

## JUEZ - Paquetes Consecutivos

### Descripción:

En una empresa de paquetería, los paquetes se almacenan ordenados de menor a mayor peso. Con el fin de optimizar el espacio, se desea saber cuántos paquetes consecutivos empezando en el primero tienen un peso que difiere como máximo en  $K$  unidades **respecto al paquete anterior** en la secuencia.

Por ejemplo, si tenemos los paquetes  $[1, 2, 3, 5, 8, 9]$  y  $K = 1$ , la secuencia más larga comenzaría en el 1 e incluiría el 2 y el 3, teniendo una longitud de 3. Esto es porque el 5 difiere en más de 1 unidad del paquete anterior en la secuencia (el 3).

### Se pide:

1. Implementar una **función recursiva** que **devuelva la longitud** de la **secuencia más larga de paquetes consecutivos** que difieren como máximo en  $K$  unidades con respecto al paquete anterior en la secuencia.
2. Calcular el coste del algoritmo y analizar si es el más eficiente para un número grande de paquetes (por ejemplo, cerca de 100,000).

### Entrada:

La entrada consiste en una serie de casos de prueba. Cada caso de prueba consta de tres líneas:

- Primera línea: número de paquetes  $N$  y diferencia máxima  $K$
- Segunda línea:  $N$  números ordenados que representan los pesos de los paquetes
- La entrada termina cuando  $N=0$ .

El número de paquetes será mayor que 0 y menor que 100,000. Los pesos son números enteros entre 1 y 1000.

### Salida:

Para cada caso de prueba se escribirá en una línea la longitud de la secuencia más larga encontrada.

→ Ejemplo de Entrada:

```
1 3
1
5 2
10 10 10 10 10
12 1
1 2 3 4 5 6 7 8 9 10 11 12
5 0
3 3 3 3 3
5 0
1 2 3 4 5
0 0
```

→ Ejemplo de Salida:

```
1
5
12
5
1
```

Dale caña y mucho ánimo 💪

