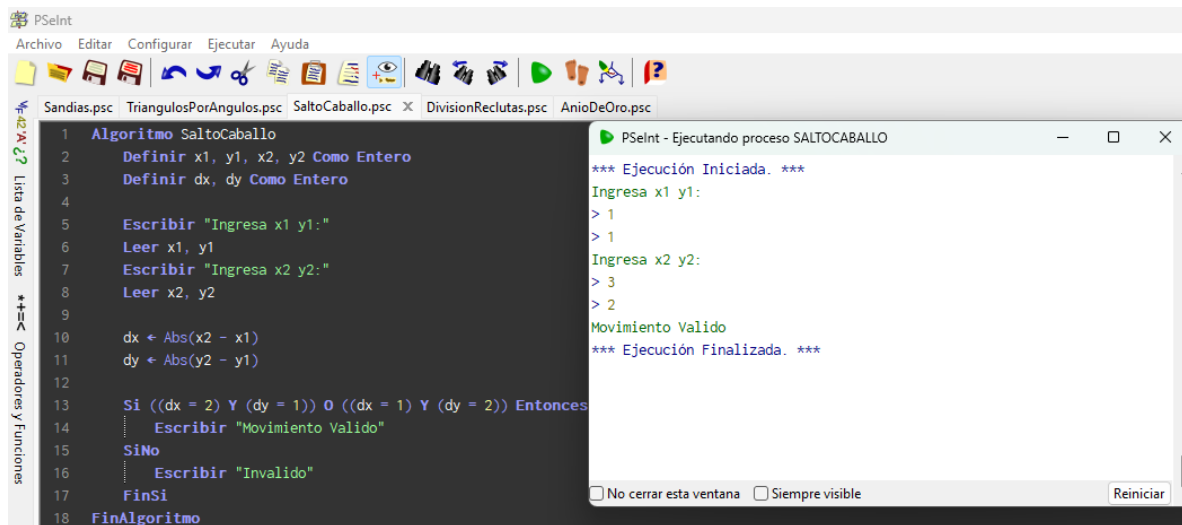
The screenshot shows an IDE with the Java implementation of the triangle classification algorithm. The code uses the Scanner class to read input and System.out.println to display the result. The logic is identical to the pseudocode in the previous image.

```
1 package a2241330016_tarea08;
2 import java.util.Scanner;
3 public class TriangulosPorAngulos {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.print("Ingresa A: ");
8         int A = sc.nextInt();
9         System.out.print("Ingresa B: ");
10        int B = sc.nextInt();
11        System.out.print("Ingresa C: ");
12        int C = sc.nextInt();
13
14        if (A > 0 && B > 0 && C > 0 && (A + B + C == 180)) {
15            if (A == 90 || B == 90 || C == 90) {
16                System.out.println("Rectangulo");
17            } else if (A > 90 || B > 90 || C > 90) {
18                System.out.println("Obtusangulo");
19            } else {
20                System.out.println("Acutangulo");
21            }
22        } else {
23            System.out.println("Angulos Invalidos");
24        }
25    }
26 }
```

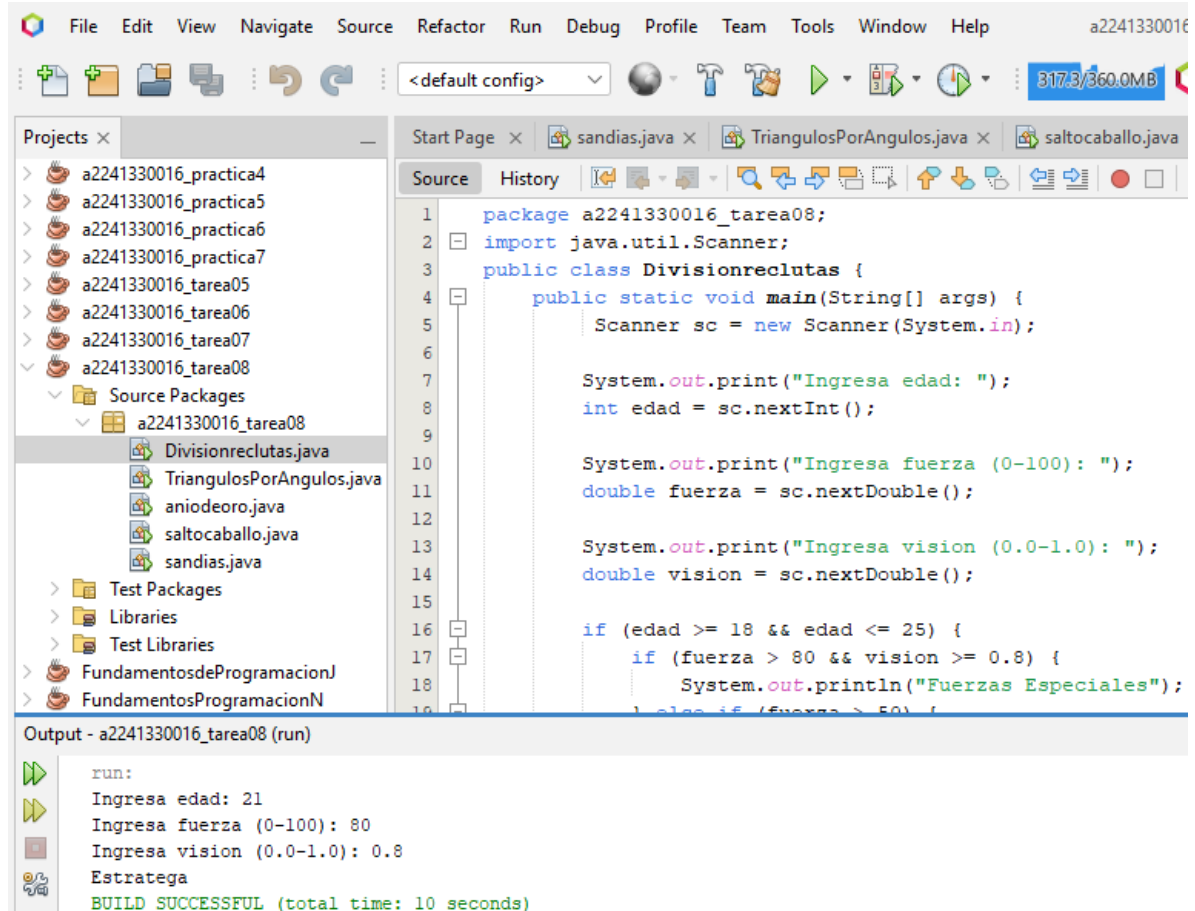
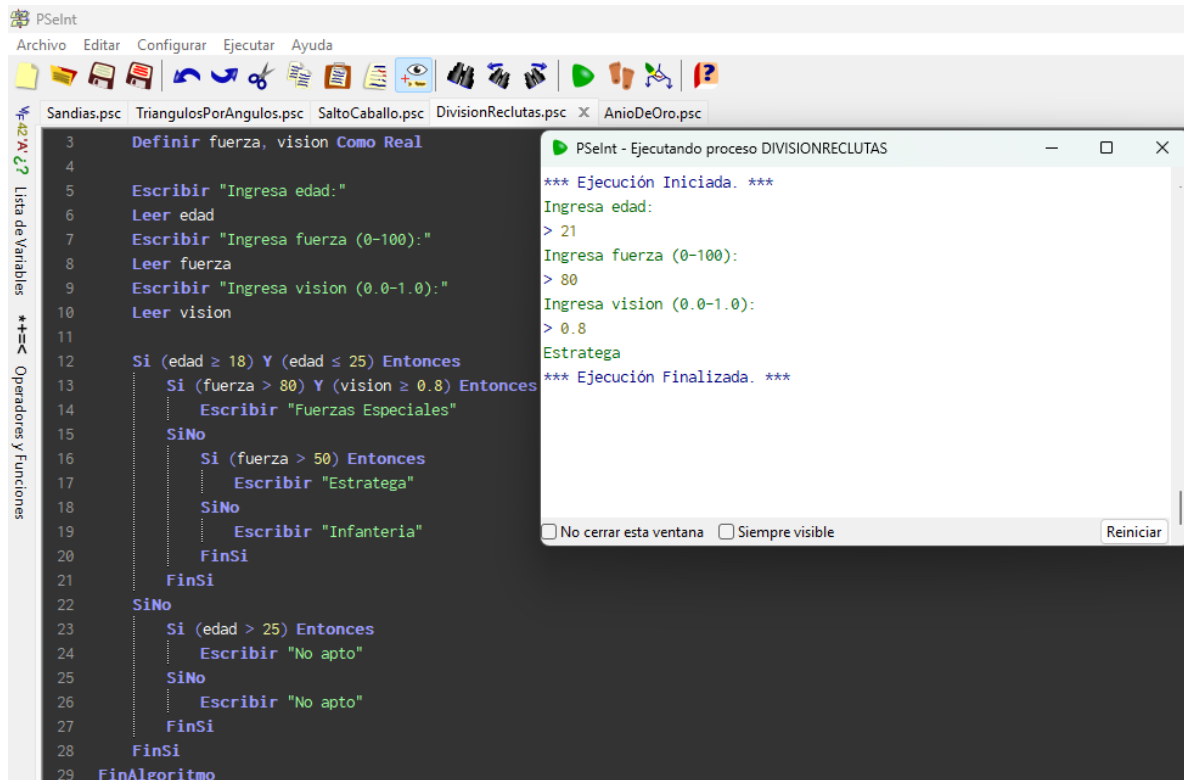
The output window shows the execution results:

```
run:
Ingresa A: 90
Ingresa B: 60
Ingresa C: 30
Rectangulo
BUILD SUCCESSFUL (total time: 7 seconds)
```

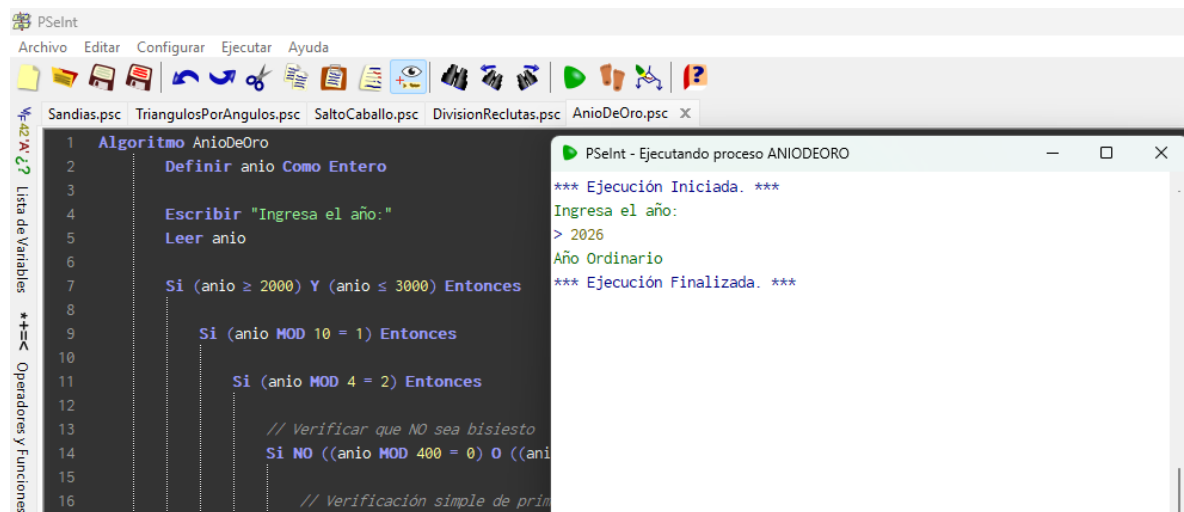
Problema 3 SaltoCaballo:



Problema 4 Division Reclutas:



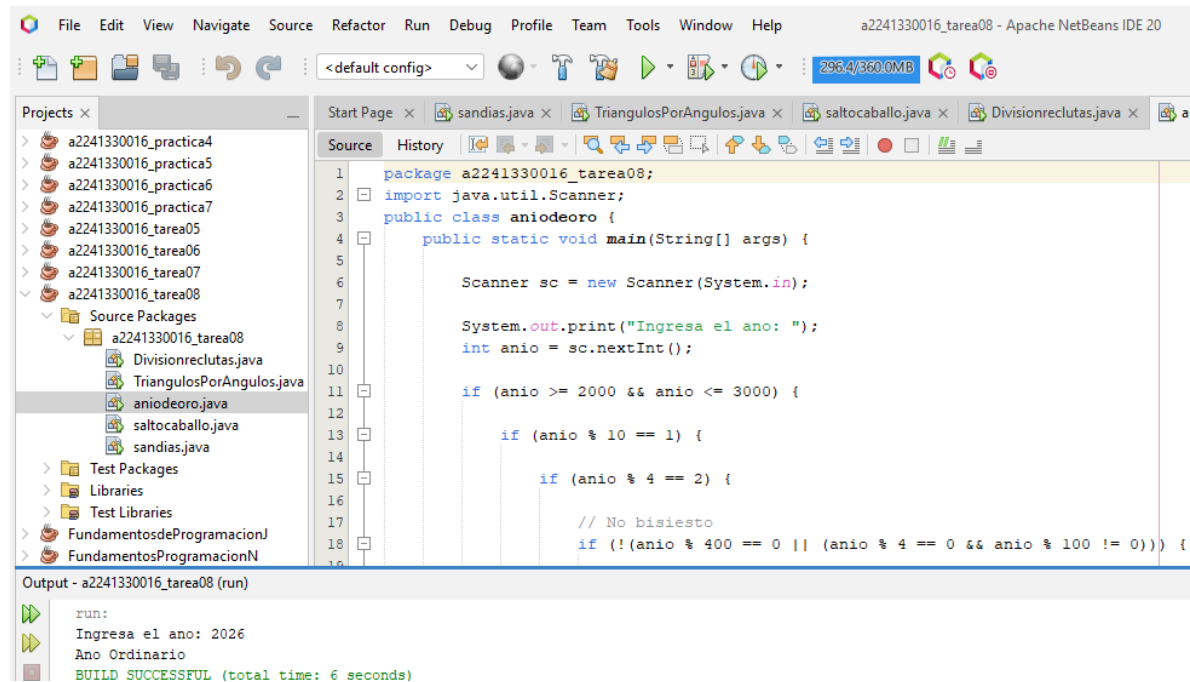
Problema 5 año de oro:



The screenshot shows the PSeInt IDE with a pseudocode algorithm for determining if a year is a 'Golden Year' (Año de Oro). The algorithm defines the year as an integer, prompts the user to enter the year, and then checks if the year is between 2000 and 3000. If it is, it checks if it is a leap year (divisible by 4 but not by 100, or divisible by 400). If it is a leap year, it prints 'Año Ordinario'. The execution output shows the year 2026 entered, which is correctly identified as an 'Año Ordinario'.

```
1 Algoritmo AñoDeOro
2   Definir anio Como Entero
3
4   Escribir "Ingresa el año:"
5   Leer anio
6
7   Si (anio ≥ 2000) Y (anio ≤ 3000) Entonces
8
9       Si (anio MOD 10 = 1) Entonces
10
11           Si (anio MOD 4 = 2) Entonces
12
13               // Verificar que NO sea bisiesto
14               Si NO ((anio MOD 400 = 0) O ((anio MOD 100 = 0) Y (anio MOD 4 = 0))) Entonces
15                   // Verificación simple de primo
16                   Escribir "Año Ordinario"
```

*** Ejecución Iniciada. ***
Ingresa el año:
> 2026
Año Ordinario
*** Ejecución Finalizada. ***



The screenshot shows the Apache NetBeans IDE with the Java source code for the 'Año de Oro' problem. The code uses the Scanner class to read the year and implements the same logic as the pseudocode. The execution output shows the year 2026 entered, which is correctly identified as an 'Año Ordinario'.

```
1 package a2241330016_tarea08;
2 import java.util.Scanner;
3 public class anioDeoro {
4     public static void main(String[] args) {
5
6         Scanner sc = new Scanner(System.in);
7
8         System.out.print("Ingresa el año: ");
9         int anio = sc.nextInt();
10
11         if (anio >= 2000 && anio <= 3000) {
12
13             if (anio % 10 == 1) {
14
15                 if (anio % 4 == 2) {
16
17                     // No bisiesto
18                     if (!(anio % 400 == 0 || (anio % 4 == 0 && anio % 100 != 0))) {
19                         System.out.println("Año Ordinario");
20                     }
21                 }
22             }
23         }
24     }
25 }
```

run:
Ingresa el año: 2026
Año Ordinario
BUILD SUCCESSFUL (total time: 6 seconds)

Problema 6 PuntoEnRectangulo:

PSelnt

Archivo Editar Configurar Ejecutar Ayuda

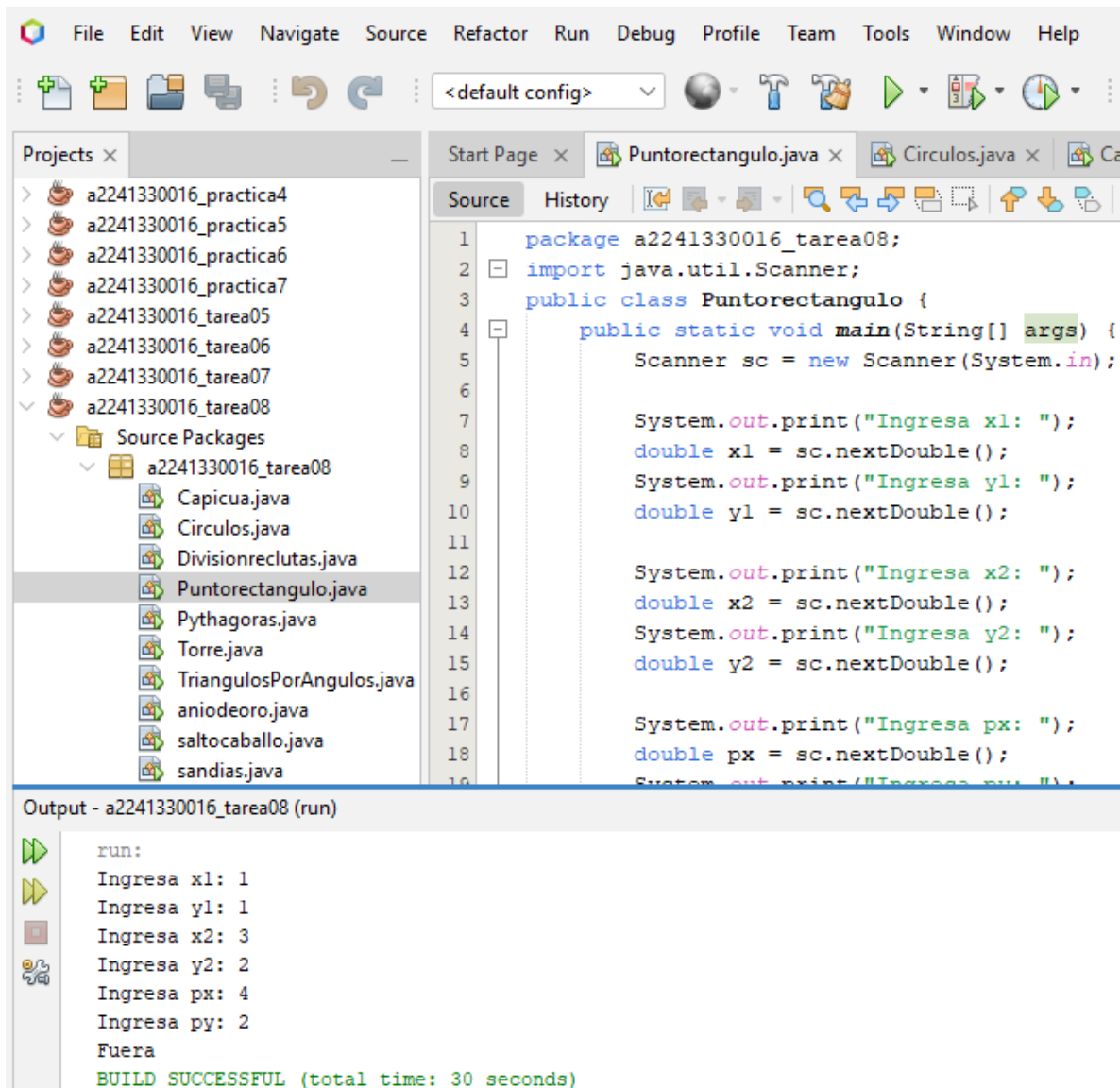
PuntoEnRectangulo.psc X Circulos.psc Capicua.psc Pythagoras.psc Torre.psc

```
1 Algoritmo PuntoEnRectangulo
2   Definir x1,y1,x2,y2,px,py Como Real
3
4   Escribir "Ingresa x1: "
5   Leer x1
6   Escribir "Ingresa y1: "
7   Leer y1
8   Escribir "Ingresa x2: "
9   Leer x2
10  Escribir "Ingresa y2: "
11  Leer y2
12  Escribir "Ingresa px: "
13  Leer px
14  Escribir "Ingresa py: "
15  Leer py
16
17  Si (px > x1 Y px < x2) Y (py > y1 Y py < y2) Entonces
18      Escribir "Dentro"
19  SiNo
20      Si (px ≥ x1 Y px ≤ x2) Y (py ≥ y1 Y py ≤ y2) Entonces
21          Escribir "Borde"
22      SiNo
23          Escribir "Fuera"
24      FinSi
25  FinSi
26 FinAlgoritmo
```

PSelnt - Ejecutando proceso PUNTOENRECTANGULO

> 1
Ingresa x2:
> 3
Ingresa y2:
> 2
Ingresa px:
> 4
Ingresa py:
> 2
Fuera
*** Ejecución Finalizada. ***

☐ No cerrar esta ventana ☐ Siempre visible Reiniciar



Problema 7 Circulos:

PSelnt

Archivo Editar Configurar Ejecutar Ayuda

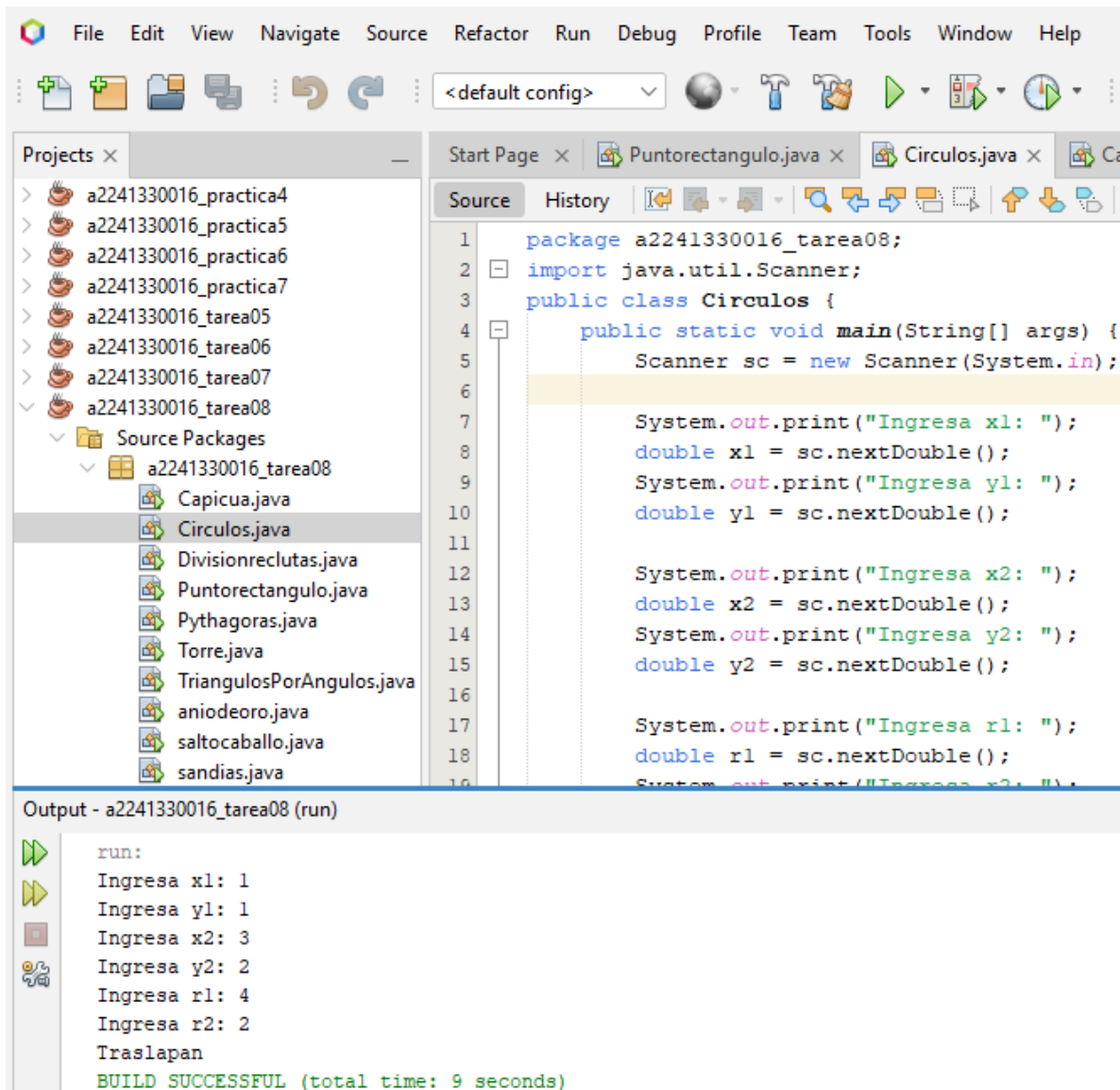
PuntoEnRectangulo.psc Círculos.psc X Capicua.psc Pythagoras.psc Torre.psc

```
1  Algoritmo Círculos
2  Definir x1,y1,x2,y2,r1,r2,dist2 Como Real
3
4  Escribir "Ingresa x1: "
5  Leer x1
6  Escribir "Ingresa y1: "
7  Leer y1
8  Escribir "Ingresa x2: "
9  Leer x2
10 Escribir "Ingresa y2: "
11 Leer y2
12 Escribir "Ingresa r1: "
13 Leer r1
14 Escribir "Ingresa r2: "
15 Leer r2
16
17 dist2 ← (x2-x1)² + (y2-y1)²
18
19 Si dist2 = (r1+r2)² Entonces
20     Escribir "Tangentes"
21 SiNo
22     Si dist2 < (r1+r2)² Entonces
23         Escribir "Traslapan"
24     SiNo
25         Escribir "No se tocan"
26     FinSi
27 FinSi
28 FinAlgoritmo
```

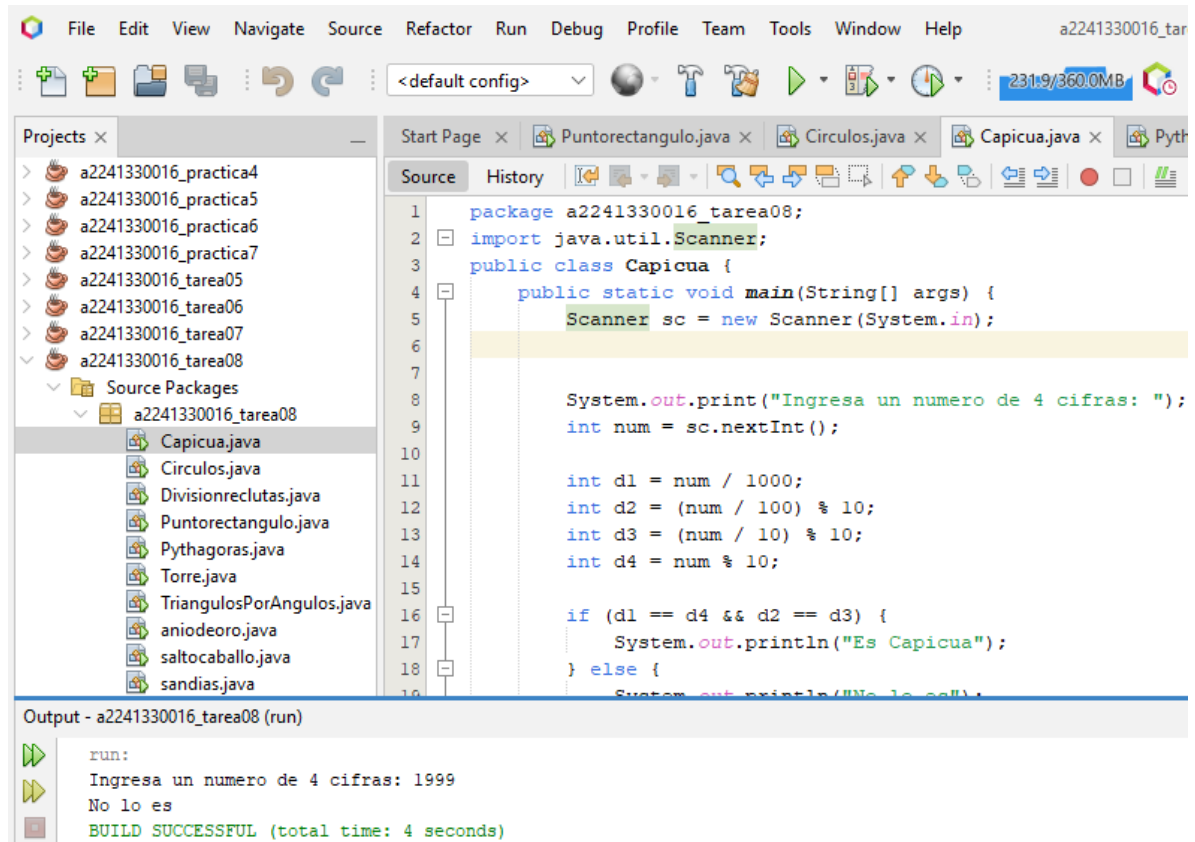
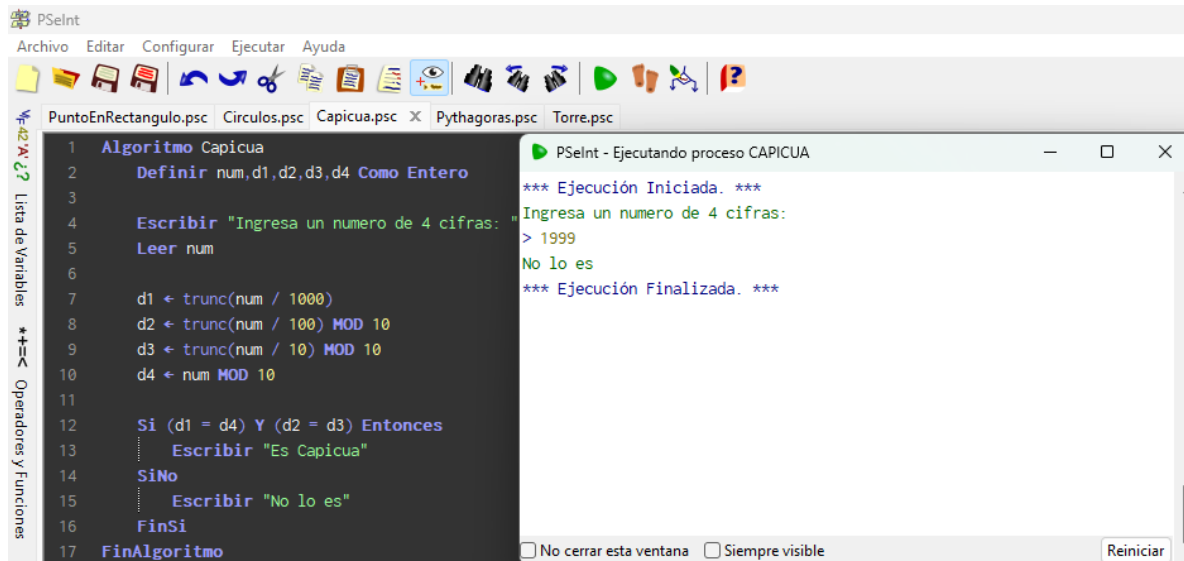
PSelnt - Ejecutando proceso CIRCULOS

```
> 1
Ingresa y1:
> 1
Ingresa x2:
> 3
Ingresa y2:
> 2
Ingresa r1:
> 4
Ingresa r2:
> 2
Traslapan
*** Ejecución Finalizada. ***
```

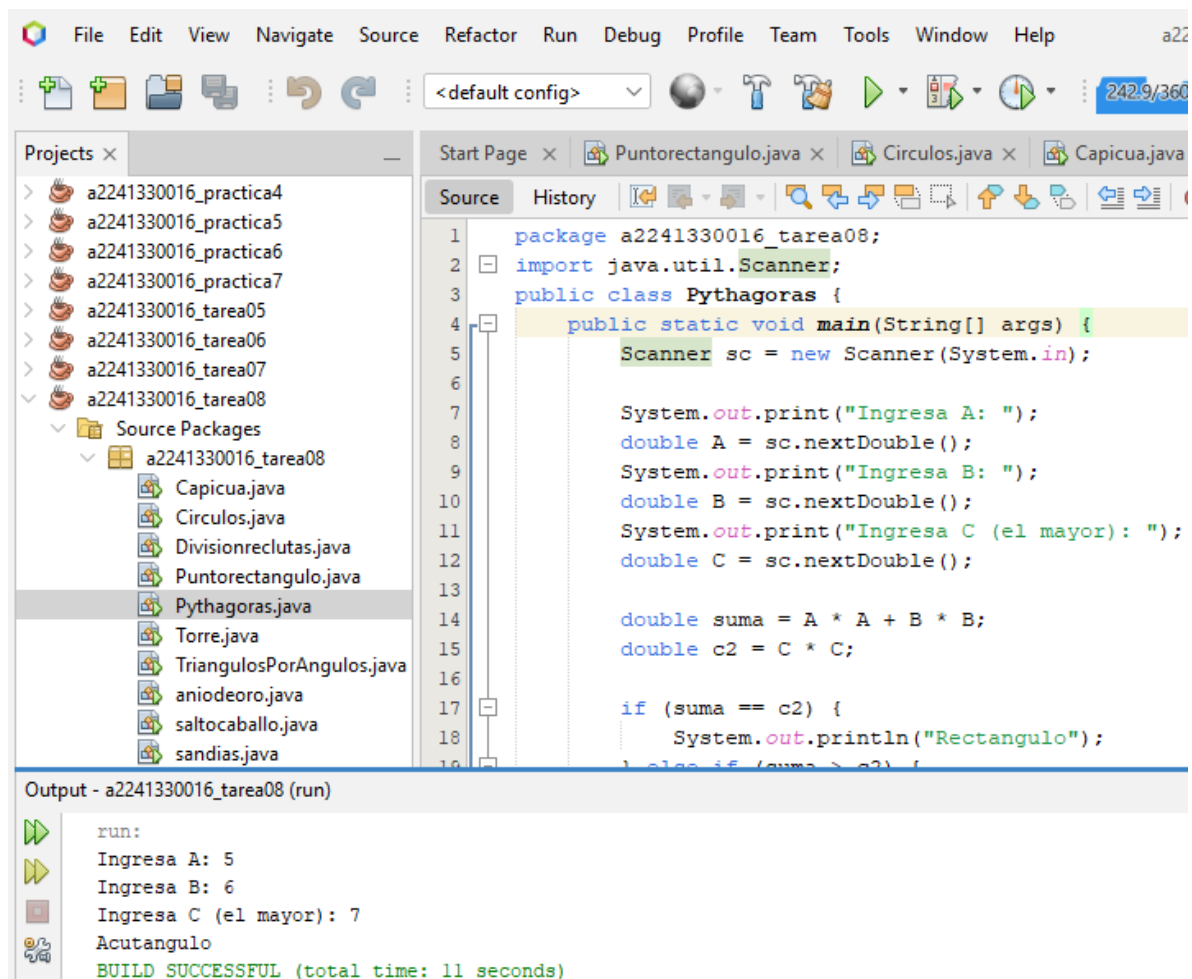
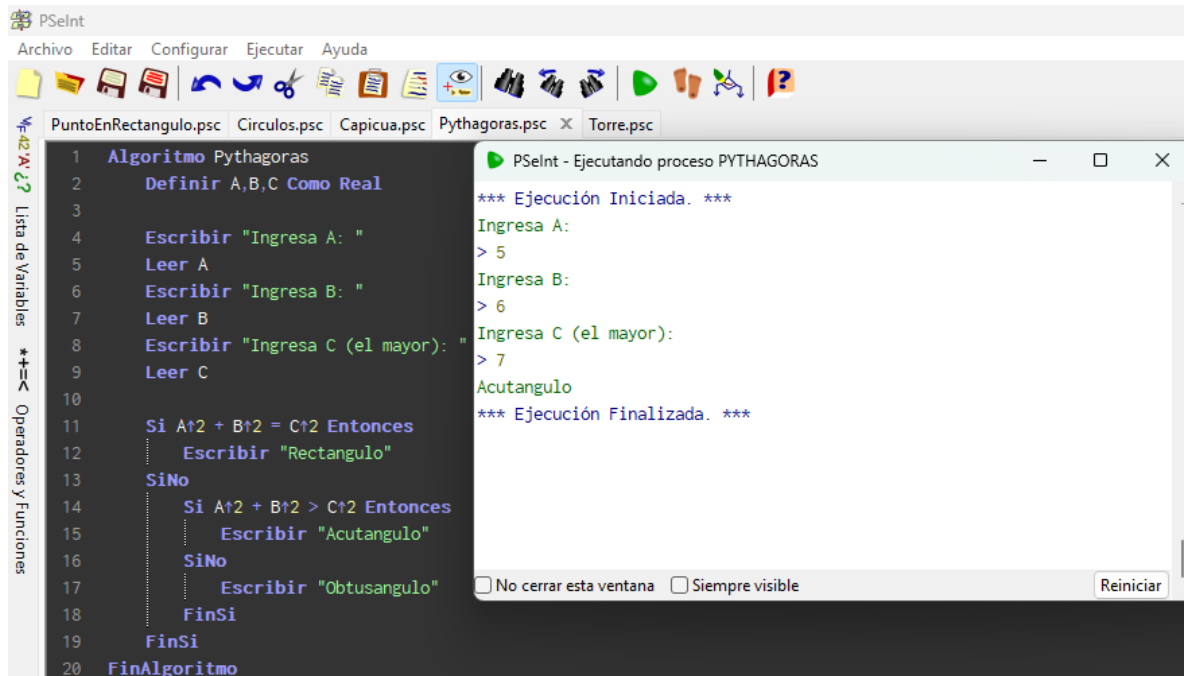
☐ No cerrar esta ventana ☐ Siempre visible Reiniciar



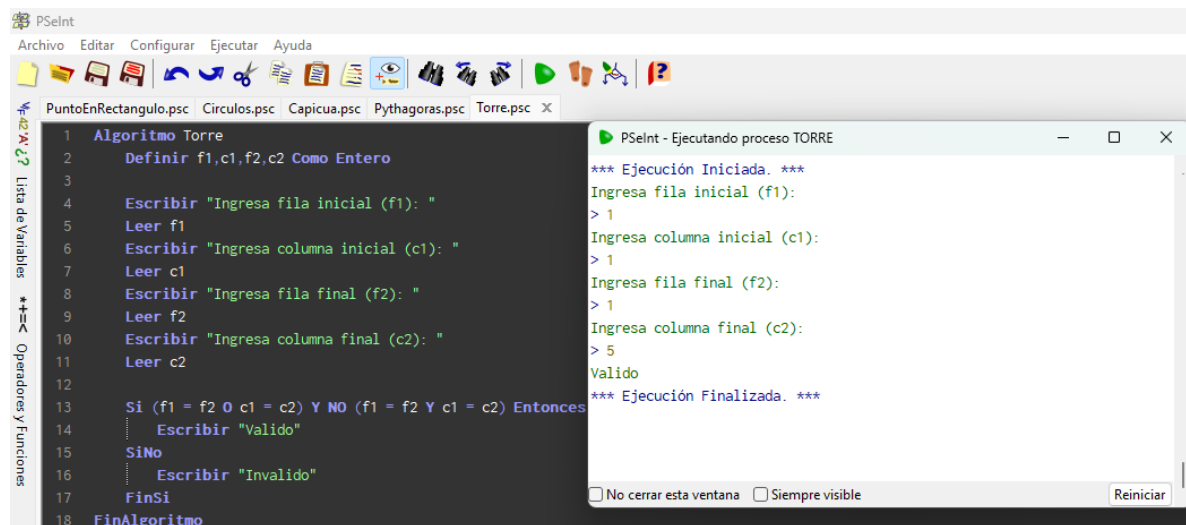
Problema 8 capicua:



Problema 9 pitagoras:

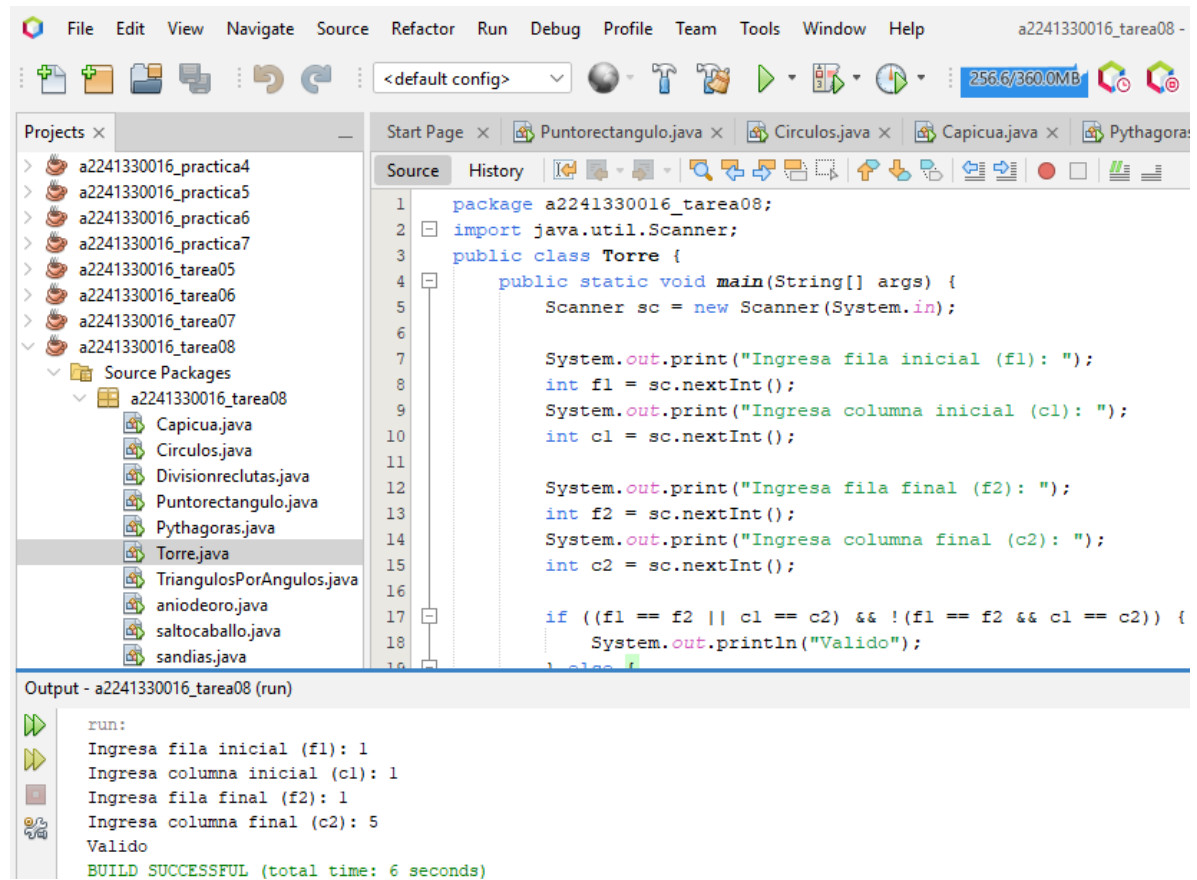


Problema 10 Torre:



The screenshot shows the PSeInt IDE with a Java program for the Tower of Hanoi problem. The program prompts the user for initial and final file and column numbers. The execution window shows the following output:

```
*** Ejecución Iniciada. ***
Ingresa fila inicial (f1):
> 1
Ingresa columna inicial (c1):
> 1
Ingresa fila final (f2):
> 1
Ingresa columna final (c2):
> 5
Valido
*** Ejecución Finalizada. ***
```



The screenshot shows an IDE with the source code of the Tower of Hanoi program. The code is as follows:

```
package a2241330016_tarea08;
import java.util.Scanner;
public class Torre {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Ingresa fila inicial (f1): ");
        int f1 = sc.nextInt();
        System.out.print("Ingresa columna inicial (c1): ");
        int c1 = sc.nextInt();

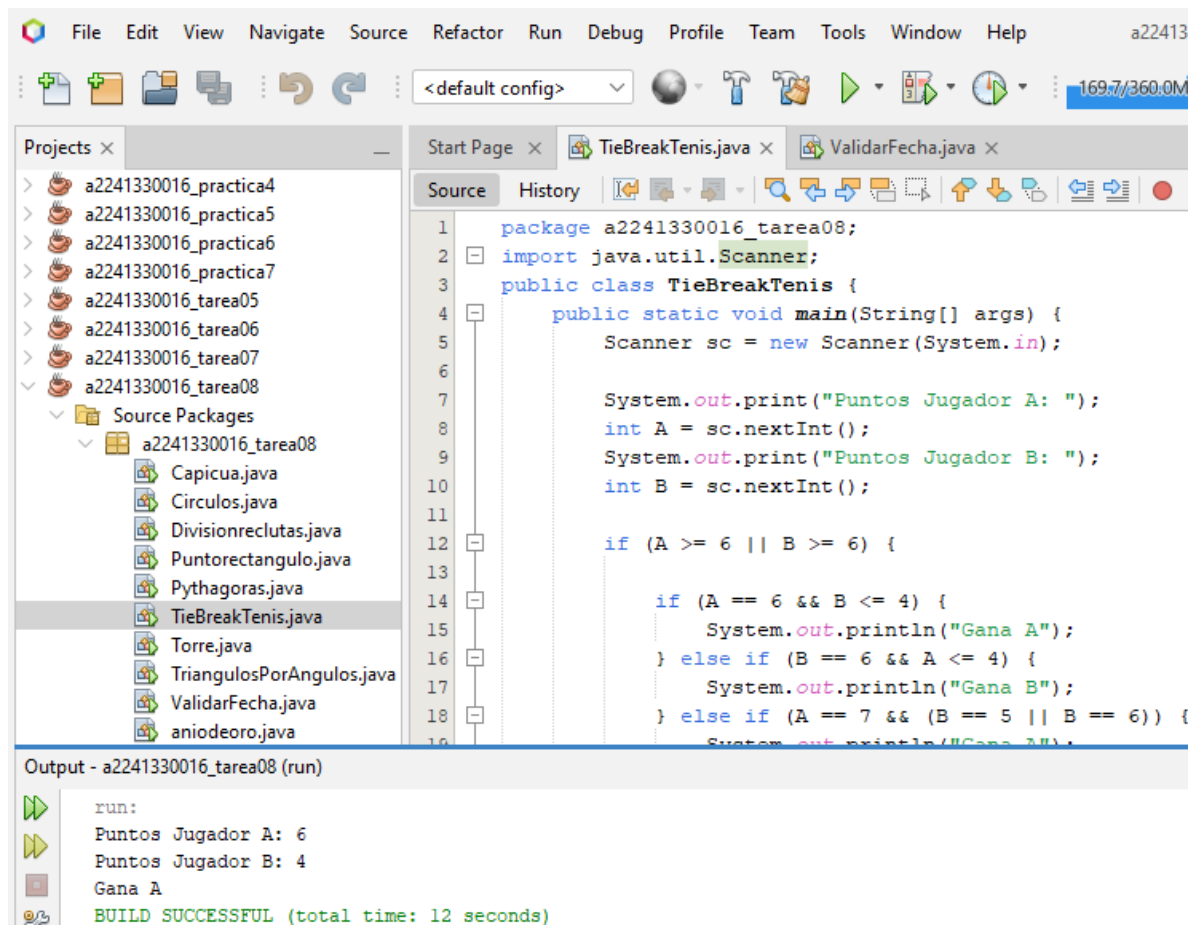
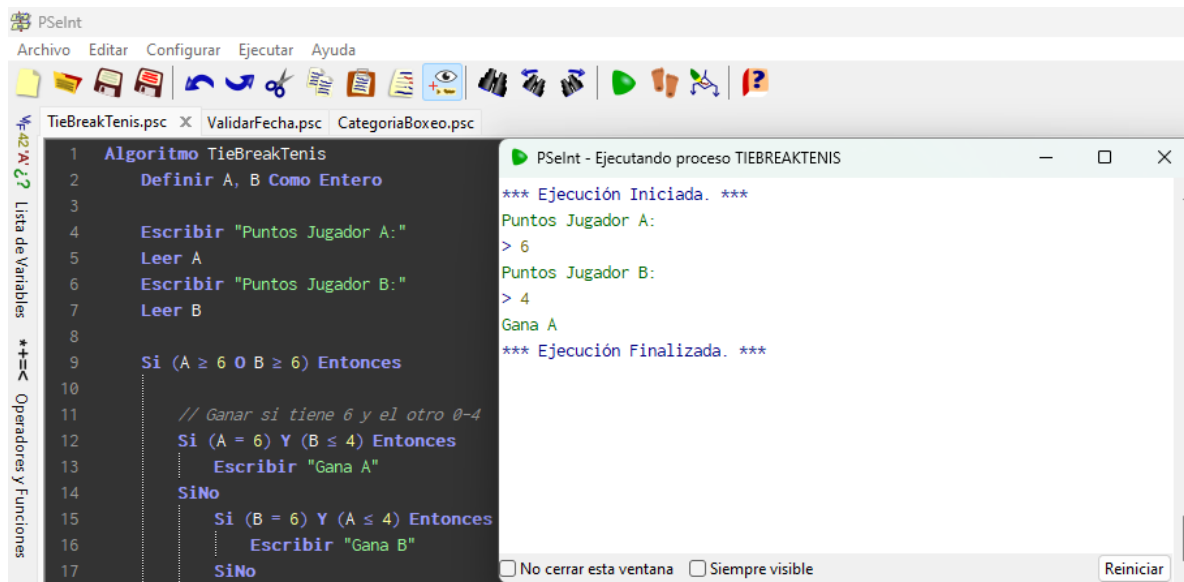
        System.out.print("Ingresa fila final (f2): ");
        int f2 = sc.nextInt();
        System.out.print("Ingresa columna final (c2): ");
        int c2 = sc.nextInt();

        if ((f1 == f2 || c1 == c2) && !(f1 == f2 && c1 == c2)) {
            System.out.println("Valido");
        }
    }
}
```

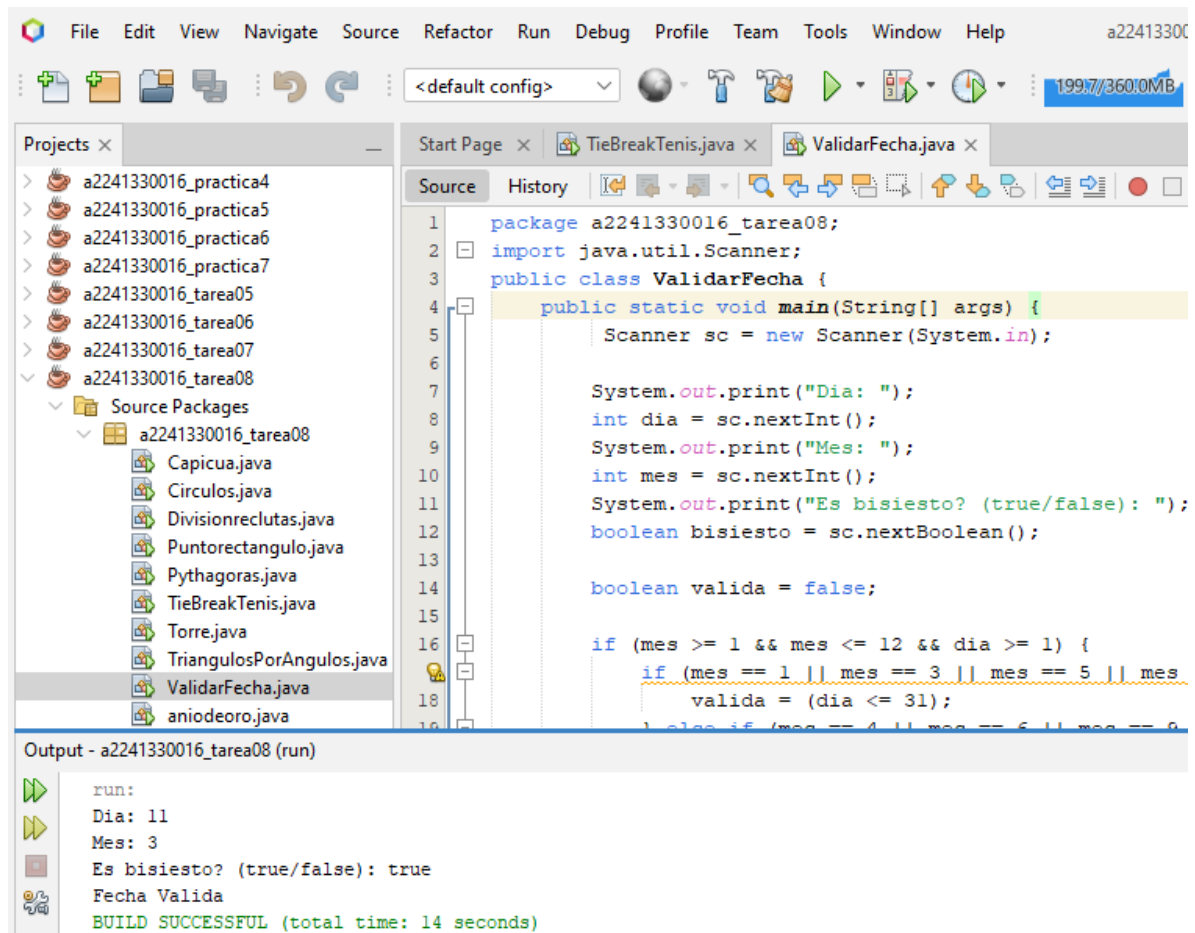
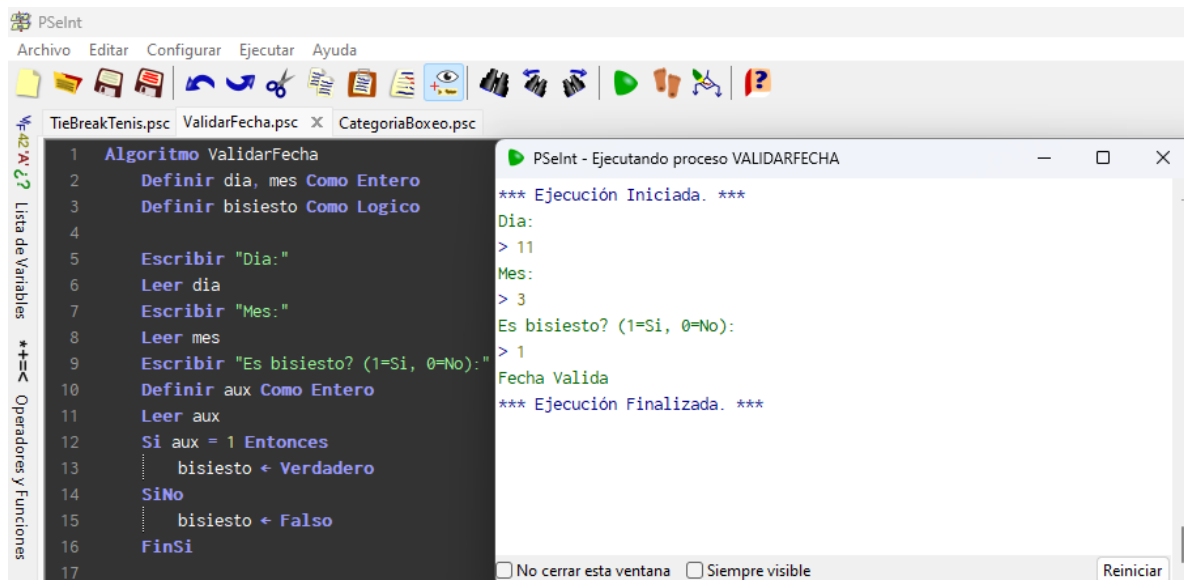
The output window shows the following execution results:

```
run:
Ingresa fila inicial (f1): 1
Ingresa columna inicial (c1): 1
Ingresa fila final (f2): 1
Ingresa columna final (c2): 5
Valido
BUILD SUCCESSFUL (total time: 6 seconds)
```

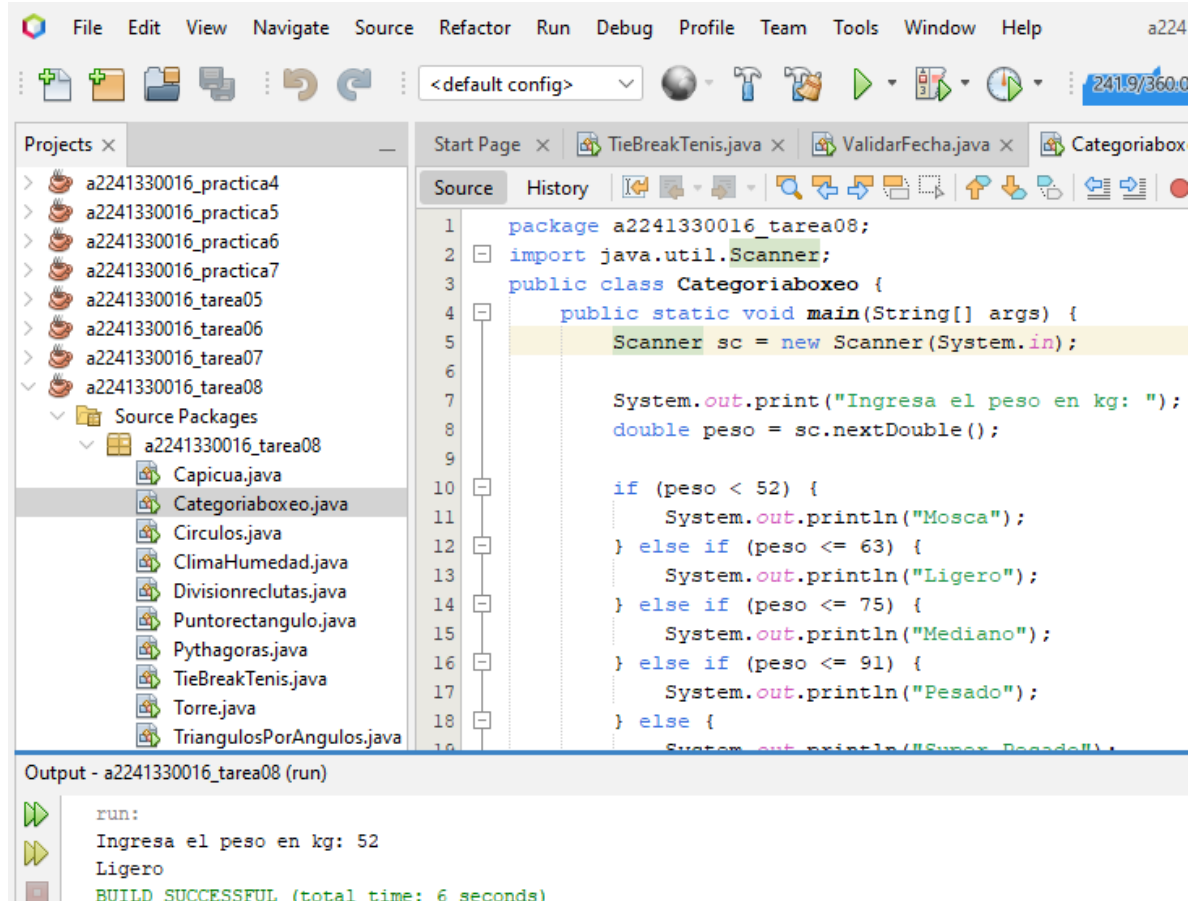
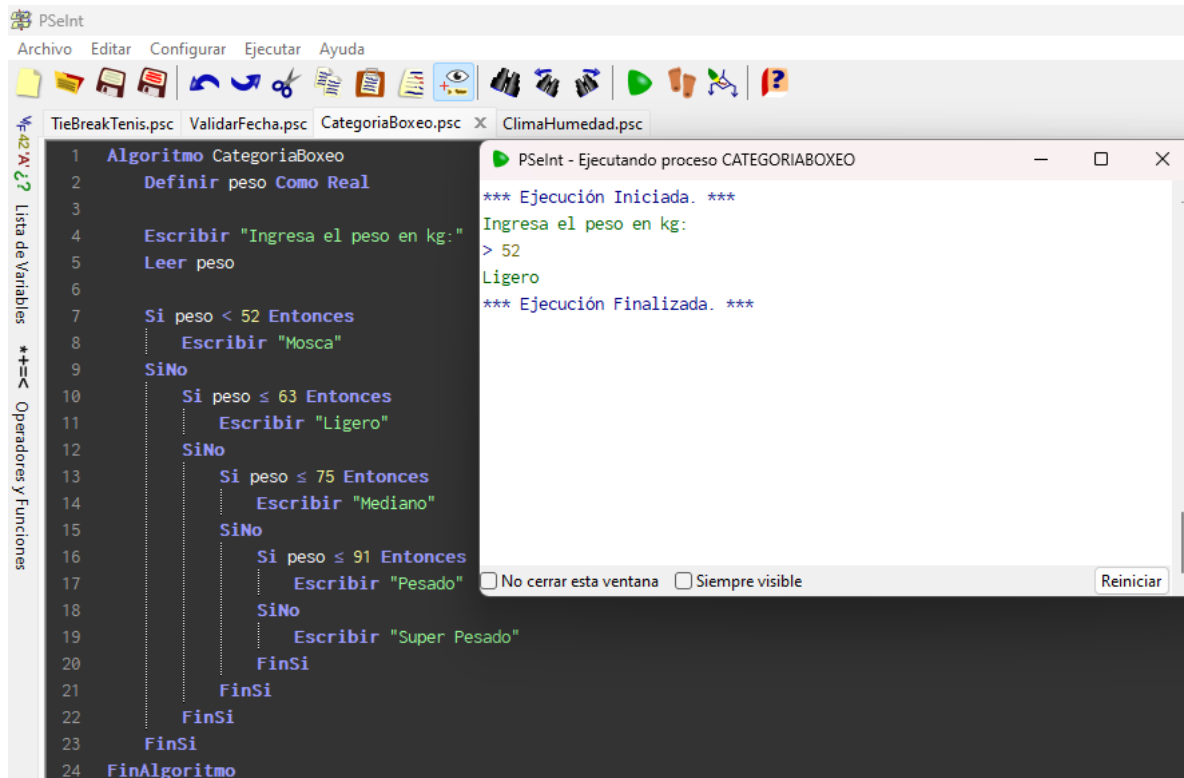
Problema 11 TieBreakTennis:



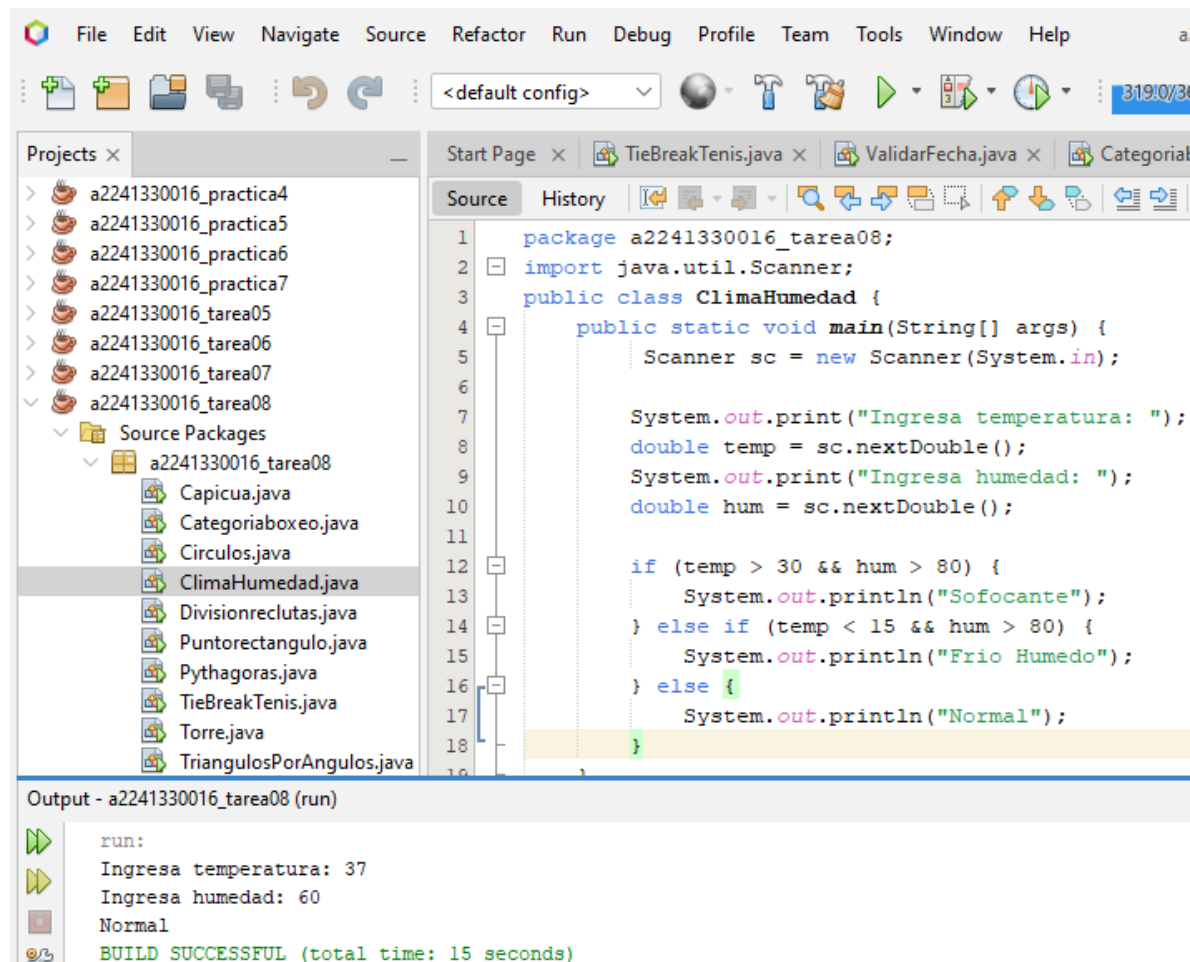
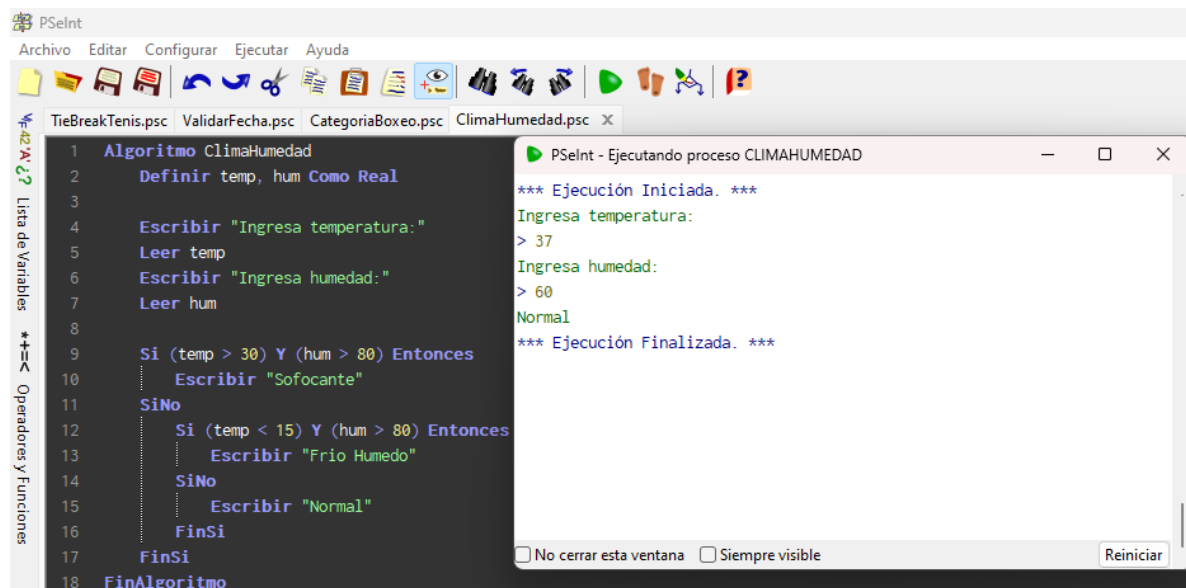
Problema 12 Validar fecha:



Problema 13 categoria boxeo:



Problema 14 Clima Humedad:



Problema 15 Sistema Medallas:

PSeInt

Archivo Editar Configurar Ejecutar Ayuda

SistemaMedallas.psc

```
1 Algoritmo SistemaMedallas
2   Definir n1, n2, n3, promedio Como Real
3
4   Escribir "Ingresa la calificacion 1:"
5   Leer n1
6   Escribir "Ingresa la calificacion 2:"
7   Leer n2
8   Escribir "Ingresa la calificacion 3:"
9   Leer n3
10
11   promedio ← (n1 + n2 + n3) / 3
12
13   Si promedio > 90 Entonces
14       Si (n1 = 100) O (n2 = 100) O (n3 = 100) Entonces
15           Escribir "Oro con Honores"
16       SiNo
17           Escribir "Oro"
18       FinSi
19   SiNo
20       Si promedio ≥ 80 Entonces
21           Escribir "Plata"
22       SiNo
23           Escribir "Sin Medalla"
24       FinSi
25   FinSi
26   FinSi
27   FinSi
28 FinAlgoritmo
```

PSeInt - Ejecutando proceso SISTEMAMEDALLAS

*** Ejecución Iniciada. ***

Ingresa la calificacion 1:
> 6

Ingresa la calificacion 2:
> 6

Ingresa la calificacion 3:
> 6

Sin Medalla

*** Ejecución Finalizada. ***

☐ No cerrar esta ventana ☐ Siempre visible Reiniciar

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help a22413300

<default config> 161.9/360.0MB

Projects x

- > a2241330016_practica4
- > a2241330016_practica5
- > a2241330016_practica6
- > a2241330016_practica7
- > a2241330016_tarea05
- > a2241330016_tarea06
- > a2241330016_tarea07
- ▼ a2241330016_tarea08
 - Source Packages
 - ▼ a2241330016_tarea08
 - Capicua.java
 - Categoriaboxeo.java
 - Circulos.java
 - ClimaHumedad.java
 - Divisionreclutas.java
 - Puntorectangulo.java
 - Pythagoras.java
 - SistemaMedallas.java
 - TieBreakTenis.java
 - Torre.java

Start Page x SistemaMedallas.java x

Source History

```
1 package a2241330016_tarea08;
2 import java.util.Scanner;
3 public class SistemaMedallas {
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6
7         System.out.print("Ingresa la calificacion 1: ");
8         double n1 = sc.nextDouble();
9
10        System.out.print("Ingresa la calificacion 2: ");
11        double n2 = sc.nextDouble();
12
13        System.out.print("Ingresa la calificacion 3: ");
14        double n3 = sc.nextDouble();
15
16        double promedio = (n1 + n2 + n3) / 3;
17
18        if (promedio > 90) {
19
```

Output - a2241330016_tarea08 (run)

run:

```
Ingresa la calificacion 1: 6
Ingresa la calificacion 2: 6
Ingresa la calificacion 3: 6
Sin Medalla
BUILD SUCCESSFUL (total time: 11 seconds)
```