

$$3n+1$$

Dado un número N, le aplicaremos las siguientes reglas:

- Si N es par entonces dividimos N entre dos
- Si es impar entonces multiplicamos N por 3 y le sumamos 1

Repetimos el proceso hasta que N sea igual a 1.

Por ejemplo, si  $N=20$ , entonces la secuencia será:

**20, 10, 5, 16, 8, 4, 2, 1**

En este caso, se necesitaron 7 pasos para llegar al uno.

### **Entrada:**

Tu programa deberá leer de entrada estándar el número N.

### **Salida:**

Tu programa deberá escribir a salida estándar el número de pasos que se necesitan para llegar al uno.

### **Consideraciones:**

- $1 \leq N \leq 100,000$

### **Ejemplo:**

<b>Entrada</b>	<b>Salida</b>
20	7
937	173

# Cuenta primos

## Problema:

Escribe un programa que dados dos enteros A y B cuente el número de primos que hay entre A y B (inclusive). Por ejemplo, si A=3 y B=13 entonces los primos que están entre A y B inclusive son 3,5,7,11 y 13, por lo tanto tu programa debe imprimir 5.

## Entrada:

Tu programa debe leer de la primera línea de entrada estándar los números A y B separados por un espacio.

## Salida:

Tu programa debe imprimir a salida estándar el número de primos que hay entre A y B inclusive.

## Consideraciones:

- $1 \leq A, B \leq 10,000,000$
- 1 no es primo

## Ejemplos:

Entrada	Salida
3 13	5
1 2	1
18 123	40

# Pascal

Las primeras 5 filas del “triángulo de Pascal” son:

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

el primero y último números de cada fila son siempre 1's, el número que va en el resto de las casillas es la suma de los dos primeros números vecinos de la fila anterior.

## Problema:

Escribe un programa que dado un número N imprima la N-ésima fila del triángulo de pascal.

## Entrada:

Tu programa deberá leer de entrada estándar el número N.

## Salida:

Tu programa deberá imprimir a salida estándar la N-ésima fila del triángulo de pascal.

## Consideraciones:

- $1 \leq N \leq 20$

## Ejemplos:

Entrada	Salida
3	1 2 1
5	1 4 6 4 1

# Tablero

Se quiere pintar un tablero con pintura azul y roja. El tablero está dividido en N cuadrados y cada cuadrado tiene un dibujo ya sea de un punto o un “#” (ver figura 1). Solo queremos pintar los cuadrados que tienen puntos (y dejar despintados los que tienen #’s). Además **no queremos** que dos cuadrados vecinos tengan el mismo color (ver figura 2).

Claramente, el color rojo es mejor que el azul, así que queremos pintar de rojo la mayor cantidad de cuadrados que sea posible.



Figura 1 (tablero inicial)



Figura 2(un posible tablero final, ningún par de tableros vecinos tiene el mismo color)

## Problema:

Dada la descripción del tablero, calcula el máximo número de cuadros que se pueden pintar de rojo de tal forma que dos cuadrados vecinos tengan el mismo color.

## Entrada:

Tu programa deberá leer de la primera línea de entrada estándar el número N de cuadrados que tiene el tablero. En la segunda línea estará una cadena de caracteres describiendo el tablero.

## Salida:

Tu programa deberá escribir a salida estándar el máximo número R de cuadrados que se pueden pintar de rojo.

## Consideraciones:

- $1 \leq N \leq 1000$
- Todos los caracteres de la cadena de texto serán ya sea ‘.’ o ‘#’.

## Ejemplos:

Entrada	Salida
5 #...#	2
20 #.....#.....#.....#..#	8