

Fundamentos de algoritmia
Grados en Ingeniería Informática. Grupo C

22 de Octubre de 2019.

Nombre: _____ Grupo: _____

Laboratorio: _____ Puesto: _____ Usuario de DOMjudge: _____

Normas de realización de la misión

1. Debes programar soluciones para cada ejercicio, probarlas y entregarlas en el juez automático accesible en la dirección <http://exacrc.domjudge/team>.
2. Escribe comentarios que expliquen tu solución, justifiquen por qué se ha hecho así y ayuden a entenderla. Calcula la complejidad de todas las funciones que implementes.
3. En el juez te identificarás con el nombre de usuario y contraseña que has recibido al comienzo del examen. El nombre de usuario y contraseña que has estado utilizando durante la evaluación continua **no** son válidos.
4. Escribid vuestro **nombre y apellidos** y nick en un comentario en la primera línea de cada fichero que subais al juez.
5. Tus soluciones serán evaluadas por el profesor independientemente del veredicto del juez automático. Para ello, el profesor tendrá en cuenta **exclusivamente** el último envío que hayas realizado de cada ejercicio.

1. Dado un vector de enteros, se desea encontrar el segmento más largo de números consecutivos. Se pide:
 1. Especifica una función que dado un vector de enteros de longitud ≥ 0 devuelva la longitud del segmento más largo del vector formado por una secuencia de números consecutivos.
 2. Diseña e implementa un algoritmo iterativo que resuelva el problema propuesto.
 3. Escribe el invariante del bucle que permite demostrar la corrección del mismo y proporciona una función de cota.
 4. Indica el coste asintótico del algoritmo en el caso peor y justifica adecuadamente tu respuesta.

Entrada

La entrada comienza con una línea que contiene el número de casos de prueba. Cada caso de prueba contendrá la longitud del vector, y a continuación los valores de tipo `int` que contiene el vector.

Salida

Por cada caso de prueba el programa escribirá una línea con la longitud del segmento más largo solicitado.

Entrada de ejemplo

```
6
0
1
5
2
3 2
10
2 3 2 3 4 5 6 5 4 3
10
1 1 2 3 4 8 7 6 5 6
10
-2 2 2 2 2 2 2 2 2 2
```

Salida de ejemplo

```
0
1
2
10
5
1
```