

Nombre y apellidos: Diana Rammal Sansón

Todos los ejercicios deberán ser resueltos mediante el uso de diagramas de flujo y luego mediante uso de pseudocódigo.

Para ver mejor los diagramas, adjunto enlace a la pizarra de Freeform donde dibujé los 20 ejercicios:

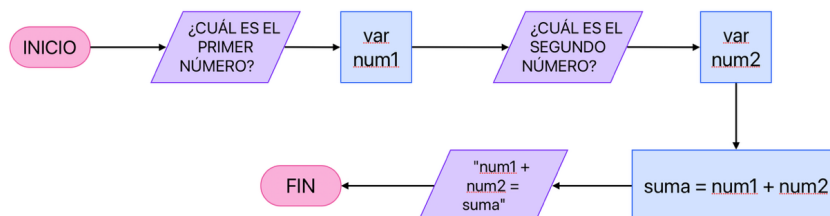
<https://www.icloud.com/freeform-copy/49BB14E3-A655-4179-880C-4DC5E9570BA8>

Entradas/Salidas y operaciones básicas

Ejercicio 1:

Crea un algoritmo que solicite dos números al usuario y muestre su suma.

Diagrama de flujo:



Pseudocódigo:

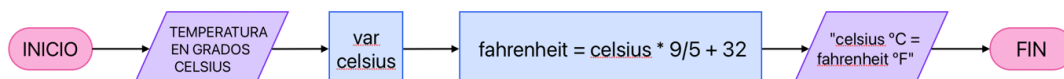
1. INICIO
2. PREGUNTAR “¿Cuál es el primer número?”
3. Variable num1 = Respuesta Paso 2
4. PREGUNTAR “¿Cuál es el segundo número?”
5. Variable num2 = Respuesta Paso 4
6. CÁLCULO suma = num1 + num2
7. MOSTRAR “num1 + num2 = suma”
8. FIN

Ejercicio 2:

Diseñar un algoritmo que convierta grados Celsius a Fahrenheit.

Nota: La fórmula para la conversión es $F = C \times \frac{9}{5} + 32$.

Diagrama de flujo:



Pseudocódigo:

1. INICIO
2. PEDIR Temperatura en grados Celsius
3. Variable celsius = Respuesta Paso 2
4. CÁLCULO fahrenheit = celsius * 9/5 + 32

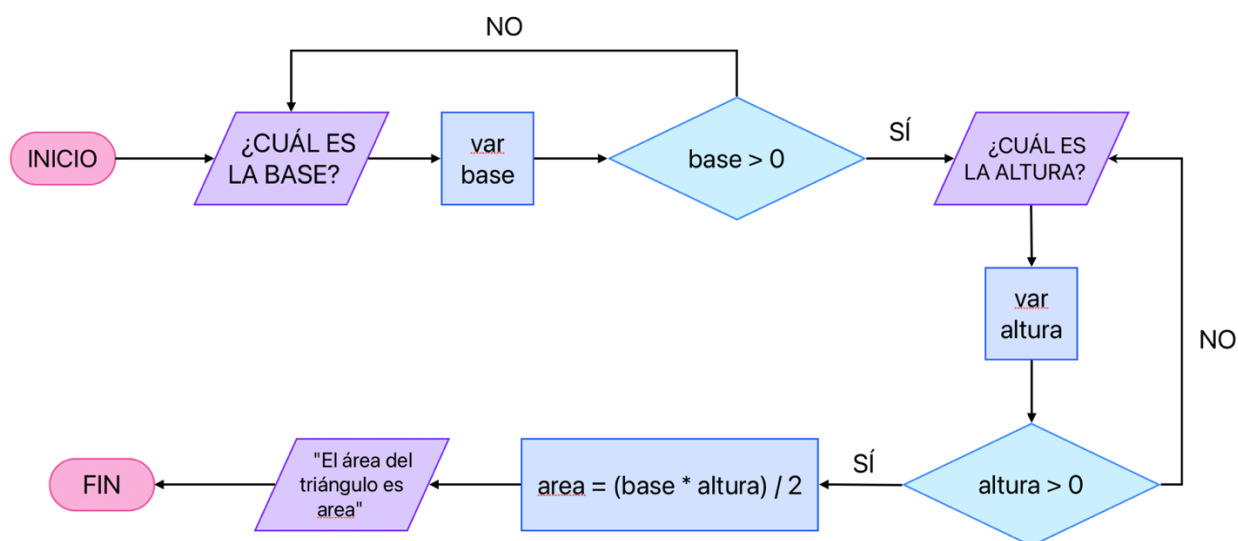
5. MOSTRAR "celsius °C = fahrenheit °F"
6. FIN

Ejercicio 3:

Crea un algoritmo que calcule el área de un triángulo dadas su base y altura.

Nota: El área de un triángulo es $\text{Área} = \frac{\text{base} \times \text{altura}}{2}$.

Diagrama de flujo:



Pseudocódigo:

```

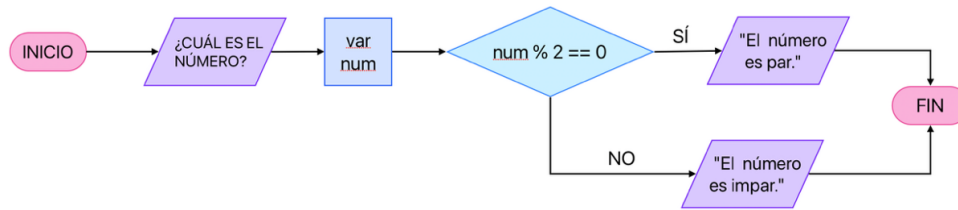
1. INICIO
2. PREGUNTAR "¿Cuál es la base?"
3. Variable base = Respuesta Paso 2
4. Si (base > 0)
    o Si es SÍ voy a 5
    o Si es NO voy a 2
5. PREGUNTAR "¿Cuál es la altura?"
6. Variable altura = Respuesta Paso 5
7. Si (altura > 0)
    o Si es SÍ voy a 8
    o Si es NO voy a 5
8. CÁLCULO área = (base * altura) / 2
9. MOSTRAR "El área del triángulo es area"
10. FIN
  
```

Condiciones simples

Ejercicio 4:

Diseña un algoritmo que determine si un número es par o impar.

Diagrama de flujo:



Pseudocódigo:

```

1. INICIO
2. PREGUNTAR "¿Cuál es el número?"
3. Variable num = Respuesta Paso 2
4. Si (num % 2 == 0) entonces
    o MOSTRAR "El número es par."
    y si no
    o MOSTRAR "El número es impar."
5. FIN
  
```

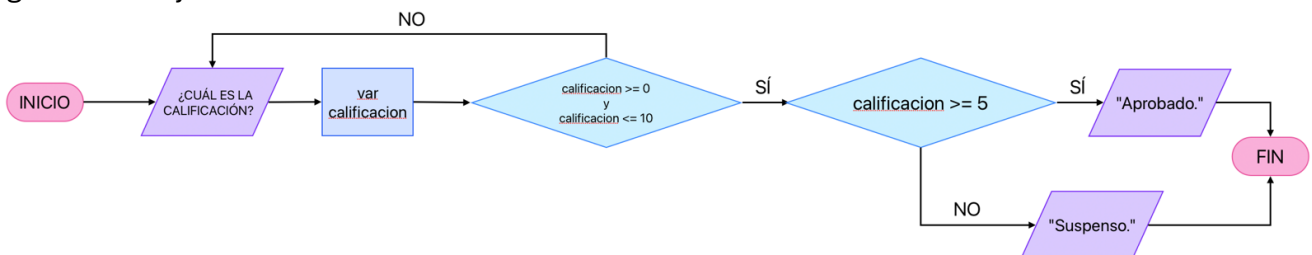
Ejercicio 5:

Crea un algoritmo que clasifique una calificación escolar en "Aprobado" o "Suspenso". El aprobado es a partir de 5. Crea una versión que permita especificar la nota de corte para aprobar o no.

PRIMERA VERSIÓN

Aclaración: he considerado que la calificación se debe encontrar entre 0 y 10.

Diagrama de flujo:



Pseudocódigo:

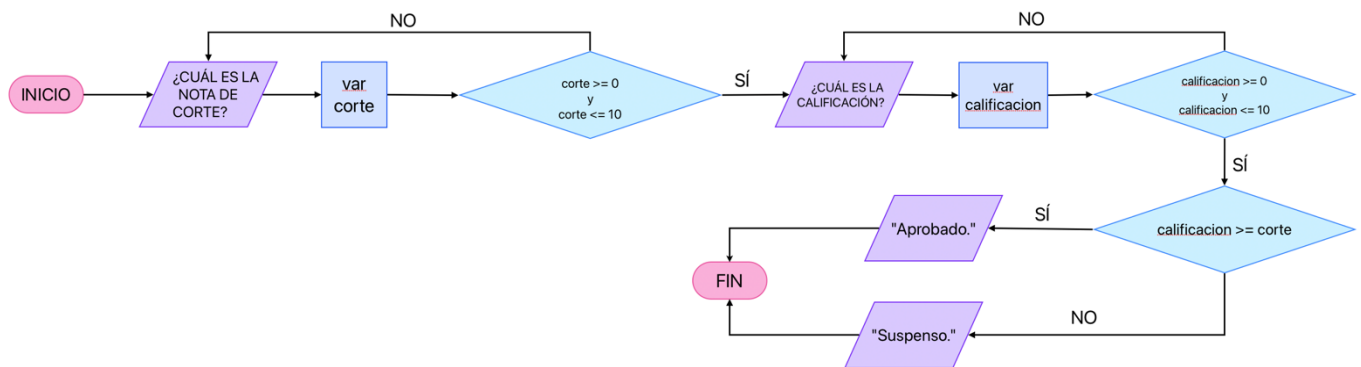
```

1. INICIO
2. PREGUNTAR "¿Cuál es la calificación?"
3. Variable calificacion = Respuesta Paso 2
4. Si (calificacion >= 0 Y calificacion <= 10)
    o Si es SÍ voy a 5
    o Si es NO voy a 2
5. Si (calificacion >= 5) entonces
    o MOSTRAR "Aprobado."
    y si no
    o MOSTRAR "Suspenso."
6. FIN
  
```

SEGUNDA VERSIÓN

Aclaración: he considerado que la nota de corte y la calificación se deben encontrar entre 0 y 10.

Diagrama de flujo:



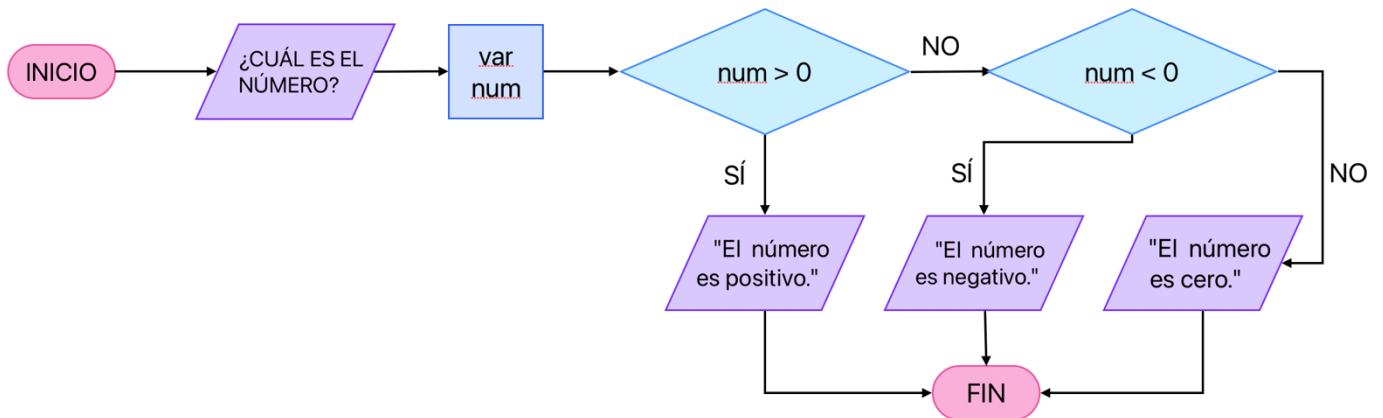
Pseudocódigo:

1. INICIO
2. PREGUNTAR “¿Cuál es la nota de corte?”
3. Variable corte = Respuesta Paso 2
4. Si (corte >= 0 Y corte <= 10)
 - Si es SÍ voy a 5
 - Si es NO voy a 2
5. PREGUNTAR “¿Cuál es la calificación?”
6. Variable calificacion = Respuesta Paso 5
7. Si (calificacion >= 0 Y calificacion <= 10)
 - Si es SÍ voy a 8
 - Si es NO voy a 5
8. Si (calificacion >= corte) entonces
 - MOSTRAR “Aprobado.”
 y si no
 - MOSTRAR “Suspenso.”
9. FIN

Ejercicio 6:

Diseña un algoritmo que, dado un número, indique si es positivo, negativo o cero.

Diagrama de flujo:



Pseudocódigo:

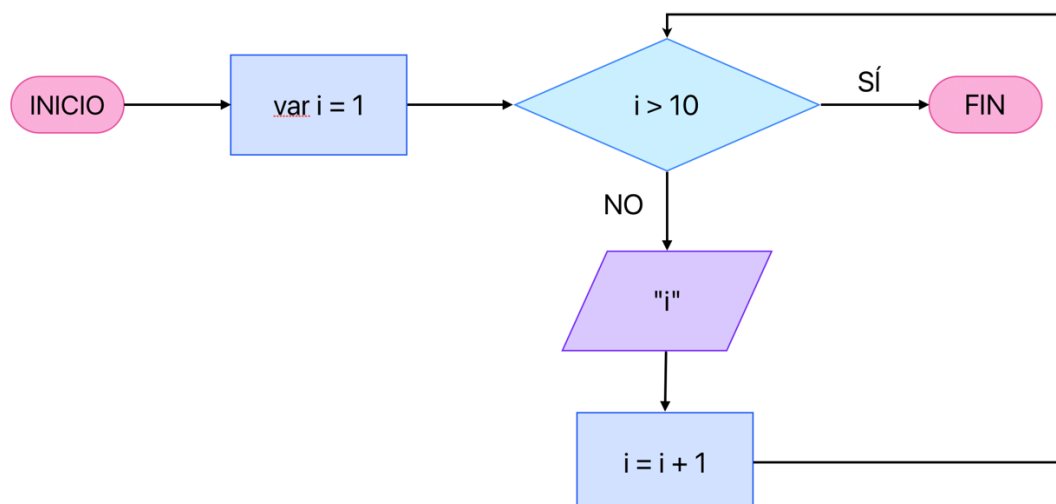
1. INICIO
2. PREGUNTAR "¿Cuál es el número?"
3. Variable num = Respuesta Paso 2
4. Si (`num > 0`) entonces
 - MOSTRAR "El número es positivo."
 y si no
 - Si (`num < 0`) entonces
 - MOSTRAR "El número es negativo."
 - Si no
 - MOSTRAR "El número es cero."
5. FIN

Bucles simples

Ejercicio 7:

Crea un algoritmo que imprima los primeros 10 números naturales.

Diagrama de flujo:



Pseudocódigo:

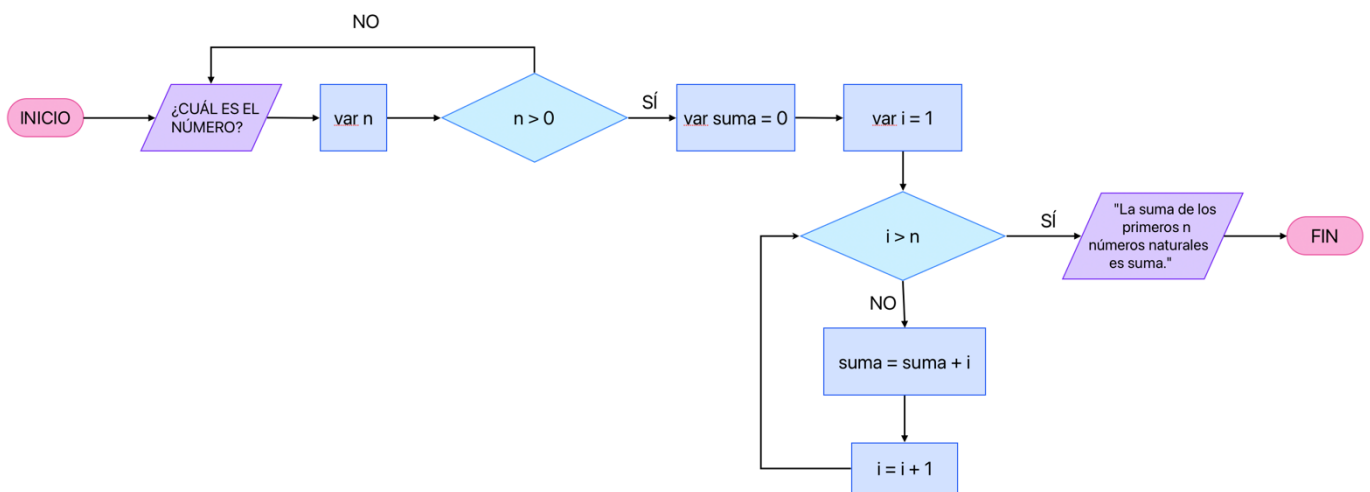
```
1. INICIO
2. Variable i = 1
3. Si (i > 10)
    o Si es SÍ voy a 7
    o Si es NO voy a 4
4. MOSTRAR "i"
5. CÁLCULO i = i + 1
6. Volver a 3
7. FIN
```

Ejercicio 8:

Diseña un algoritmo que calcule la suma de los primeros N números naturales, donde N es un valor especificado por el usuario.

Aclaración: se utilizará un bucle, aunque utilizando la fórmula $N \times (N + 1) / 2$ sea más eficiente.

Diagrama de flujo:



Pseudocódigo:

```
1. INICIO
2. PREGUNTAR "¿Cuál es el número?"
3. Variable n = Respuesta Paso 2
4. Si (n > 0)
    o Si es SÍ voy a 5
    o Si es NO voy a 2
5. Variable suma = 0
6. Variable i = 1
7. Si (i > n)
    o Si es SÍ voy a 11
    o Si es NO voy a 8
8. CÁLCULO suma = suma + i
```

```

9. CÁLCULO  $i = i + 1$ 
10. Volver a 7
11. MOSTRAR "La suma de los primeros n números naturales es suma."
12. FIN

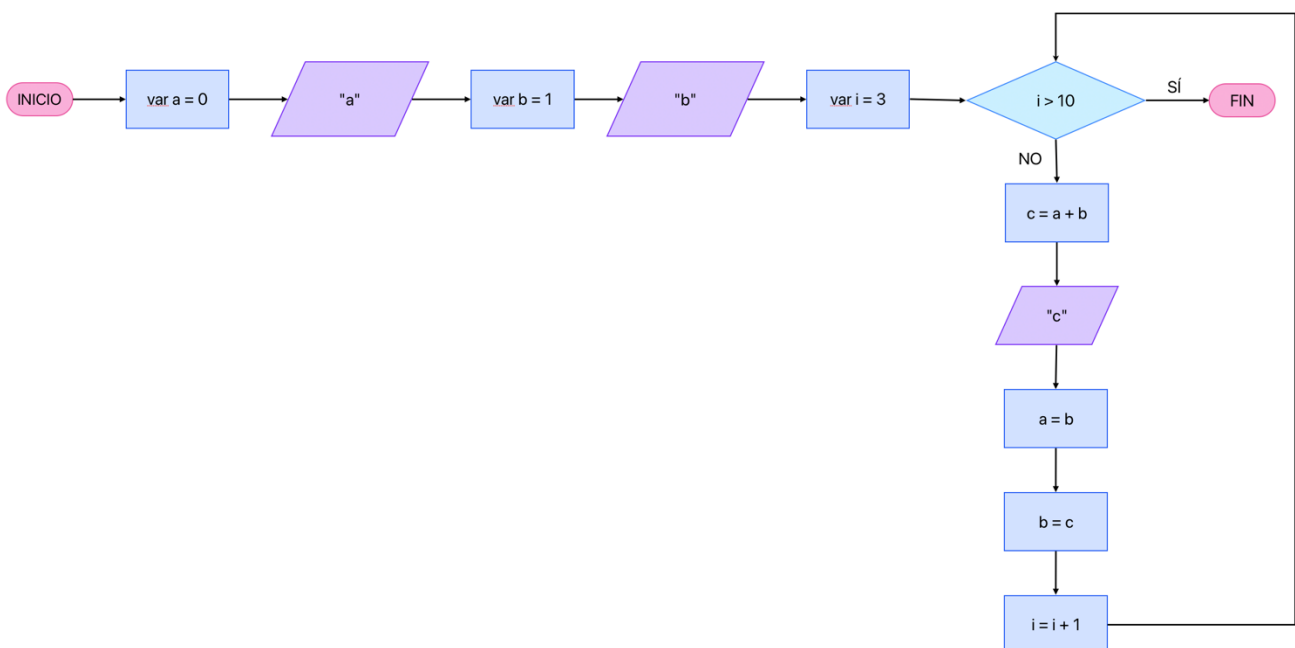
```

Ejercicio 9:

Crea un algoritmo que imprima la serie de Fibonacci hasta el décimo término de la misma.

Nota: La serie de Fibonacci es una secuencia donde cada número es la suma de los dos anteriores: 0, 1, 1, 2, 3, 5, 8...

Diagrama de flujo:



Pseudocódigo:

```

1. INICIO
2. Variable a = 0
3. MOSTRAR "a"
4. Variable b = 1
5. MOSTRAR "b"
6. Variable i = 3
7. Si (i > 10)
    o Si es SÍ voy a 13
    o Si es NO voy a 8
8. CÁLCULO c = a + b
9. MOSTRAR "c"
10. CÁLCULO a = b
11. CÁLCULO b = c
12. CÁLCULO i = i + 1
13. Volver a 7
14. FIN

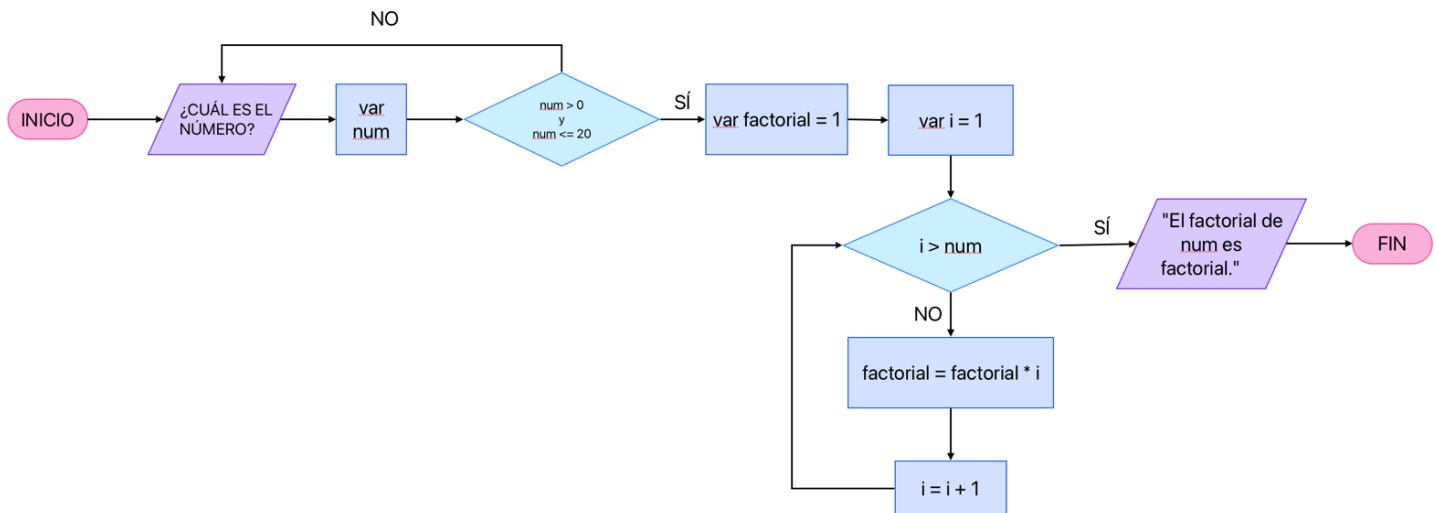
```

Ejercicio 10:

Diseña un algoritmo que calcule el factorial de un número que el usuario introduzca y que no sea mayor de 20.

Nota: El factorial de un número n es $n! = n \times (n - 1) \times (n - 2) \times \dots \times 1$

Diagrama de flujo:



Pseudocódigo:

```

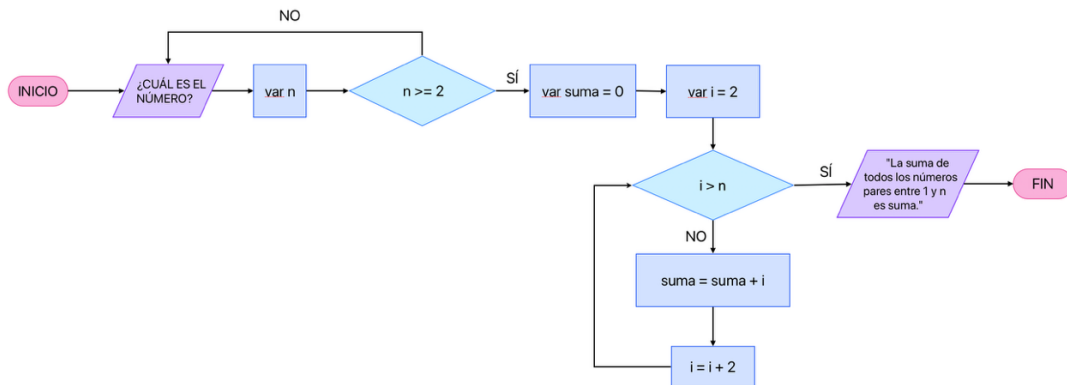
1. INICIO
2. PREGUNTAR "¿Cuál es el número?"
3. Variable num = Respuesta Paso 2
4. Si (num > 0 Y num <= 20)
    o Si es SÍ voy a 5
    o Si es NO voy a 2
5. Variable factorial = 1
6. Variable i = 1
7. Si (i > num)
    o Si es SÍ voy a 11
    o Si es NO voy a 8
8. CÁLCULO factorial = factorial * i
9. CÁLCULO i = i + 1
10. Volver a 7
11. MOSTRAR "El factorial de num es factorial."
12. FIN
  
```

Ejercicio 11:

Crea un algoritmo que sume todos los números pares entre 1 y N.

Nota: El número N siempre será un número solicitado al usuario.

Diagrama de flujo:



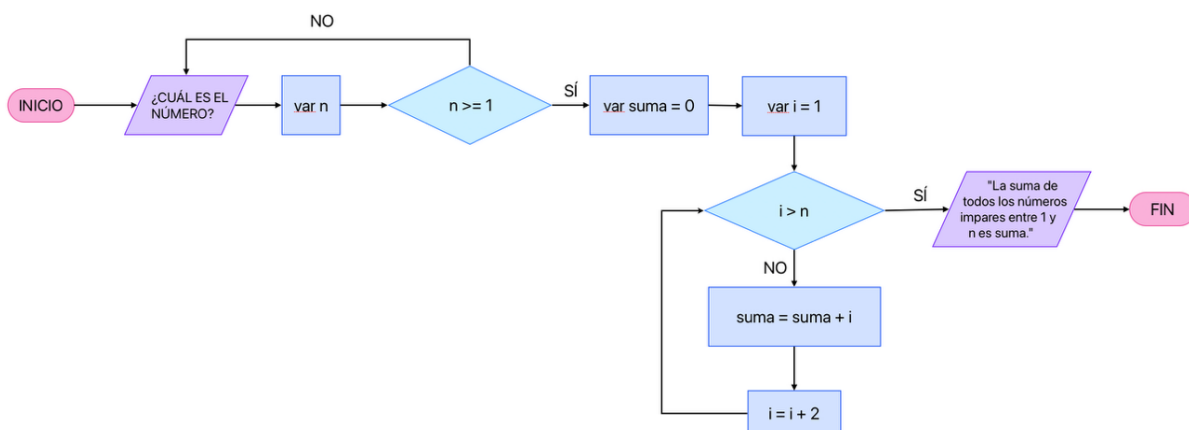
Pseudocódigo:

1. INICIO
2. PREGUNTAR “¿Cuál es el número?”
3. Variable n = Respuesta Paso 2
4. Si (n >= 2)
 - Si es SÍ voy a 5
 - Si es NO voy a 2
5. Variable suma = 0
6. Variable i = 2
7. Si (i > n)
 - Si es SÍ voy a 11
 - Si es NO voy a 8
8. CÁLCULO suma = suma + i
9. CÁLCULO i = i + 2
10. Volver a 7
11. MOSTRAR “La suma de todos los números pares entre 1 y n es suma.”
12. FIN

Ejercicio 12:

Crea un algoritmo que calcule la suma de todos los números impares entre 1 y N.

Diagrama de flujo:



Pseudocódigo:

```

1. INICIO
2. PREGUNTAR "¿Cuál es el número?"
3. Variable n = Respuesta Paso 2
4. Si (n >= 1)
    o Si es SÍ voy a 5
    o Si es NO voy a 2
5. Variable suma = 0
6. Variable i = 1
7. Si (i > n)
    o Si es SÍ voy a 11
    o Si es NO voy a 8
8. CÁLCULO suma = suma + i
9. CÁLCULO i = i + 2
10. Volver a 7
11. MOSTRAR "La suma de todos los números impares entre 1 y n es suma."
12. FIN

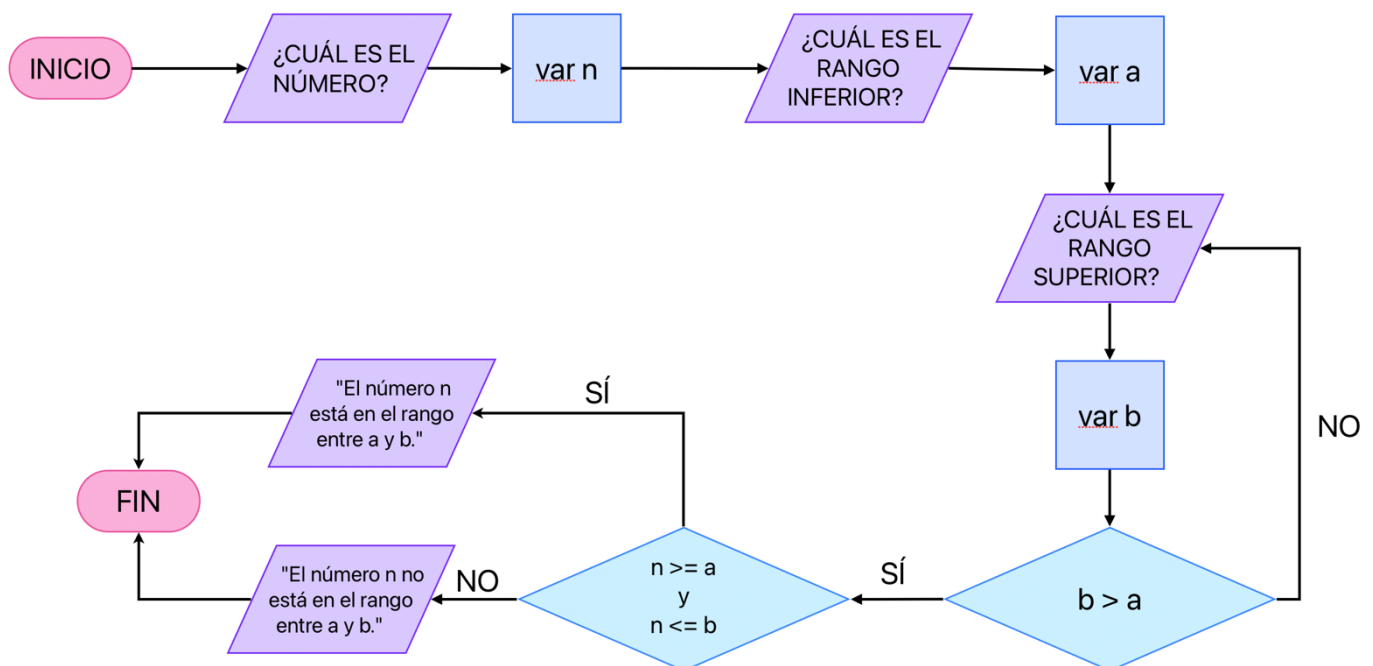
```

Ejercicio 13:

Diseña un algoritmo capaz de determinar si un número está en el rango entre A y B (inclusive).

Nota: Tanto el número N como A y B son indicados por el usuario.

Diagrama de flujo:



Pseudocódigo:

1. INICIO
2. PREGUNTAR "¿Cuál es el número?"
3. Variable n = Respuesta Paso 2
4. PREGUNTAR "¿Cuál es el rango inferior?"
5. Variable a = Respuesta Paso 4
6. PREGUNTAR "¿Cuál es el rango superior?"
7. Variable b = Respuesta Paso 6
8. Si (b > a)
 - o Si es SÍ voy a 9
 - o Si es NO voy a 6
9. Si (n >= a Y n <= b) entonces
 - o MOSTRAR "El número n está en el rango entre a y b."
 - y si no
 - o MOSTRAR "El número n no está en el rango entre a y b."
10. FIN

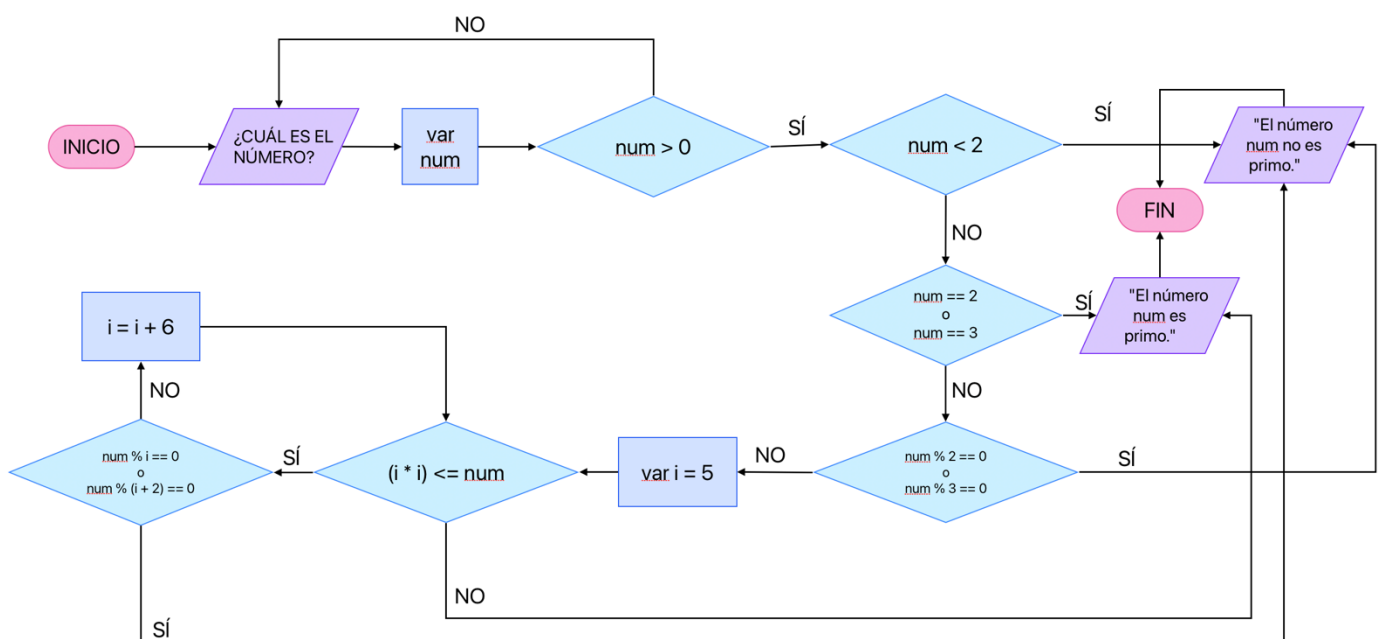
Ejercicio 14:

Crea un algoritmo que determine si un número es primo o no.

Nota: Un número primo solo tiene dos divisores: 1 y él mismo.

Aclaración: he utilizado el condicional $\text{num} < 2$ para ver el caso en que el número sea 1 (el cual no es primo), mientras que con $\text{num} == 2$ o $\text{num} == 3$ vemos los casos en los que el número sea 2 o 3 (los cuales son primos). Si usamos solamente la condición dada en el ejemplo de clase $\text{num} \leq 3$ es primo, entonces estamos indicando erróneamente que 1 es primo.

Diagrama de flujo:



Pseudocódigo:

```

1. INICIO
2. PREGUNTAR "¿Cuál es el número?"
3. Variable num = Respuesta Paso 2
4. Si (num > 0)
    o Si es SÍ voy a 5
    o Si es NO voy a 2
5. Si (num < 2) entonces
    o MOSTRAR "El número num no es primo."
    o Ir a paso 10
6. Si (num == 2 O num == 3) entonces
    o MOSTRAR "El número num es primo."
    o Ir a paso 10
7. Si (num % 2 == 0 O num % 3 == 0) entonces
    o MOSTRAR "El número num no es primo."
    o Ir a paso 10
8. Variable i = 5
9. Si ((i * i) <= num) entonces
    o Si (num % i == 0 O (num % (i + 2) == 0)) entonces
        ▪ MOSTRAR "El número num no es primo."
        ▪ Ir a paso 10
    o y si no
        ▪ CÁLCULO i = i + 6
        ▪ Volver a paso 9
    y si no
    o MOSTRAR "El número num es primo."
10. FIN

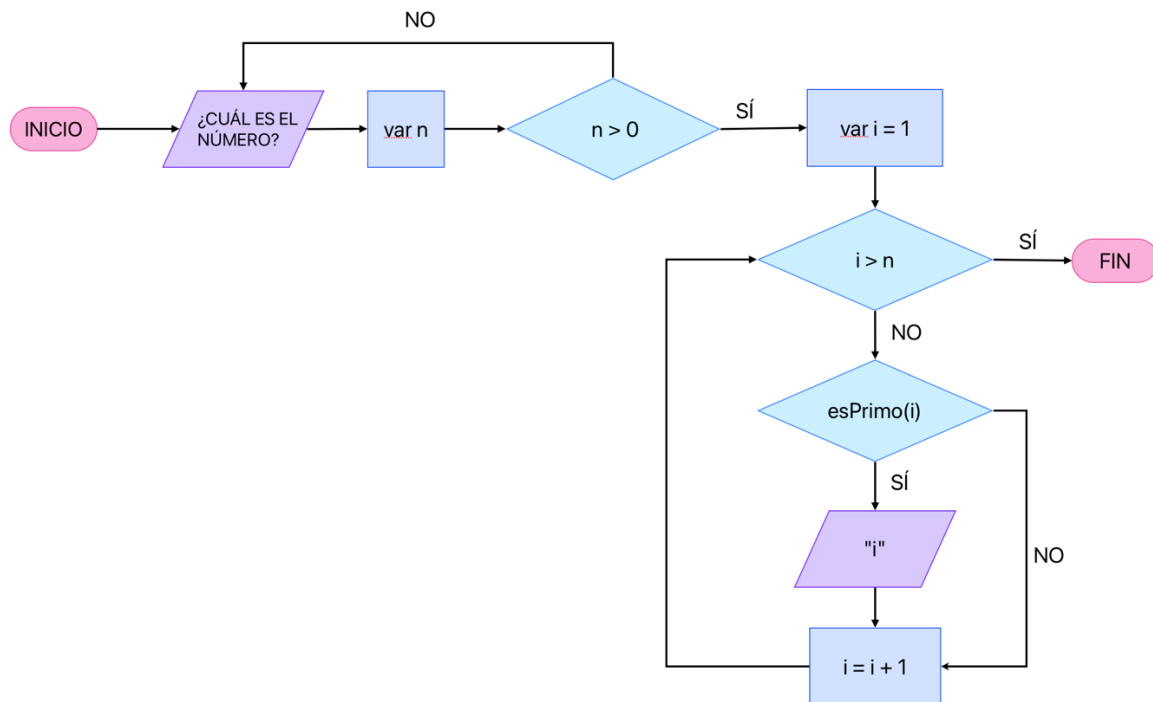
```

Ejercicio 15:

Crea un algoritmo que imprima todos los números primos menores que N.

Aclaración: para realizar este ejercicio supondremos que disponemos ya de una función denominada *esPrimo(n: Int)* que devuelve un booleano. El funcionamiento sería el mismo que el del ejercicio 14, pero en vez de imprimir un String devolveremos un Boolean el cual será true si el número es primo o será false si el número no es primo.

Diagrama de flujo:



Pseudocódigo:

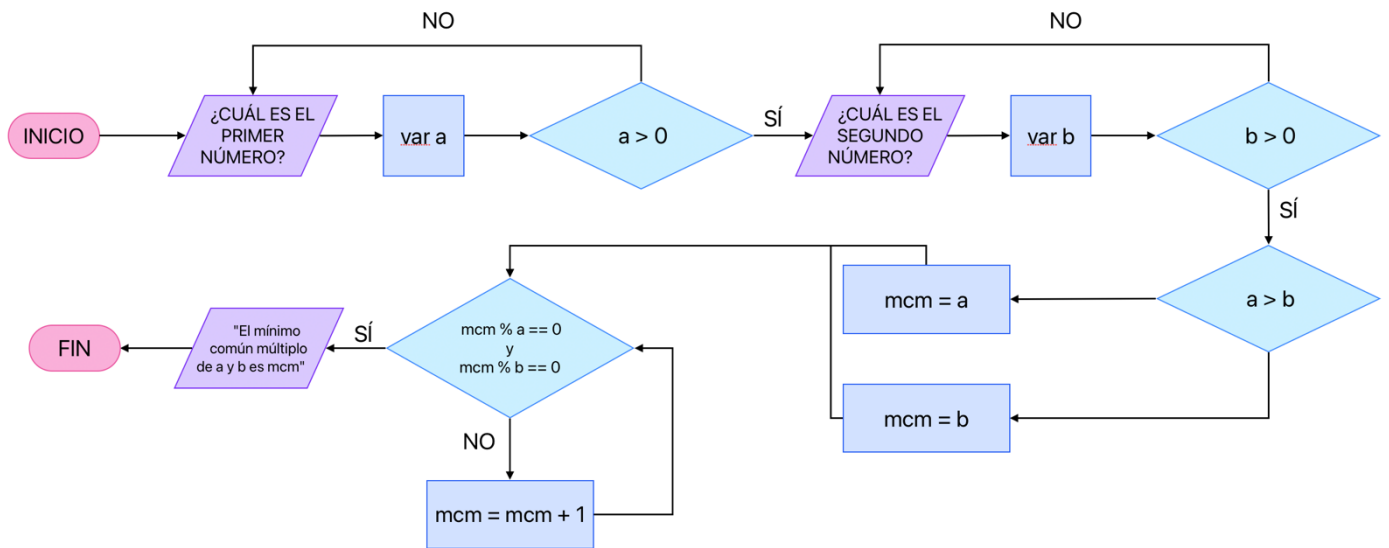
1. INICIO
2. PREGUNTAR “¿Cuál es el número?”
3. Variable n = Respuesta Paso 2
4. Si (n > 0)
 - Si es SÍ voy a 5
 - Si es NO voy a 2
5. Variable i = 1
6. Si (i > n)
 - Si es SÍ voy a 8
 - Si es NO voy a 7
7. Si (esPrimo(i)) entonces
 - MOSTRAR “i”
 - CÁLCULO i = i + 1
 - Volver a paso 6
 y si no
 - CÁLCULO i = i + 1
 - Volver a paso 6
8. FIN

Ejercicio 16:

Diseña un algoritmo que encuentre el mínimo común múltiplo de dos números A y B dados por el usuario.

Nota: El mínimo común múltiplo (MCM) es el menor número que es múltiplo de ambos números.

Diagrama de flujo:



Pseudocódigo:

```

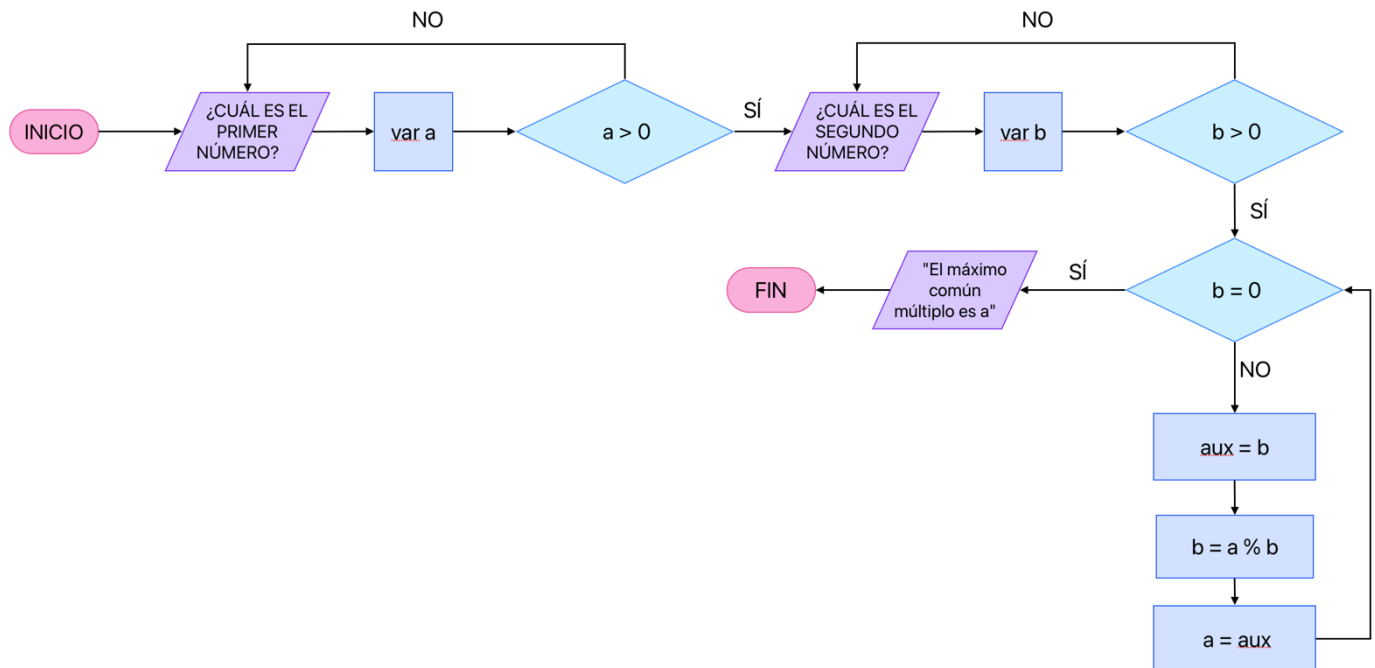
1. INICIO
2. PREGUNTAR "¿Cuál es el primer número?"
3. Variable a = Respuesta Paso 2
4. Si (a > 0)
    o Si es SÍ voy a 5
    o Si es NO voy a 2
5. PREGUNTAR "¿Cuál es el segundo número?"
6. Variable b = Respuesta Paso 5
7. Si (b > 0)
    o Si es SÍ voy a 8
    o Si es NO voy a 5
8. Si (a > b) entonces
    o Variable mcm = a
    y si no
    o Variable mcm = b
9. Si (mcm % a == 0 Y mcm % b == 0)
    o Si es SÍ voy a 12
    o Si es NO voy a 10
10. CÁLCULO mcm = mcm + 1
11. Volver a paso 9
12. MOSTRAR "El mínimo común múltiplo de a y b es mcm"
13. FIN
  
```

Ejercicio 17:

Crea un algoritmo que encuentre el máximo común divisor de dos números.

Nota: El máximo común divisor (MCD) es el mayor número que divide a ambos números con resto de división igual a 0 (sin resto).

Diagrama de flujo:



Pseudocódigo:

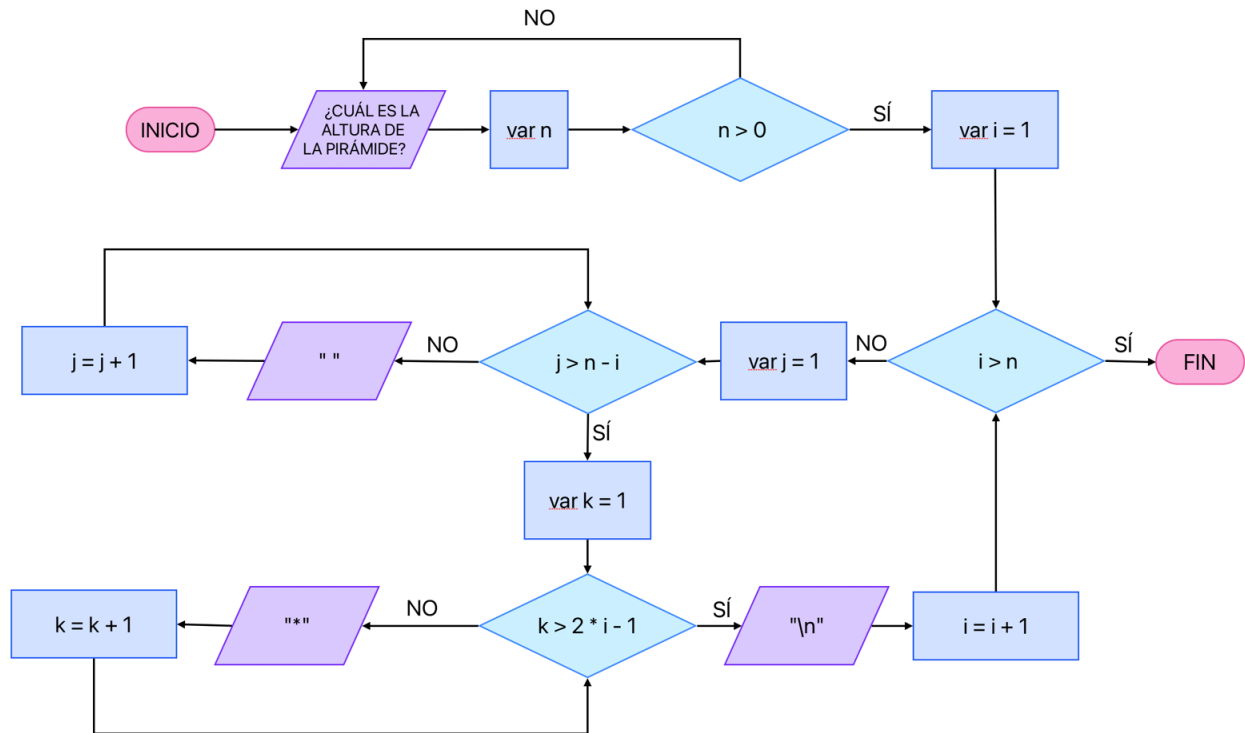
```

1. INICIO
2. PREGUNTAR "¿Cuál es el primer número?"
3. Variable a = Respuesta Paso 2
4. Si (a > 0)
    o Si es SÍ voy a 5
    o Si es NO voy a 2
5. PREGUNTAR "¿Cuál es el segundo número?"
6. Variable b = Respuesta Paso 5
7. Si (b > 0)
    o Si es SÍ voy a 8
    o Si es NO voy a 5
8. Si (b == 0)
    o Si es SÍ voy a 13
    o Si es NO voy a 9
9. Variable aux = b
10. CÁLCULO b = a % b
11. CÁLCULO a = aux
12. Volver a paso 8
13. MOSTRAR "El máximo común múltiplo es a"
14. FIN
  
```

Ejercicio 18:

Diseña un algoritmo que imprima una pirámide de asteriscos de altura N.

Diagrama de flujo:



Pseudocódigo:

1. INICIO
2. PREGUNTAR “¿Cuál es la altura de la pirámide?”
3. Variable n = Respuesta Paso 2
4. Si (a > 0)
 - Si es SÍ voy a 5
 - Si es NO voy a 2
5. Variable i = 1
6. Si (i > n)
 - Si es SÍ voy a 20
 - Si es No voy a 7
7. Variable j = 1
8. Si (j > n - i)
 - Si es SÍ voy a 12
 - Si es NO voy a 9
9. MOSTRAR “ ”
10. CÁLCULO j = j + 1
11. Volver a paso 8
12. Variable k = 1
13. Si (k > 2 * i -1)
 - Si es SÍ voy a 17
 - Si es NO voy a 14
14. MOSTRAR “*”
15. CÁLCULO k = k + 1
16. Volver a paso 13
17. MOSTRAR “\n”
18. CÁLCULO i = i + 1

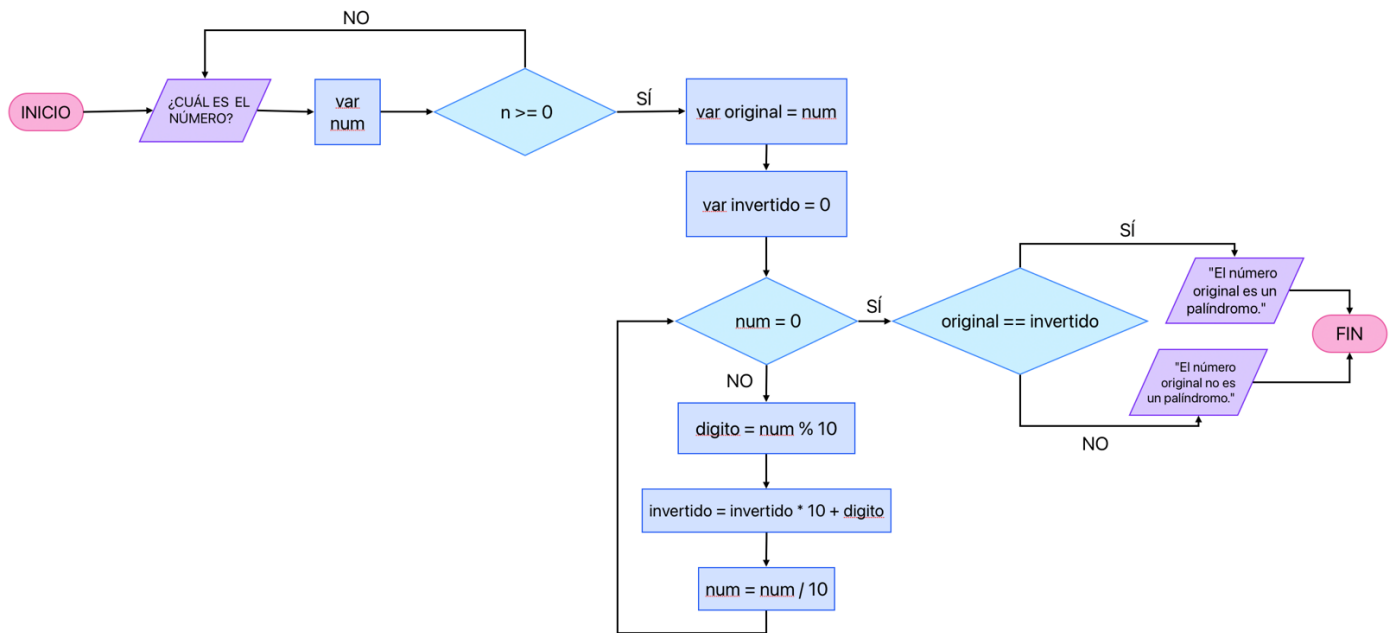
19. Volver a paso 6
20. FIN

Ejercicio 19:

Crea un algoritmo que determine si un número es un palíndromo.

Nota: Un palíndromo es un número que se lee igual de izquierda a derecha que de derecha a izquierda.

Diagrama de flujo:



Pseudocódigo:

1. INICIO
2. PREGUNTAR “¿Cuál es el número?”
3. Variable num = Respuesta Paso 2
4. Si (a > 0)
 - Si es SÍ voy a 5
 - Si es NO voy a 2
5. Variable original = num
6. Variable invertido = 0
7. Si (num = 0)
 - Si es SÍ voy a 12
 - Si es NO voy a 8
8. CÁLCULO digito = num % 10
9. CÁLCULO invertido = invertido * 10 + digito
10. CÁLCULO num = num / 10
11. Volver a paso 7
12. Si (original == invertido) entonces
 - MOSTRAR “El número original es un palíndromo.”y si no
 - MOSTRAR “El número original no es un palíndromo.”
13. FIN

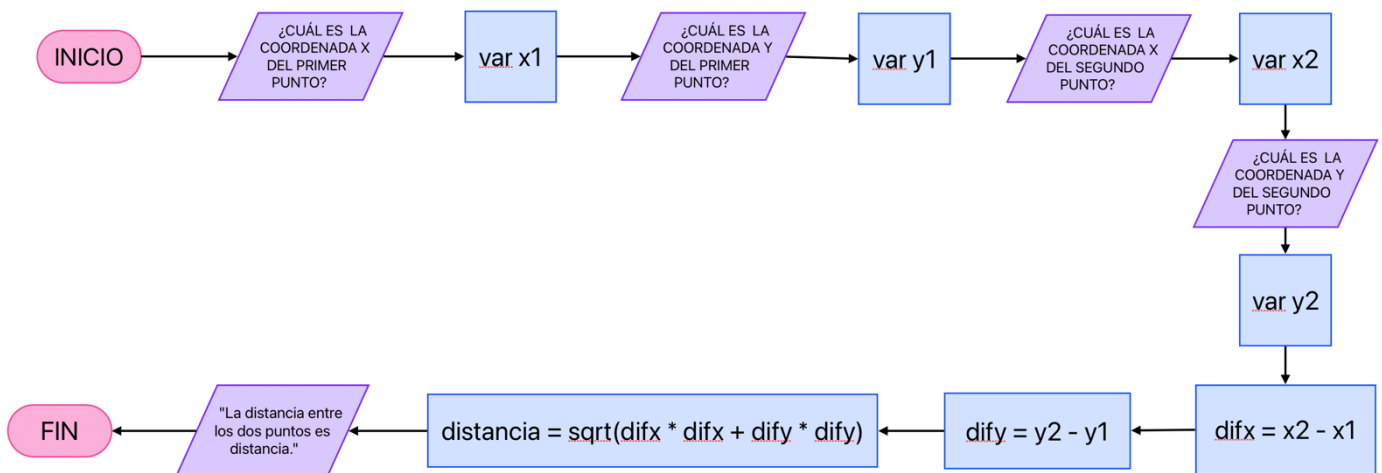
Ejercicio 20:

Crea un algoritmo que calcule la distancia entre dos puntos en un plano cartesiano. Los puntos se definen por sus coordenadas: (x_1, y_1) y (x_2, y_2) .

Nota: La fórmula para calcular la distancia entre dos puntos en un plano cartesiano: Distancia = $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$.

*Para calcular la raíz cuadrada usa un proceso predefinido llamado **sqrt**.*

Diagrama de flujo:



Pseudocódigo:

1. INICIO
2. PREGUNTAR "¿Cuál es la coordenada x del primer punto?"
3. Variable x1 = Respuesta Paso 2
4. PREGUNTAR "¿Cuál es la coordenada y del primer punto?"
5. Variable y1 = Respuesta Paso 4
6. PREGUNTAR "¿Cuál es la coordenada x del segundo punto?"
7. Variable x2 = Respuesta Paso 6
8. PREGUNTAR "¿Cuál es la coordenada y del segundo punto?"
9. Variable y2 = Respuesta Paso 8
10. CÁLCULO difx = x2 - x1
11. CÁLCULO dify = y2 - y1
12. CÁLCULO distancia = sqrt(difx * difx + dify * dify)
13. MOSTRAR "La distancia entre los dos puntos es distancia."
14. FIN