

Números entretenidos

El problema

Se dice que un número es *entretenido* si no tiene dos dígitos seguidos iguales. Por ejemplo, 1343 es entretenido, y 1334 no es entretenido.

Debe diseñarse un algoritmo que, dado un número $n \geq 0$, determine la cantidad de números entretenidos menores o iguales que n (es decir, la cantidad de números entretenidos en el intervalo $[0, n]$).

Trabajo a realizar

Para realizar el control se proporciona un archivo `plantilla.cpp`. El código proporcionado supone que el punto de entrada al algoritmo es la función `num_entretenidos`. Si se comenta la definición `#define DOM_JUDGE` el resultado del algoritmo se contrasta, sobre 1000 casos de prueba generados aleatoriamente, con una implementación *naïf* del mismo. La definición debe estar des-comentada a la hora de subirla al juez.

Cada caso de prueba consiste en el número n . Los casos terminan con una línea que contiene -1.

Ejemplo de entrada / salida:

| Entrada | Salida |
|--------------------|-------------------|
| 0 | 1 |
| 9 | 10 |
| 44 | 41 |
| 123456789123456789 | 21107057695839172 |
| -1 | |

Tu trabajo consiste en:

- Diseñar el algoritmo, rellenando los comentarios incluidos en el archivo.
- Implementar este algoritmo.
- Entregar la solución a través del juez en línea de la asignatura.

Importante:

- No modificar el código proporcionado. Únicamente deben responderse a los distintos apartados, en el interior de los comentarios, implementar la función `num_entretenidos` y todas las funciones auxiliares que se consideren necesarias. Si se quiere que el programa se ejecute en modo *autocorrección*, puede comentarse `#define DOM_JUDGE`. Pero debe recordarse que dicha definición debe estar des-comentada cuando se realice la entrega en el juez.