



Primer acercamiento

Debido a que la primera solución de kimi es incorrecta intentaremos una solución diferente

Orden 1

$a[0,0]=1$

$n=1$
1

n^2-1
1 0 a 0

Orden 2

$a[0,0]=1$ $a[0,1]=2$
 $a[1,0]=3$ $a[1,1]=4$

$n=2$
1,2,1

3 0 a 2

Orden 3

$a[0,0]=1$ $a[0,1]=2$ $a[0,2]=6$
 $a[1,0]=3$ $a[1,1]=5$ $a[1,2]=7$
 $a[2,0]=4$ $a[2,1]=8$ $a[2,2]=9$

$n=3$
1,2,3,2,1

5 0 a 4

Orden 4

$a[0,0]=1$ $a[0,1]=2$ $a[0,2]=6$ $a[0,3]=7$
 $a[1,0]=3$ $a[1,1]=5$ $a[1,2]=8$ $a[1,3]=13$
 $a[2,0]=4$ $a[2,1]=9$ $a[2,2]=12$ $a[2,3]=14$
 $a[3,0]=10$ $a[3,1]=11$ $a[3,2]=15$ $a[3,3]=16$

$n=4$
1,2,3,4,3,2,1

7 0 a 6

Orden 5

$a[0,0]=1$ $a[0,1]=2$ $a[0,2]=6$ $a[0,3]=7$ $a[0,4]=15$
 $a[1,0]=3$ $a[1,1]=5$ $a[1,2]=8$ $a[1,3]=14$ $a[1,4]=16$
 $a[2,0]=4$ $a[2,1]=9$ $a[2,2]=13$ $a[2,3]=17$ $a[2,4]=22$
 $a[3,0]=10$ $a[3,1]=12$ $a[3,2]=18$ $a[3,3]=21$ $a[3,4]=23$
 $a[4,0]=11$ $a[4,1]=19$ $a[4,2]=20$ $a[4,3]=24$ $a[4,4]=25$

$n=5$ 9 0 a 8
1,2,3,4,5,4,3,2,1

Orden 6

$a[0,0]=1$ $a[0,1]=2$ $a[0,2]=6$ $a[0,3]=7$ $a[0,4]=15$ $a[0,5]=16$
 $a[1,0]=3$ $a[1,1]=5$ $a[1,2]=8$ $a[1,3]=14$ $a[1,4]=17$ $a[1,5]=26$
 $a[2,0]=4$ $a[2,1]=9$ $a[2,2]=13$ $a[2,3]=18$ $a[2,4]=25$ $a[2,5]=27$
 $a[3,0]=10$ $a[3,1]=12$ $a[3,2]=19$ $a[3,3]=24$ $a[3,4]=28$ $a[3,5]=33$
 $a[4,0]=11$ $a[4,1]=20$ $a[4,2]=23$ $a[4,3]=29$ $a[4,4]=32$ $a[4,5]=34$
 $a[5,0]=21$ $a[5,1]=22$ $a[5,2]=30$ $a[5,3]=31$ $a[5,4]=35$ $a[5,5]=36$

$n=6$ 11 0 a 10

1, 2, 3, 4, 5, 6, 5, 4, 3, 2, 1

Construir código flowgorithm para que dada una "n" genere la siguientes secuencias de números

para n=1 Secuencia= 1

para n=2 Secuencia= 1,2,1

para n=3 Secuencia= 1,2,3,2,1

para n=4 Secuencia= 1,2,3,4,3,2,1

para n=5 Secuencia= 1,2,3,4,5,4,3,2,1

para n=6 Secuencia= 1,2,3,4,5,6,5,4,3,2,1

Y así sucesivamente para una n cualquiera. Utilizar únicamente un ciclo para generar las secuencias de números.

INSTRUCTOR



Solicitando a Kimi

Construir código flowgorithm para que dada una "n" genere la siguientes secuencias de números

para n=1 Secuencia= 1

para n=2 Secuencia= 1,2,1

para n=3 Secuencia= 1,2,3,2,1

para n=4 Secuencia= 1,2,3,4,3,2,1

para n=5 Secuencia= 1,2,3,4,5,4,3,2,1

para n=6 Secuencia= 1,2,3,4,5,6,5,4,3,2,1

Y así sucesivamente para una n cualquiera. Utilizar únicamente un ciclo para generar las secuencias de números.



🔧 Secuencia "montaña" con un solo ciclo

Patrón: sube hasta n y baja hasta $1 \rightarrow$ longitud $2n - 1$.

Guarda como `MontañaUnCiclo.fprg`

```
<?xml version="1.0"?>
<flowgorithm fileversion="3.0">
  <function name="Main" type="None">
    <parameter/>
    <body>
      <!-- Entrada -->
      <declare name="n" type="Integer" array="False"/>
      <output text="n: "/>
      <input variable="n"/>
      <if expression="n &lt;= 0">
        <then>
          <output text="Error: n debe ser mayor que 0" newline="True"/>
          <return/>
        </then>
      </if>

      <!-- Longitud = 2n - 1 -->
      <declare name="len" type="Integer" array="False"/>
      <assign variable="len" expression="2 * n - 1"/>

      <!-- Arreglo 1D -->
      <declare name="S" type="Integer" array="True" size="1000"/>

      <!-- UN SOLO CICLO: sube y baja con matemática -->
      <for variable="k" start="0" end="len - 1" direction="inc" step="1">
        <assign variable="S[k]" expression="n - abs(n - 1 - k)"/>
      </for>

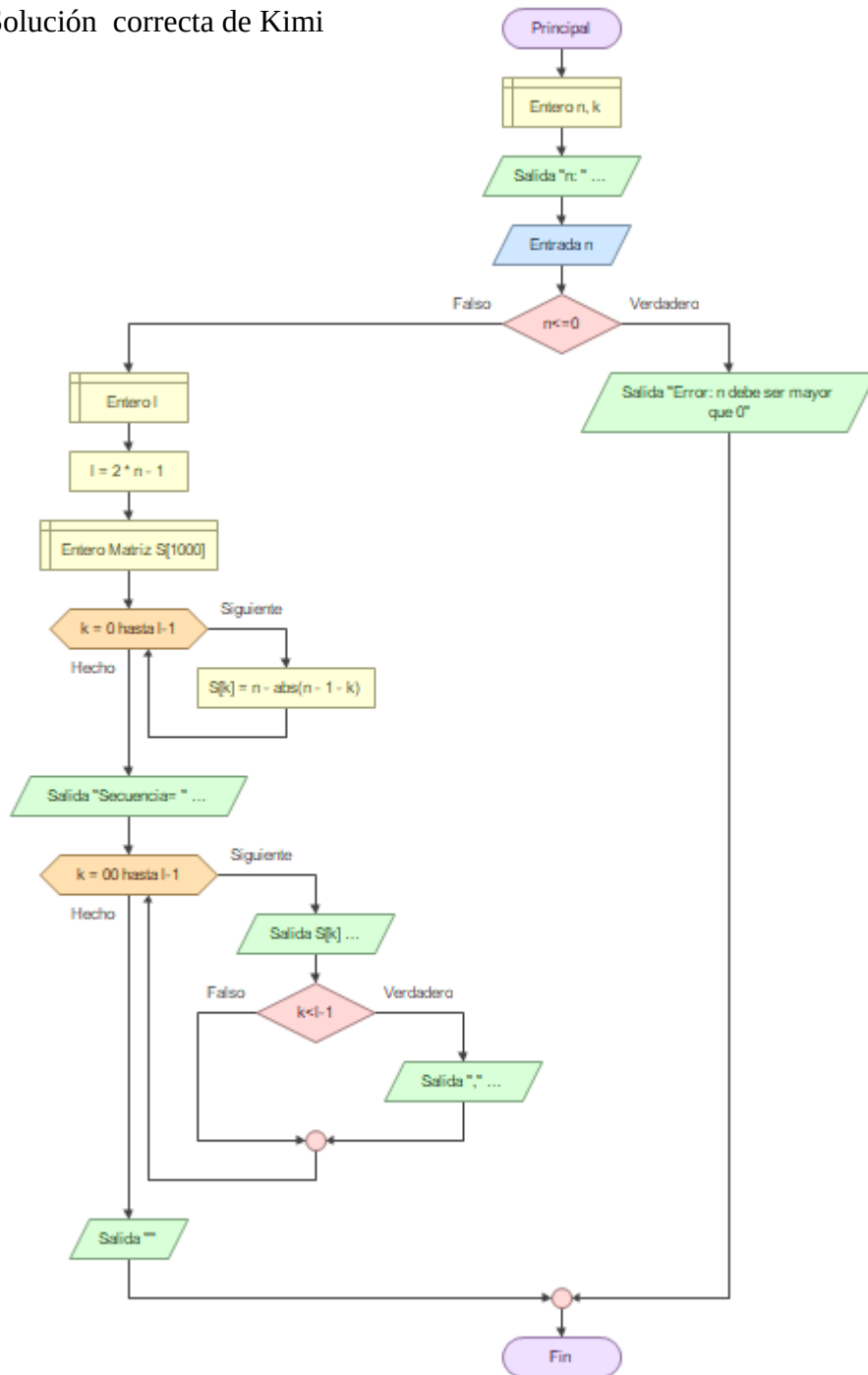
      <!-- Mostrar resultado -->
      <output text="Secuencia= " newline="False"/>
      <for variable="k" start="0" end="len - 1" direction="inc" step="1">
        <output expression="S[k]" newline="False"/>
        <if expression="k &lt; len - 1">
          <then><output expression="&quot;;&quot;" newline="False"/></then>
        </if>
      </for>
      <output expression="&quot;&quot;" newline="True"/>
    </body>
  </function>
</flowgorithm>
```

Guarda → importa → ejecuta:

Un solo ciclo → montaña lista para cualquier $n \geq 1$.



Solución correcta de Kimi



Ejecución

