

Ejercicio 1) Lotería

Un apostador necesita procesar los 20 premios de la lotería, para luego conocer el promedio de todos los impares, el mayor, menor y la cantidad de los números pares que han salido sorteados.

Desarrolle una aplicación que permita ingresar los 20 números y presente los resultados.

Solución:

Análisis:

Entrada:

- 20 Números premiados de la lotería

Salida:

- Promedio de números impares
- Mayor número ingresado
- Menor número ingresado
- Cantidad de Números pares

Procesos:

$$\text{Número } N \bmod 2 = \text{Resto} \quad \left\{ \begin{array}{l} \text{"N es Impar", Resto } \neq 0 \\ \text{"N es Par", Resto } = 0 \end{array} \right.$$

Número mayor $M = \{ M \geq n, \forall n / n \text{ y } M \in \text{Números ingresados} \}$

Número menor $m = \{ m \leq n, \forall n / n \text{ y } m \in \text{Números ingresados} \}$

Estrategia:

1. Inicializar *cPar* en 0
2. Inicializar *cImpar* en 0
3. Inicializar *sumaImpar* en 0
4. Solicitar el primer número ganador
5. Asignar el número a *mayor*
6. Asignar el número a *menor*
7. Verificar si número es par:
 - a. Si se verifica se incrementa *cPar* en 1.
 - b. Si no verifica se asigna el número a *sumaImpar* se incrementa *cImpar* en 1.
8. Repetir 19 veces:
 - a. Solicitar el siguiente número ganador.
 - b. Verificar si el numero es par:
 - i. Si se verifica se incrementa *cPar* en 1.
 - ii. Si no verifica, se incrementa *cImpar* en 1 y se suma el número a *sumaImpar*.
 - c. Verificar si el número es mayor a *mayor*:
 - i. Si se verifica se asigna el número a *mayor*.
 - d. Verificar si el número es menor a *menor*:
 - i. Si se verifica se asigna el número a *menor*.
9. Calcular el *promedio* de impares con *sumaImpar* y *cImpar*.
10. Mostrar por pantalla, *promedio*, *cPar*, *mayor* y *menor*.

Ambiente:

Variable	Tipo	Descripción
num	Entero	Número ingresado
cPar	Entero	Cantidad de números pares ingresados
cImpar	Entero	Cantidad de números impares ingresados
sumaImpar	Entero	Suma de los números impares ingresados
max	Entero	Mayor número ingresado
min	Entero	Menor número ingresado

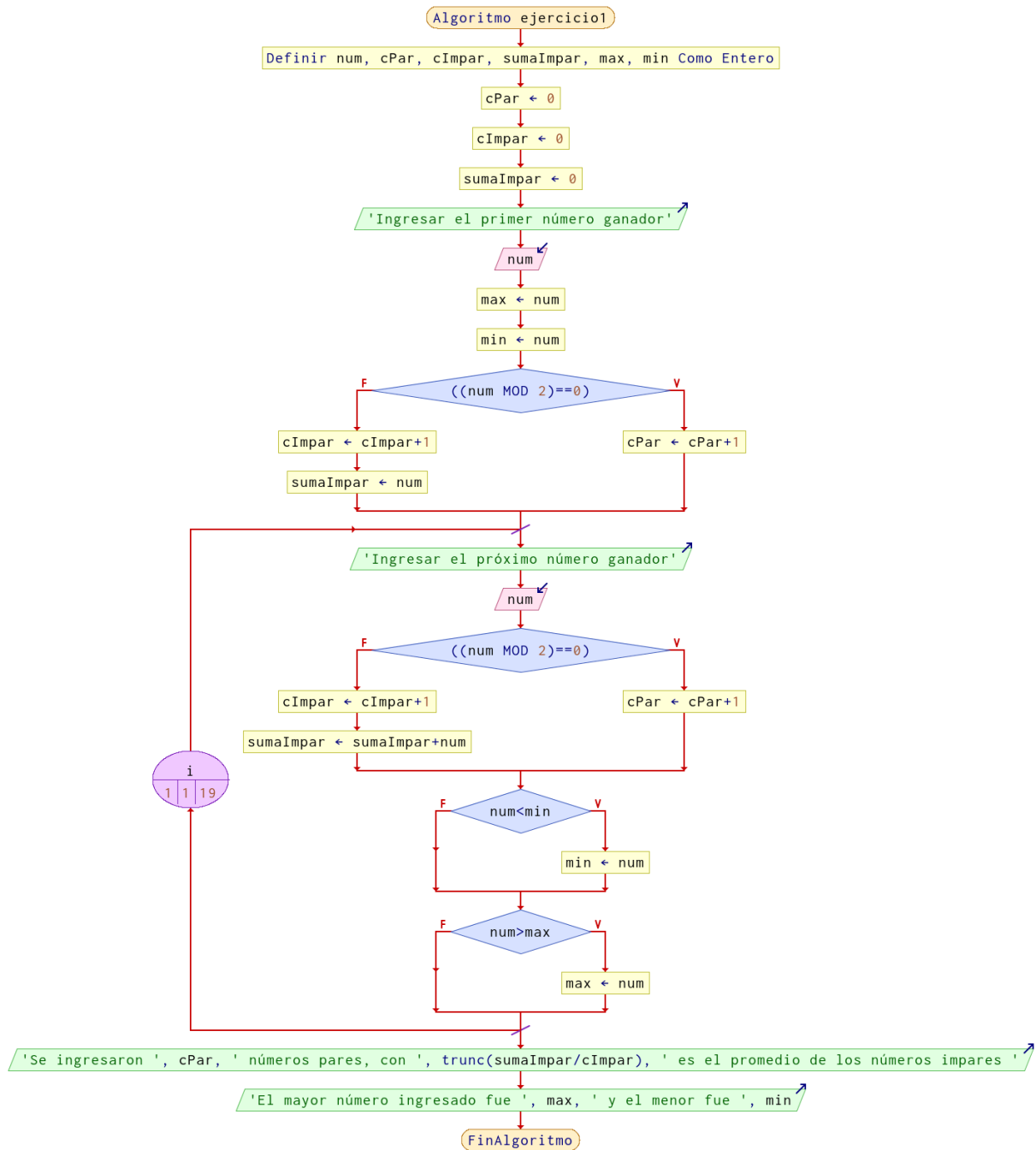
Algoritmo:

```
Algoritmo ejercicio1
    Definir num, cPar, cImpar, sumaImpar, max, min Como Entero
    cPar <- 0;
    cImpar <- 0;
    sumaImpar <- 0;

    Escribir "Ingresar el primer número ganador";
    Leer num;
    max <- num;
    min <- num;
    Si ((num mod 2) == 0) Entonces
        cPar <- cPar + 1;
    Sino
        cImpar <- cImpar + 1;
        sumaImpar <- sumaImpar + num;
    FinSi
    Para i<-1 Hasta 19 Con Paso 1 Hacer
        Escribir "Ingresar el próximo número ganador";
        Leer num;
        Si ((num mod 2) == 0) Entonces
            cPar <- cPar + 1;
        Sino
            cImpar <- cImpar + 1;
            sumaImpar <- sumaImpar + num;
        FinSi
        Si num < min Entonces
            min <- num
        FinSi
        Si num > max Entonces
            max <- num
        FinSi
    Fin Para
    Escribir "Se ingresaron ",cPar," números pares, con
    ",trunc(sumaImpar/cImpar) ," es el promedio de los números impares "
```

Escribir "El mayor número ingresado fue ", max," y el menor fue ", min
FinAlgoritmo

Diagrama de Flujo:



Seguimiento:

n°	num	cPar	cImpar	sumaImpar	max	min	Salida
1		0	0	0			"Ingresar el primer número ganador"
2	3539	0	0	0			
3	3539	0	1	3539	3539	3539	
4	3539	0	1	3539	3539	3539	"Ingresar el próximo número ganador"
5	1463	0	1	3539	3539	3539	
6	1463	0	2	5002	3539	1463	
7	1463	0	2	5002	3539	1463	"Ingresar el próximo número ganador"
8	1471	0	2	5002	3539	1463	
9	1471	0	3	6473	3539	1463	
10	1471	0	3	6473	3539	1463	"Ingresar el próximo número ganador"
11	3936	0	3	6473	3539	1463	
12	3936	1	3	6473	3936	1463	
13	3936	1	3	6473	3936	1463	"Ingresar el próximo número ganador"
14	3154	1	3	6473	3936	1463	
15	3154	2	3	6473	3936	1463	
16	3154	2	3	6473	3936	1463	"Ingresar el próximo número ganador"
17	8740	2	3	6473	3936	1463	
18	8740	3	3	6473	8740	1463	
19	8740	3	3	6473	8740	1463	"Ingresar el próximo número ganador"
20	3730	3	3	6473	8740	1463	
21	3730	4	3	6473	8740	1463	
22	3730	4	3	6473	8740	1463	"Ingresar el próximo número ganador"
23	2342	4	3	6473	8740	1463	
24	2342	5	3	6473	8740	1463	
25	2342	5	3	6473	8740	1463	"Ingresar el próximo número ganador"
26	5528	5	3	6473	8740	1463	
27	5528	6	3	6473	8740	1463	

28	5528	6	3	6473	8740	1463	"Ingresar el próximo número ganador"
29	9731	6	3	6473	8740	1463	
30	9731	6	4	16204	9731	1463	
31	9731	6	4	16204	9731	1463	"Ingresar el próximo número ganador"
32	3497	6	4	16204	9731	1463	
33	3497	6	5	19701	9731	1463	
34	3497	6	5	19701	9731	1463	"Ingresar el próximo número ganador"
35	6720	6	5	19701	9731	1463	
36	6720	7	5	19701	9731	1463	
37	6720	7	5	19701	9731	1463	"Ingresar el próximo número ganador"
38	4044	7	5	19701	9731	1463	
39	4044	8	5	19701	9731	1463	
40	4044	8	5	19701	9731	1463	"Ingresar el próximo número ganador"
41	9985	8	5	19701	9731	1463	
42	9985	8	6	29686	9985	1463	
43	9985	8	6	29686	9985	1463	"Ingresar el próximo número ganador"
44	9266	8	6	29686	9985	1463	
45	9266	9	6	29686	9985	1463	
46	9266	9	6	29686	9985	1463	"Ingresar el próximo número ganador"
47	6642	9	6	29686	9985	1463	
48	6642	10	6	29686	9985	1463	
49	6642	10	6	29686	9985	1463	"Ingresar el próximo número ganador"
50	0579	10	6	29686	9985	1463	
51	0579	10	7	30265	9985	0579	
52	0579	10	7	30265	9985	0579	"Ingresar el próximo número ganador"
53	3446	10	7	30265	9985	0579	
54	3446	11	7	30265	9985	0579	
55	3446	11	7	30265	9985	0579	"Ingresar el próximo número ganador"
56	2744	11	7	30265	9985	0579	

57	2744	12	7	30265	9985	0579	
58	2744	12	7	30265	9985	0579	"Ingresar el próximo número ganador"
59	7476	12	7	30265	9985	0579	
60	7476	13	7	30265	9985	0579	
61	7476	13	7	30265	9985	0579	"Se ingresaron 13 números pares, con 4323 es el promedio de los números impares. El mayor número ingresado fue 9985 y el menor fue 579"

Ejercicio 1) Parciales

Dadas las notas de 3 parciales, calcular promedio y decir si promocionó o rinde final.

Solución:

Análisis:

Entrada:

notas de los 3 parciales

Salida:

{ "Promocionó", "Rinde final" }

Procesos:

Promedio = $\{(Nota1 + Nota2 + Nota3) / 3\}$

Condición = $\begin{cases} \text{"Promocionó",} & \text{Promedio} \geq 80 \\ \text{"Rinde Final",} & \text{Promedio} < 80 \end{cases}$

Estrategia:

1. Solicitar las notas de los 3 parciales
2. Calcular el promedio de los tres parciales.
3. Verificar que el promedio es igual o superior a 80.
 - a. Si verificar, mostrar el mensaje "Promociona la materia".
 - b. Si no verifica, mostrar el mensaje "Debe rendir final para aprobar la materia"

Ambiente:

Variable	Tipo	Descripción
nota1	Entero	Nota del 1er Parcial

nota2	Entero	Nota del 2do Parcial
nota3	Entero	Nota del 3er Parcial
promedio	Real	Promedio de los tres parciales

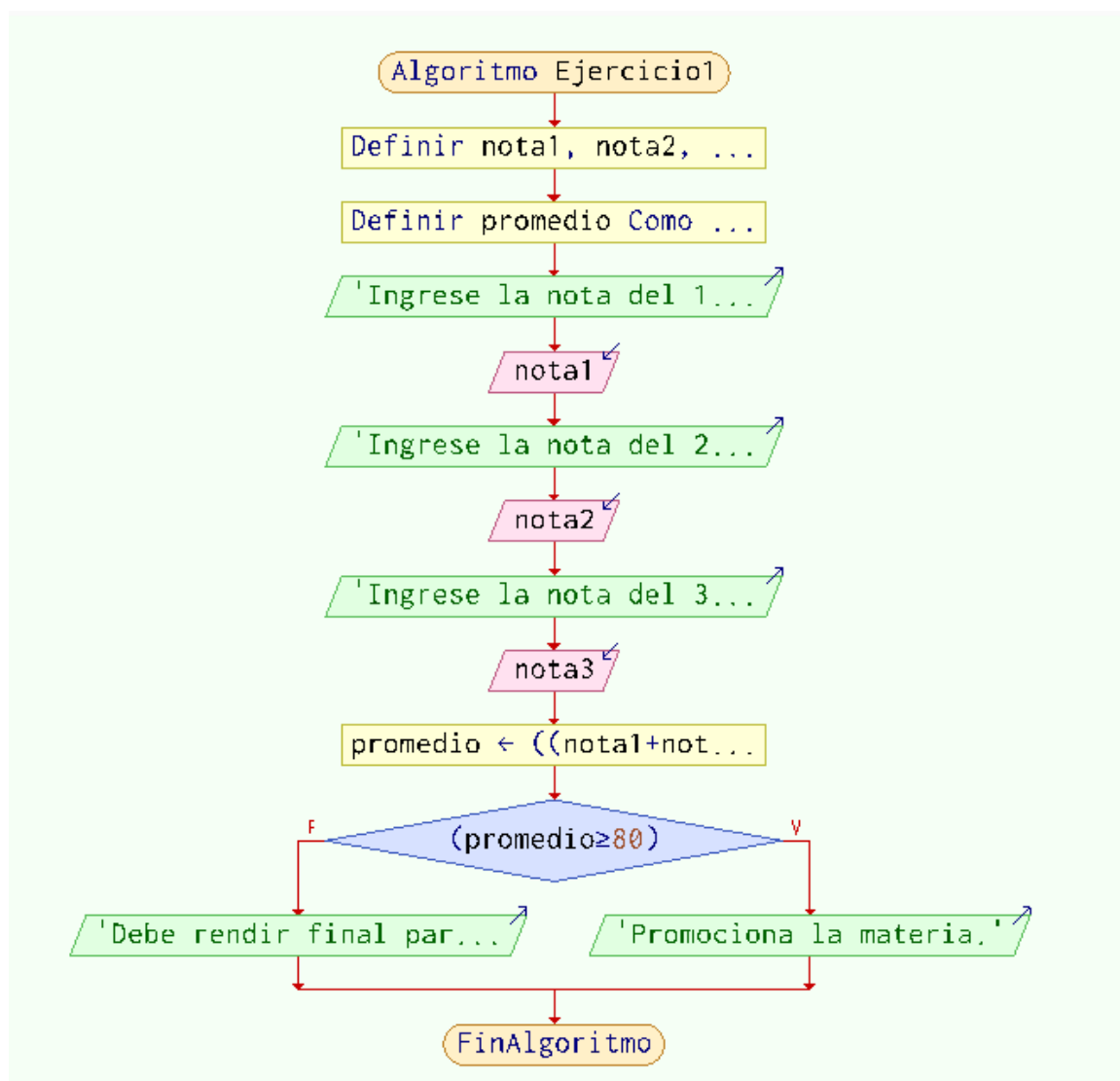
Algoritmo:

```

Algoritmo Ejercicio1
    Definir nota1, nota2, nota3 Como Entero
    Definir promedio Como Real
    Escribir 'Ingrese la nota del 1er Parcial: '
    Leer nota1
    Escribir 'Ingrese la nota del 2do Parcial: '
    Leer nota2
    Escribir 'Ingrese la nota del 3er Parcial: '
    Leer nota3
    promedio <- ((nota1+nota2+nota3)/3)
    Si (promedio>=80) Entonces
        Escribir 'Promociona la materia.'
    SiNo
        Escribir 'Debe rendir final para promocionar la materia'
    FinSi
FinAlgoritmo

```

Diagrama de Flujo:



Sequimiento:

Nº	nota1	nota2	nota3	promedio	Salida
1					"Ingrese nota del 1er Parcial"
2	90				
3	90				"Ingrese nota del 2do Parcial"
4	90	85			
5	90	85			"Ingrese nota del 3er Parcial"
6	90	85	70		
7	90	85	70	81,6667	
8	90	85	70	81,6667	"Promociona la materia"

Ejercicio 2) Par o impar

Se necesita un algoritmo que informe si un número ingresado es PAR o IMPAR mediante un mensaje.

Solución:

Análisis:

Entrada:

número a verificar

Salida:

{ "Es par", "Es impar" }

Procesos:

resto = número % 2

Condición = $\left\{ \begin{array}{ll} \text{"Es par",} & \text{resto} == 0 \\ \text{"Es impar",} & \text{resto} <> 0 \end{array} \right.$

Estrategia:

1. Solicitar el número a verificar
2. Calcular el resto de la división del número por 2
3. Verificar si el resto es igual a 0
 - a. Si verifica, mostrar el mensaje "Es par"
 - b. Si no verifica, mostrar el mensaje "Es impar"

Ambiente:

Variable	Tipo	Descripción
num	Entero	Número a verificar
resto	Entero	Resto de la división del número por dos

Algoritmo:

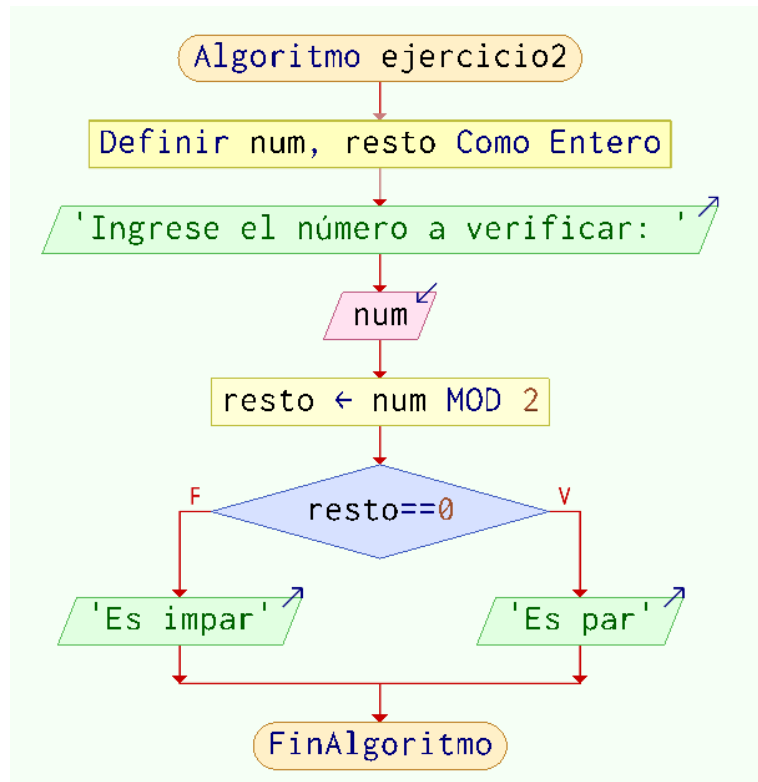
```
Algoritmo ejercicio2
    Definir num, resto Como Entero

    Escribir "Ingrese el número a verificar: ";
    Leer num;

    resto <- num % 2;
    Si resto == 0 Entonces
        Escribir "Es par";
    SiNo
        Escribir "Es impar";
    Fin Si

FinAlgoritmo
```

Diagrama de Flujo:



Seguimiento:

Nº	num	resto	Salida
1			"Ingrese numero a verificar"
2	55		
3	55	1	
4	55	1	"Es impar"

Ejercicio 3) Positivo, negativo o cero

Se desea saber si el número ingresado es positivo, negativo o cero.

Solución:

Análisis:

Entrada:

número a verificar

Salida:

{ "Es cero", "Es Negativo", "Es Positivo" }

Procesos:

Resultado = {

- "Es cero", numero == 0
- "Es Positivo", numero > 0
- "Es Negativo", numero < 0

Estrategia:

1. Solicitar el número a verificar
2. Verificar si el número es distinto a 0
 - a. Si verifica, verificar si es mayor a 0
 - i. Si verifica, mostrar "Es positivo"
 - ii. Si no verifica, mostrar "Es negativo"
 - b. Si no verifica, es igual a 0 y mostrar "Es cero"

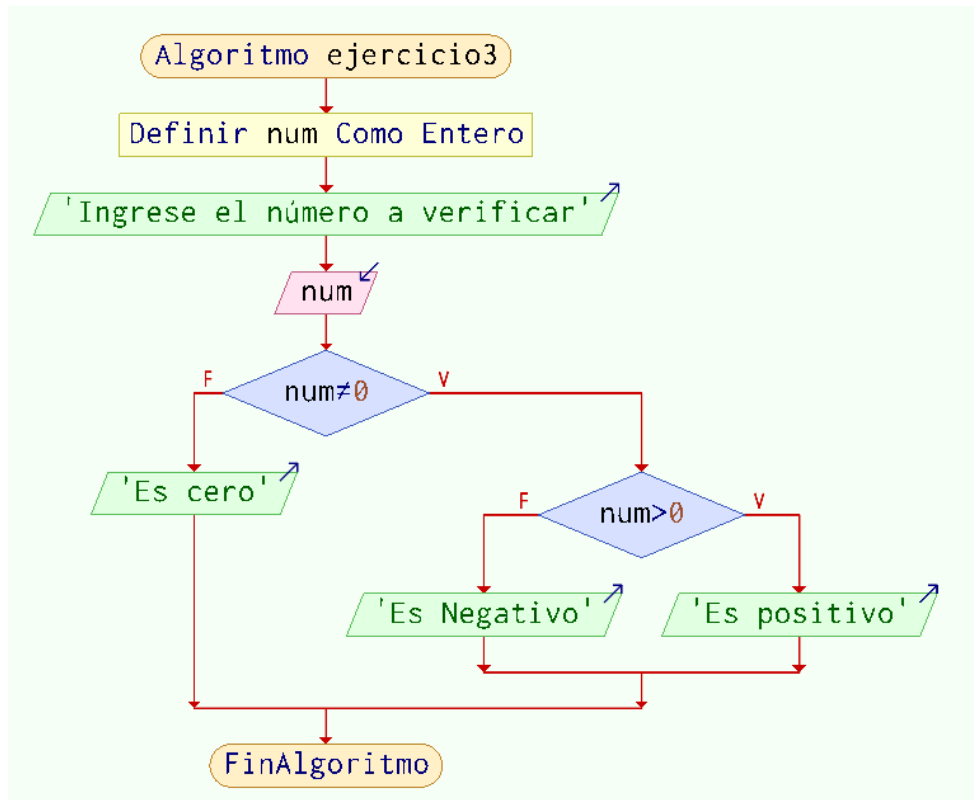
Ambiente:

Variable	Tipo	Descripción
num	Entero	Número a verificar

Algoritmo:

```
Algoritmo ejercicio3
  Definir num Como Entero
  Escribir "Ingrese el número a verificar"
  Leer num
  Si num <> 0 Entonces
    Si num > 0 Entonces
      Escribir "Es positivo"
    SiNo
      Escribir "Es Negativo"
    Fin Si
  SiNo
    Escribir "Es cero"
  Fin Si
FinAlgoritmo
```

Diagrama de Flujo:



Seguimiento:

Nº	num	Salida
1		"Ingrese numero a verificar"
2	0	
3	0	"Es cero"

Ejercicio 4) Triángulo

Se requiere de un algoritmo que permita determinar si 3 segmentos de recta pueden formar un triángulo.

Nota: En cualquier triángulo el mayor de los lados es menor que la suma de los restantes, o en general, la suma de 2 lados debe ser mayor que el lado restante.

Solución:

Análisis:

Entrada:

Tres segmentos de recta.

Salida:

{ "Puede ser Triángulo", "No puede ser Triangulo" }

Procesos:

Resultado = $\left\{ \begin{array}{ll} \text{"Puede ser triángulo",} & \text{Lado Mayor} < \text{Suma lados menores} \\ \text{"No puede ser triángulo",} & \text{Lado Mayor} > \text{Suma lados menores} \end{array} \right.$

Estrategia:

1. Solicitar segmento 1 del supuesto triángulo
2. Toma el segmento 1 como el Lado mayor
3. Solicitar segmento 2 del supuesto triángulo
4. Verificar si el segmento 2 es mayor a nuestro Lado mayor
 - a. Si verifica, segmento 2 es el nuevo Lado mayor
5. Solicitar segmento 3 del supuesto triángulo
6. Verificar si el segmento 3 es mayor a nuestro Lado mayor
 - a. Si verifica, segmento 3 es el nuevo Lado mayor
7. Verificar si la suma de todos los lados menos el Lado mayor es mayor al Lado mayor.
 - a. Si verifica, mostrar "Puede ser triángulo"
 - b. Si no verifica, mostrar "No puede ser triángulo"

Ambiente:

Variable	Tipo	Descripción
s1	Entero	1er Segmento de recta
s2	Entero	2do Segmento de recta
s3	Entero	3er Segmento de recta
ladoMayor	Entero	Lado Mayor

Algoritmo:

Algoritmo ejercicio4

Definir s1, s2, s3, ladoMayor Como Entero

Escribir "Ingresar 1er segmento de recta"

Leer s1;

ladoMayor \leftarrow s1;

Escribir "Ingresar 2do segmento de recta"

Leer s2;

Si (s2 > ladoMayor) Entonces

 ladoMayor \leftarrow s2;

Fin Si

Escribir "Ingresar 3er segmento de recta"

Leer s3;

Si (s3 > ladoMayor) Entonces

 ladoMayor \leftarrow s3;

Fin Si

Si ((s1+s2+s3-ladoMayor) > ladoMayor) Entonces

 Escribir "Puede ser triángulo"

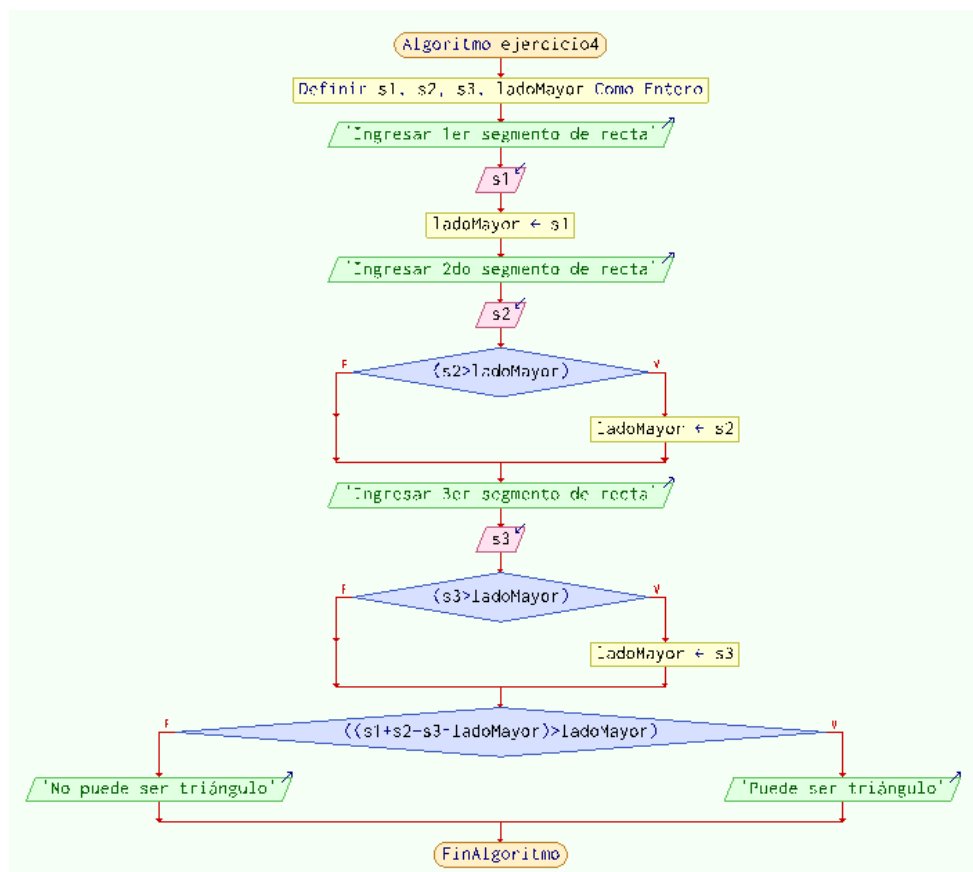
SiNo

 Escribir "No puede ser triángulo"

Fin Si

FinAlgoritmo

Diagrama de Flujo:



Seguimiento:

Nº	s1	s2	s3	ladoMayor	Salida
1					"Ingrese 1er segmento de recta"
2	4				
3	4			4	
4	4			4	"Ingrese 2do segmento de recta"
5	4	5		4	
6	4	5		5	
7	4	5		5	"Ingrese 3er segmento de recta"
8	4	5	3	5	
9	4	5	3	5	
10	4	5	3	5	"Puede ser un triangulo"

Ejercicio 5) Números de libreta

Se ingresa el nombre y Nro de libreta de 3 alumnos. Muestre la lista ordenada por Nro de libreta.

Solución:

Análisis:

Entrada:

Tres nombres y números de libretas.

Salida:

Lista Ordenada de los alumnos.

Estrategia:

1. Solicitar nombre del primer alumno.
2. Solicitar nº de libreta del primer alumno.
3. Asignar nombre de alumno como primerNombre
4. Asignar nº de libreta como primerLibreta
5. Solicitar nombre del segundo alumno.
6. Solicitar nº de libreta del segundo alumno.
7. Verificar que nº de libreta es menor a primerLibreta.
 - a. Si verifica, se asigna primerLibreta a segundaLibreta y nº de libreta queda como primerLibreta
8. Solicitar nombre del tercer alumno.
9. Solicitar nº de libreta del tercer alumno.
10. Verificar que nº de libreta es menor a segundaLibreta.
 - a. Verificar si nº de libreta es menor a primerLibreta
 - i. Si verifica, se asigna primerLibreta a segundaLibreta, segundaLibreta a tercerLibreta y nº de Libreta a primerLibreta.
 - ii. Si no verifica, se asigna segundaLibreta a tercerLibreta y nº de libreta a segundaLibreta

11. Se muestran los nombres y numero de primerLibreta
12. Se muestran los nombres y numero de segundaLibreta
13. Se muestran los nombres y número de tercerLibreta

Ambiente:

Variable	Tipo	Descripción
l1	Entero	Nº de Libreta de primer alumno
l2	Entero	Nº de Libreta de segundo alumno
l3	Entero	Nº de Libreta de tercer Alumno
auxL	Entero	Variable Auxiliar para ordenamiento de lista
n1	Cadena	Nombre de primer Alumno
n2	Cadena	Nombre de segundo Alumno
n3	Cadena	Nombre de tercer Alumno
auxN	Cadena	Variable Auxiliar para ordenamiento de lista

Algoritmo:

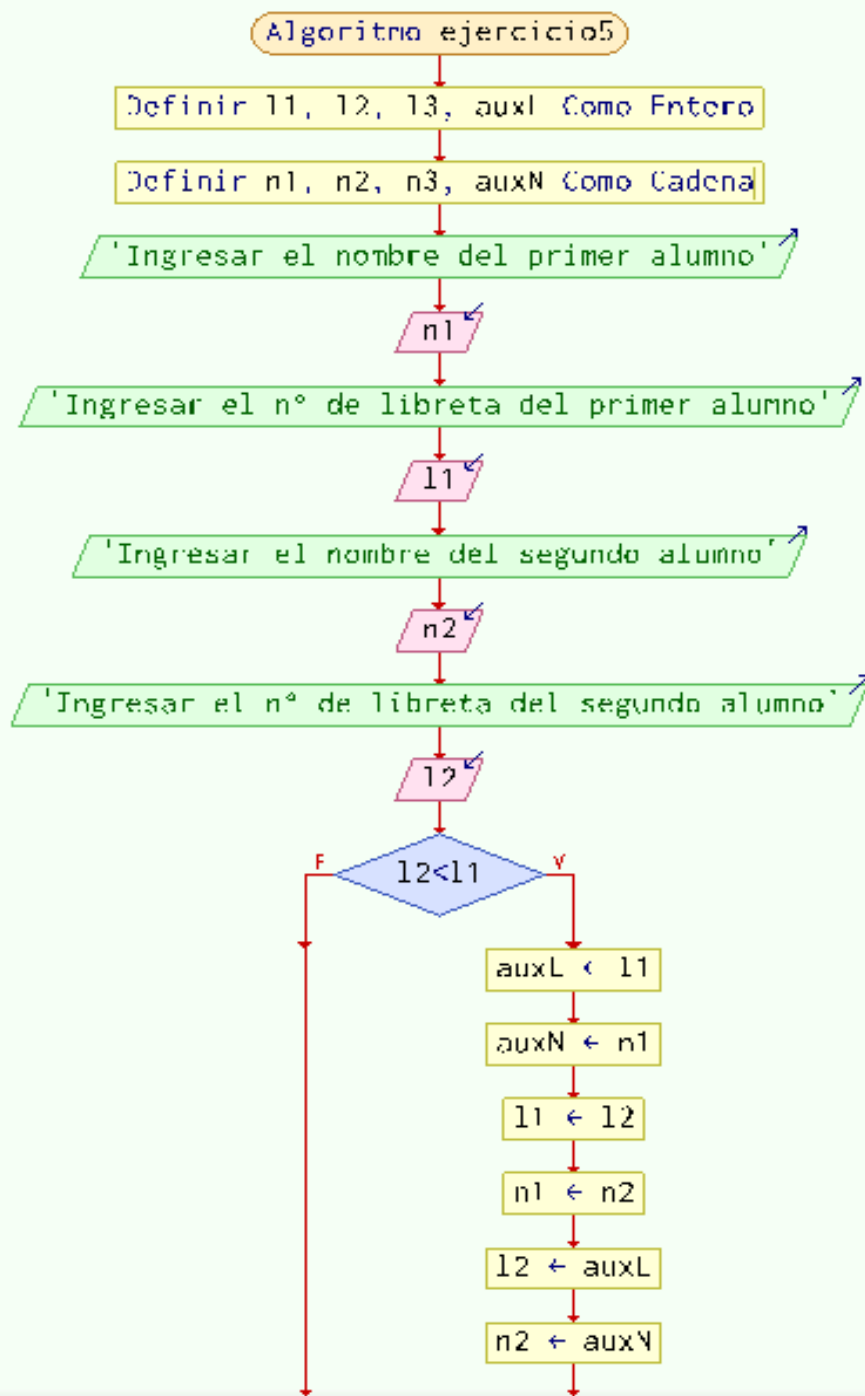
Algoritmo ejercicio5

```
Definir l1,l2,l3, auxL Como Entero
Definir n1,n2,n3, auxN Como Caracter

Escribir "Ingresar el nombre del primer alumno"
Leer n1
Escribir "Ingresar el n° de libreta del primer alumno"
Leer l1;
Escribir "Ingresar el nombre del segundo alumno"
Leer n2
Escribir "Ingresar el n° de libreta del segundo alumno"
Leer l2;
Si l2 < l1 Entonces
    auxL <- l1;
    auxN <- n1;
    l1 <- l2;
    n1 <- n2;
    l2 <- auxL;
    n2 <- auxN;
Fin Si
Escribir "Ingresar el nombre del tercer alumno"
Leer n3
Escribir "Ingresar el n° de libreta del tercer alumno"
Leer l3;
Si l3 < l2 Entonces
    Si l3 < l1 Entonces
        auxL <- l1;
        auxN <- n1;
        l1 <- l3;
        n1 <- n3;
        l3 <- l2;
        n3 <- n2;
        l2 <- auxL
        n2 <- auxN
    SiNo
        auxL <- l2;
        auxN <- n2;
        l2 <- l3;
        n2 <- n3;
        l3 <- auxL;
        n3 <- auxN;
    Fin Si
Fin Si
Escribir "Libreta N° ", l1, " - ", n1;
Escribir "Libreta N° ", l2, " - ", n2;
Escribir "Libreta N° ", l3, " - ", n3;
```

FinAlgoritmo

Diagrama de flujo:



Sequimiento:

n°	l1	l2	l3	auxL	n1	n2	n3	auxN	Salida
1									(1)
2					"Juan"				
3					"Juan"				(2)
4	234				"Juan"				
5	234				"Juan"				(3)
6	234				"Juan"	"Jose"			
7	234				"Juan"	"Jose"			(4)
8	234	256			"Juan"	"Jose"			
9	234	256			"Juan"	"Jose"			
10	234	256			"Juan"	"Jose"			(5)
11	234	256			"Juan"	"Jose"	"Pedro"		
12	234	256			"Juan"	"Jose"	"Pedro"		(6)
13	234	256	124		"Juan"	"Jose"	"Pedro"		
14	234	256	124	234	"Juan"	"Jose"	"Pedro"	"Juan"	
15	124	256	256	234	"Pedro"	"Jose"	"Jose"	"Juan"	
16	124	234	256	234	"Pedro"	"Juan"	"Jose"	"Juan"	
17	124	234	256	234	"Pedro"	"Juan"	"Jose"	"Juan"	(7)
18	124	234	256	234	"Pedro"	"Juan"	"Jose"	"Juan"	(8)
19	124	234	256	234	"Pedro"	"Juan"	"Jose"	"Juan"	(9)

- (1) "Ingrese el nombre del primer alumno"
- (2) "Ingrese el numero de libreta del primer alumno"
- (3) "Ingrese el nombre del segundo alumno"
- (4) "Ingrese el numero de libreta del segundo alumno"
- (5) "Ingrese el nombre del tercer alumno"
- (6) "Ingrese el numero de libreta del tercer alumno"
- (7) "Libreta N° 124 - Pedro"
- (8) "Libreta N° 234 - Juan"
- (9) "Libreta N° 256 - Jose"

Ejercicio 6) Rectángulo

Realice un algoritmo que, tomando como datos la base y la altura de un rectángulo, informe si este es horizontal o vertical. Sin dejar de considerar el caso particular del cuadrado. Finalmente calcule el área de la figura.

Solución:

Análisis:

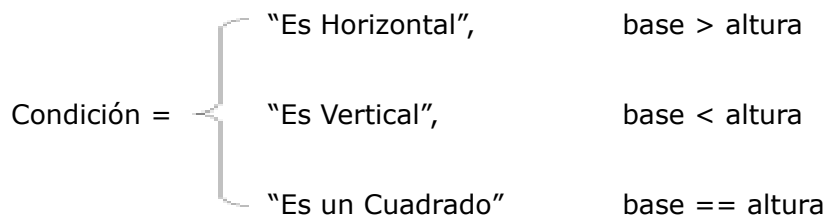
Entrada:

- Base del rectángulo
- Altura del rectángulo

Salida:

- { "Es Horizontal", "Es Vertical", "Es un Cuadrado" }
- Área del rectángulo

Procesos:



Estrategia:

1. Solicitar la base del rectángulo
2. Solicitar la altura del rectángulo
3. Calcular el área del rectángulo.
4. Verificar si la base es igual a la altura del rectángulo
 - a. Si verifica, mostrar "Es un cuadrado"
 - b. Sino verifica, Verificar si la base es menor a la altura
 - i. Si verifica, mostrar "Es Vertical"
 - ii. Sino verifica, mostrar "Es Horizontal"
5. Mostrar "El Área del rectángulo es: " area

Ambiente:

Variable	Tipo	Descripción
base	Entero	base del rectángulo
altura	Entero	altura del rectángulo
area	Entero	área del rectángulo

Algoritmo:

Algoritmo ejercicio6

Definir base, altura, area Como Entero

Escribir "Ingrese base del rectangulo";

Leer base;

Escribir "Ingrese altura del rectangulo";

Leer altura;

area <- base * altura

Si (base == altura) Entonces

 Escribir "Es un Cuadrado"

SiNo

 Si (base < altura) Entonces

 Escribir "Es Vertical"

 SiNo

 Escribir "Es Horizontal"

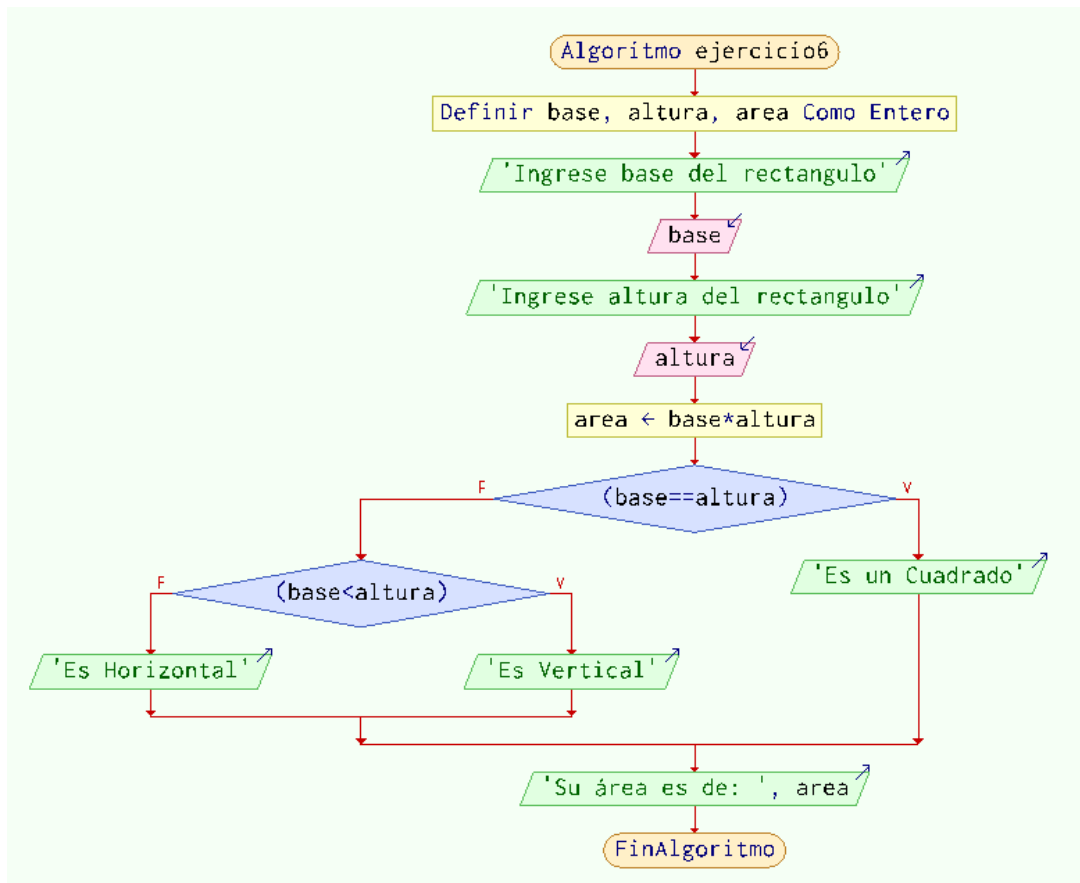
 Fin Si

Fin Si

Escribir "Su área es de: ", area

FinAlgoritmo

Diagrama de Flujo:



Seguimiento:

N°	base	altura	area	Salida
1				"Ingrese la base del rectángulo"
2	45			
3	45			"Ingrese la altura del rectángulo"
4	45	60		
5	45	60	2700	
6	45	60	2700	"Es Vertical"
7	45	60	2700	"El área es de: 2700"

Ejercicio 7) Mayor valor

Realice un algoritmo que pida 5 valores al usuario y luego informe cual es el mayor de los ingresados.

Restricción: la aplicación solo puede tener 2 variables.

Solución:

Análisis:

Entrada:

Cinco número a comparar

Salida:

El mayor de los números ingresados

Estrategia:

1. Solicitar el primer número
2. Asignar el número a Mayor
3. Solicitar el segundo número
4. Verificar si el número ingresado es mayor a Mayor
 - a. Si verifica, se asigna el número ingresado a Mayor
5. Solicitar el tercer número
6. Verificar si el número ingresado es mayor a Mayor
 - a. Si verifica, se asigna el número ingresado a Mayor
7. Solicitar el cuarto número
8. Verificar si el número ingresado es mayor a Mayor
 - a. Si verifica, se asigna el número ingresado a Mayor
9. Solicitar el quinto número
10. Verificar si el número ingresado es mayor a Mayor
 - a. Si verifica, se asigna el número ingresado a Mayor
11. Mostrar el Mayor

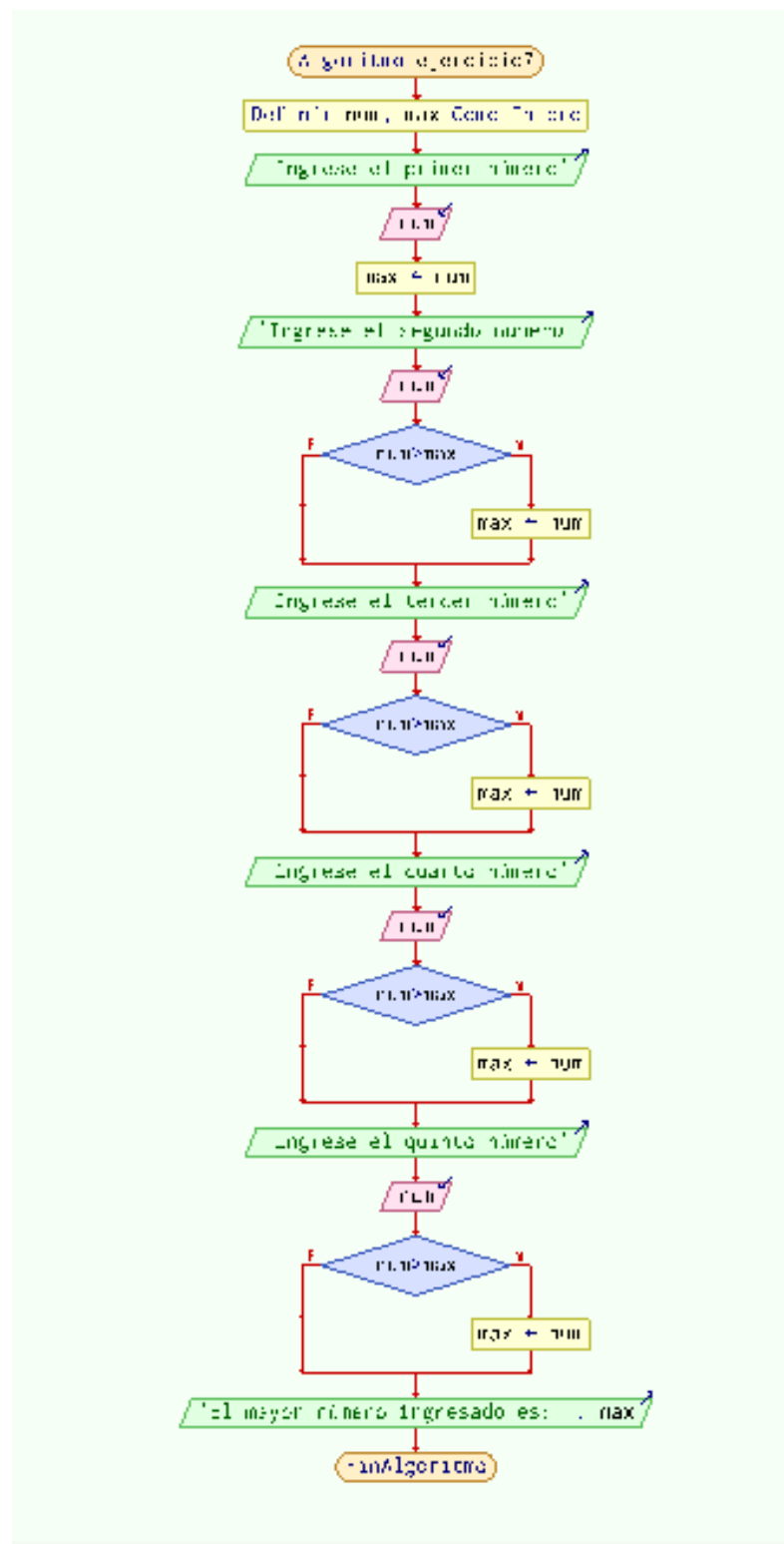
Ambiente:

Variable	Tipo	Descripción
mayor	Entero	Mayor de los números ingresados
num	Entero	Número ingresado

Algoritmo:

```
Algoritmo ejercicio7
  Definir num, max Como Entero
  Escribir "Ingrese el primer número"
  Leer num;
  max <- num;
  Escribir "Ingrese el segundo número"
  Leer num;
  Si num > max Entonces
    max <- num;
  Fin Si
  Escribir "Ingrese el tercer número"
  Leer num;
  Si num > max Entonces
    max <- num;
  Fin Si
  Escribir "Ingrese el cuarto número"
  Leer num;
  Si num > max Entonces
    max <- num;
  Fin Si
  Escribir "Ingrese el quinto número"
  Leer num;
  Si num > max Entonces
    max <- num;
  Fin Si
  Escribir "El mayor número ingresado es: ", max
FinAlgoritmo
```


Diagrama de Flujo:



Seguimiento:

N°	num	max	Salida
1			"Ingrese el primer número"
2	23		
3	23	23	
4	23	23	"Ingrese el segundo número"
5	46	23	
6	46	46	
7	46	46	"Ingrese el tercer número"
8	15	46	
9	15	46	
10	15	46	"Ingrese el cuarto número"
11	34	46	
12	34	46	
13	34	46	"Ingrese el quinto número"
14	45	46	
15	45	46	
16	45	46	"El mayor número ingresado es: 46"

Ejercicio 8) Orden que ocurrió el menor

Realice un algoritmo que pida 5 valores al usuario y luego informe cual es el número de orden en que se ingresó el menor de ellos.

Restricción: la aplicación solo puede tener 3 variables.

Solución:

Análisis:

Entrada:

- número ingresado

Salida:

- Orden de ingreso del menor número

Estrategia:

1. Solicitar el primer número
2. Asignar el número a Min, y 1 a Orden
3. Solicitar el segundo número
4. Verificar si el número ingresado es menor a Min
 - a. Si verifica, asignar el número ingresado a Min y 2 a Orden
5. Solicitar el tercer número
6. Verificar si el número ingresado es menor a Min
 - a. Si verifica, asignar el número ingresado a Min y 3 a Orden
7. Solicitar el cuarto número
8. Verificar si el número ingresado es menor a Min
 - a. Si verifica, asignar el número ingresado a Min y 4 a Orden
9. Solicitar el quinto número
10. Verificar si el número ingresado es menor a Min
 - a. Si verifica, asignar el número ingresado a Min y 5 a Orden
11. Mostra Orden.

Ambiente:

Variable	Tipo	Descripción
num	Entero	Número Ingresado
min	Entero	Mínimo ingresado
orden	Entero	Orden del Mínimo Ingresado

Algoritmo:

Algoritmo ejercicio8

Definir num, min, orden Como Entero

Escribir "Ingresar primer número ";

Leer num;

min <- num;

orden <- 1;

Escribir "Ingresar segundo número ";

Leer num;

Si num < min Entonces

 min <- num;

 orden <- 2;

Fin Si

Escribir "Ingresar tercer número ";

Leer num;

Si num < min Entonces

 min <- num;

 orden <- 3;

Fin Si

Escribir "Ingresar cuarto número ";

Leer num;

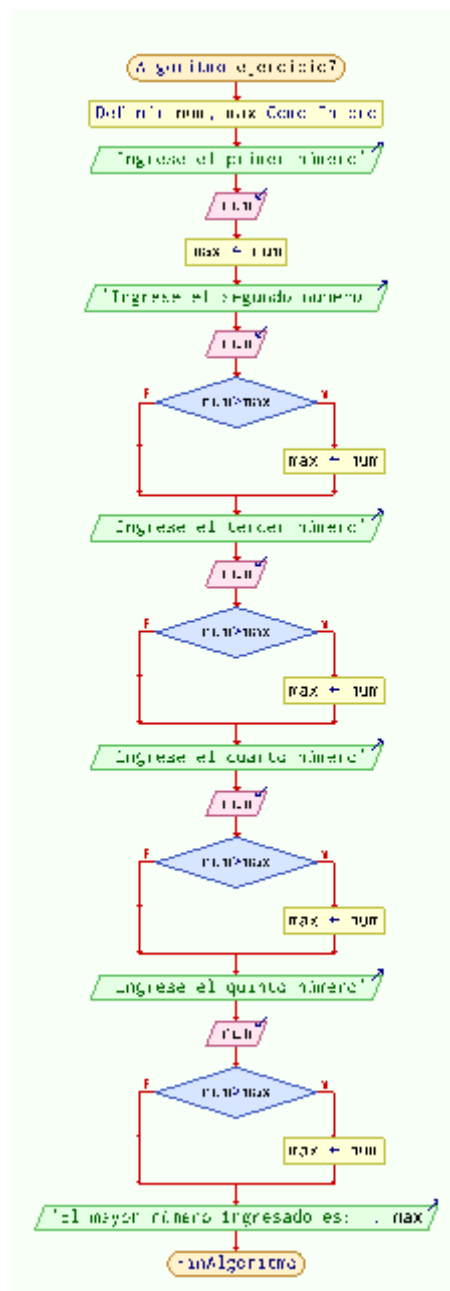
Si num < min Entonces

```

        min <- num;
        orden <- 4;
    Fin Si
    Escribir "Ingresar quinto número ";
    Leer num;
    Si num < min Entonces
        min <- num;
        orden <- 5;
    Fin Si
    Escribir "El Orden del menor número ingresado es: ", orden
FinAlgoritmo

```

Diagrama de Flujo:



Ejercicio 9) Tenis

Ingresado el nombre de los jugadores y el resultado de cada set (3) de un partido de tenis, informe en pantalla cuál es el ganador.

Ejemplo: Nadal, Del Potro: 7,5,4,6,6,2

Ganador Nadal

Solución:

Análisis:

Entrada:

- Nombre de los jugadores
- games ganados en cada set

Salida:

- Nombre del jugador ganador

Procesos:

$$\text{Set Jugador} = \begin{cases} \text{Jugador 1} & \text{Games Jugador 2} < \text{Games Jugador 1} \\ \text{Jugador 2} & \text{Games Jugador 1} < \text{Games Jugador 2} \end{cases}$$

$$\text{Resultado} = \begin{cases} \text{Jugador 1} & \text{Set Jugador 2} < \text{Set Jugador 1} \\ \text{Jugador 2} & \text{Set Jugador 1} < \text{Set Jugador 2} \end{cases}$$

Estrategia:

1. Solicitar nombre del Jugador 1.
2. Solicitar nombre del Jugador 2.
3. Solicitar los games del primer set ganados por el Jugador 1.
4. Solicitar los games del primer set ganados por el Jugador 2.
5. Verificar si los games ganados por el Jugador 1 son más que los ganados por el Jugador 2.
 - a. Si verifica, se da por ganado el set al Jugador 1 y se incrementa su total de sets en 1.
 - b. Si no verifica, se da por ganado el set al Jugador 2 y se incrementa su total de sets en 1.
6. Solicitar los games del segundo set ganados por el Jugador 1.
7. Solicitar los games del segundo set ganados por el Jugador 2.
8. Verificar si los games ganados por el Jugador 1 son más que los ganados por el Jugador 2.
 - a. Si verifica, se da por ganado el set al Jugador 1 y se incrementa su total de sets en 1.
 - b. Si no verifica, se da por ganado el set al Jugador 2 y se incrementa su total de sets en 1.
9. Solicitar los games del tercer set ganados por el Jugador 1.
10. Solicitar los games del tercer set ganados por el Jugador 2.
11. Verificar si los games ganados por el Jugador 1 son más que los ganados por el Jugador 2.

- a. Si verifica, se da por ganado el set al Jugador 1 y se incrementa su total de sets en 1.
 - b. Si no verifica, se da por ganado el set al Jugador 2 y se incrementa su total de sets en 1.
12. Verificar si los sets ganados por el Jugador 1 son más que los ganados por el Jugador 2.
- a. Si verifica, se da por ganador al Jugador 1 y se muestra su nombre por pantalla.
 - b. Si no verifica, se da por ganador al Jugador 2 y se muestra su nombre por pantalla.

Ambiente:

Variable	Tipo	Descripción
nombre1	Cadena	Nombre del Jugador 1
nombre2	Cadena	Nombre del Jugador 2
total1	Entero	Cantidad de sets ganados por Jugador 1
total2	Entero	Cantidad de sets ganados por Jugador 2
puntos1	Entero	Cantidad de games ganados por Jugador 1
puntos2	Entero	Cantidad de games ganados por Jugador 2

Algoritmo:

Algoritmo ejercicio9

Definir total1, total2, puntos1, puntos2 Como Entero
 Definir nombre1, nombre2 Como Caracter

Escribir "Ingrese el nombre de Jugador 1";
 Leer nombre1
 Escribir "Ingrese el nombre de Jugador 2";
 Leer nombre2
 total1 <- 0;
 total2 <- 0;

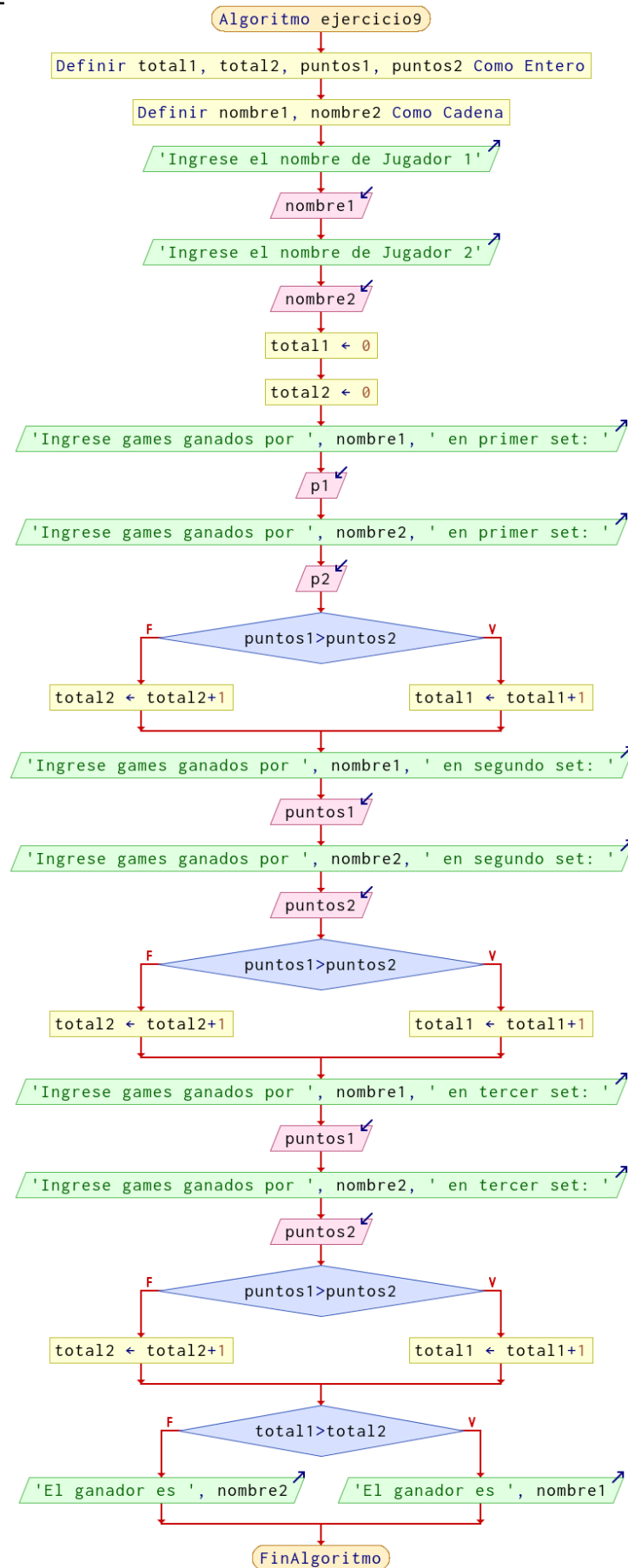
Escribir "Ingrese games ganados por ", nombre1, " en primer set: ";
 Leer p1
 Escribir "Ingrese games ganados por ", nombre2, " en primer set: ";
 Leer p2
 Si puntos1 > puntos2 Entonces
 total1 = total1 + 1;
 SiNo
 total2 = total2 + 1;
 Fin Si
 Escribir "Ingrese games ganados por ", nombre1, " en segundo set: ";
 Leer puntos1
 Escribir "Ingrese games ganados por ", nombre2, " en segundo set: ";
 Leer puntos2

```
Si puntos1 > puntos2 Entonces
    total1 = total1 + 1;
SiNo
    total2 = total2 + 1;
Fin Si
Escribir "Ingrese games ganados por ", nombre1," en tercer set: ";
Leer puntos1
Escribir "Ingrese games ganados por ", nombre2," en tercer set: ";
Leer puntos2
Si puntos1 > puntos2 Entonces
    total1 = total1 + 1;
SiNo
    total2 = total2 + 1;
Fin Si

Si total1 > total2 Entonces
    Escribir "El ganador es ",nombre1;
SiNo
    Escribir "El ganador es ",nombre2;
Fin Si
```

FinAlgoritmo

Diagrama de Flujo:



Seguimiento:

n°	nombre1	nombre2	puntos1	puntos2	total1	total2	Salida
1	Nadal						"Ingrese el nombre del jugador 1"
2	Nadal	Federer					"Ingrese el nombre del jugador 2"
3	Nadal	Federer			0	0	"Ingrese los games ganados por Nadal en primer set"
4	Nadal	Federer	4		0	0	
5	Nadal	Federer	4				"Ingrese los games ganados por Federer en primer set"
6	Nadal	Federer	4	6	0	1	
7	Nadal	Federer	4	6	0	1	"Ingrese los games ganados por Nadal en segundo set"
8	Nadal	Federer	6	6	0	1	
9	Nadal	Federer	6	6	0	1	"Ingrese los games ganados por Federer en segundo set"
10	Nadal	Federer	6	3	0	1	
11	Nadal	Federer	6	3	1	1	"Ingrese los games ganados por Nadal en tercer set"
12	Nadal	Federer	7	3	1	1	
13	Nadal	Federer	7	3	1	1	"Ingrese los games ganados por Federer en tercer set"
14	Nadal	Federer	7	6	1	1	
15	Nadal	Federer	7	6	2	1	
16	Nadal	Federer	7	6	2	1	"El ganador es Nadal"

Ejercicio 10)

Implemente un algoritmo que permita determinar si un año es bisiesto o no. Un año es bisiesto si es múltiplo de 4 (por ejemplo 1984). Los años múltiplos de 100 no son bisiestos, salvo si ellos son también múltiplos de 400 (2000 es bisiesto, pero; 1800 no lo es).

Solución:

Análisis:

Entrada:

- Nombre de los jugadores
- games ganados en cada set

Salida:

- {"El año es bisiesto","El año no es bisiesto" }

Procesos:

$$\text{Año Bisiesto} = \begin{cases} \text{Año múltiplo de 4 y no múltiplos de 100} \\ \text{Año múltiplo de 4 y múltiplos de 100 y de 400} \end{cases}$$

Estrategia:

1. Solicitar el año a analizar.
2. Verificar si el año es múltiplo de 4.
 - a. Si verifica, verificar que es múltiplo de 100.
 - i. Si verifica, verificar que es múltiplo de 400.
 1. Si verifica, mostrar el mensaje es "Año Bisiesto"
 2. Si no verifica, mostrar el mensaje "No es Año Bisiesto".
 - ii. Si no verifica, mostrar el mensaje "No es Año Bisiesto".
 - b. Si no verifica, mostrar el mensaje "Año Bisiesto".
3. Si no verifica, mostrar el mensaje "No es Año Bisiesto".

Ambiente:

Variable	Tipo	Descripción
anio	Entero	Año a analizar

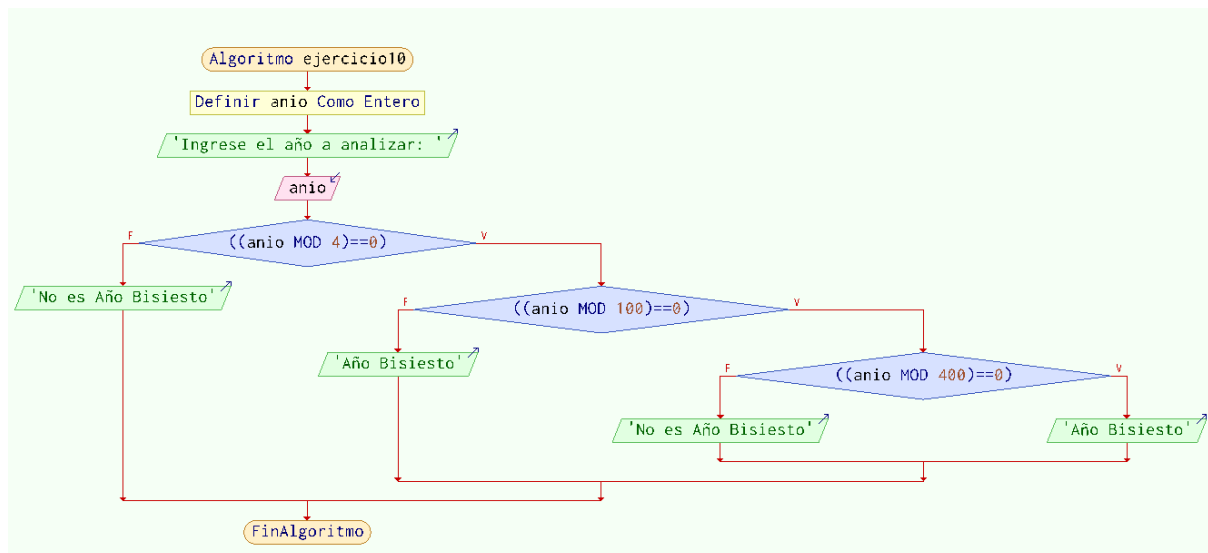
Algoritmo:

```
Algoritmo ejercicio10
  Definir anio Como Entero

  Escribir "Ingrese el año a analizar: "
  Leer anio

  Si ((anio mod 4) == 0) Entonces
    Si ((anio mod 100) == 0) Entonces
      Si ((anio mod 400) == 0) Entonces
        Escribir "Año Bisiesto"
      SiNo
        Escribir "No es Año Bisiesto"
    Fin Si
  SiNo
    Escribir "Año Bisiesto"
  Fin Si
SiNo
  Escribir "No es Año Bisiesto"
Fin Si
FinAlgoritmo
```

Diagrama de Flujo:



Seguimiento:

Nº	anio	Salida
1		"Ingrese el año a analizar: "
2	1976	
3	1976	"Año Bisiesto"

Ejercicio 11)

Realice un algoritmo que permita ingresar el número del mes y determine cuántos días tiene. Para el caso de Febrero, el algoritmo deberá indicar que no cuenta con la información necesaria para dar la respuesta.

Solución:

Análisis:

Entrada:

- Mes a analizar

Salida:

- Número de días del mes

Procesos:

$$\text{Nº de días} = \begin{cases} \text{No Válidos} & \text{Nº de mes igual a 2} \\ 31 \text{ días} & \text{Nº de mes impar y menor o igual a 7,} \\ & \text{o Nº de mes Par y mayor a 7} \\ 30 \text{ días} & \text{Nº de mes par y menor a 7,} \\ & \text{o Nº de mes Impar y mayor a 7} \end{cases}$$

Estrategia:

1. Solicitar el mes a analizar.
2. Verificar que el número de mes sea distinto de 2, mayor a 0 y menor a 13..
 - a. Si verifica, verificamos que mes sea impar.
 - i. Si verifica, verificamos que sea menor o igual a 7.
 1. Si verifica, mostramos "El mes tiene 31 días"
 2. Sino verifica, mostramos "El mes tiene 30 días"
 - ii. Sino verifica, verificamos si mayor a 7.
 1. Si verifica, mostramos "El mes tiene 31 días"
 2. Sino verifica, mostramos "El mes tiene 30 días"
 - b. Sino verifica, mostramos "Mes inválido"

Ambiente:

Variable	Tipo	Descripción
mes	Entero	mes a analizar

Algoritmo:

Algoritmo ejercicio11

Definir mes Como Entero

Escribir "Ingrese el número de mes a analizar";

Leer mes;

Si (mes <> 2)&&(mes > 0)&&(mes < 13) Entonces

 Si ((mes mod 2) == 1) Entonces

 Si (mes <= 7) Entonces

 Escribir "El mes tiene 31 días"

 SiNo

 Escribir "El mes tiene 30 días"

 Fin Si

SiNo

 Si (mes > 7) Entonces

 Escribir "El mes tiene 31 días"

 SiNo

 Escribir "El mes tiene 30 días"

 Fin Si

Fin Si

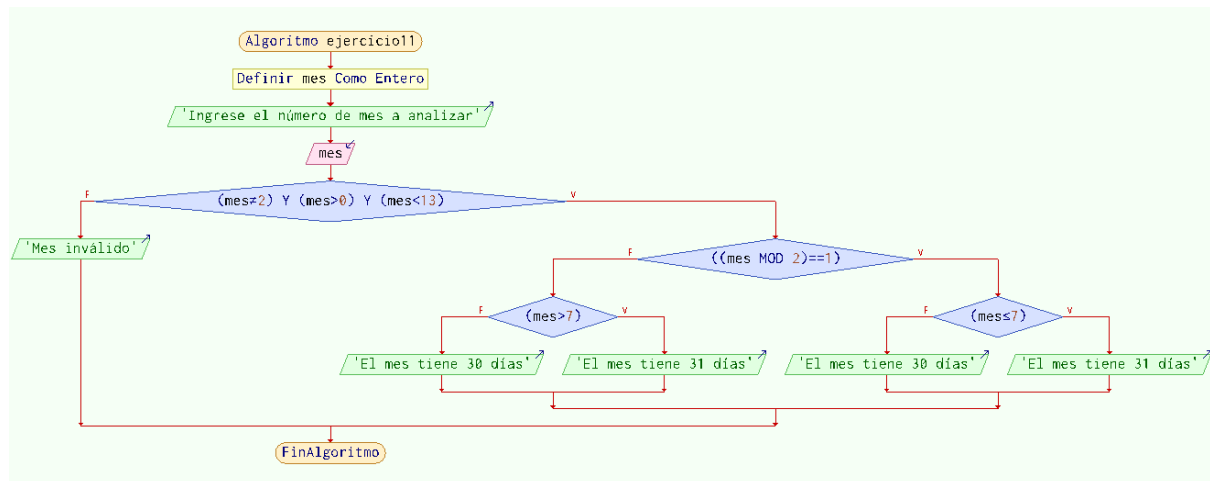
SiNo

 Escribir "Mes inválido"

Fin Si

FinAlgoritmo

Diagrama de flujo:



Sequimiento:

Nº	mes	Salida
1		"Ingrese el número de mes a analizar: "
2	9	
3	9	"El mes tiene 30 días"

