

Pregunta **9**

Respuesta  
guardada

Puntúa como 5

Se tiene un programa P1 que recibe como parámetro un vector de tamaño  $n$ . Sabemos que el programa tiene dos ciclos anidados.  
¿Qué podemos decir respecto a su complejidad de peor caso?

Seleccione una o más de una:

- ☐ a. No se puede afirmar nada con respecto a la complejidad sin conocer el código del programa
- ☐ b. Si tiene dos ciclos anidados, su complejidad es de  $O(n^2)$
- ☒ c. No se puede decir con precisión qué complejidad tiene el programa, pero como tiene dos ciclos anidados podemos afirmar que es al menos  $O(n^2)$  (podría ser  $O(n^2)$ ,  $O(n^3)$ , ...)
- ☒ d. La complejidad del programa será bien  $O(n^2)$  o  $O(n^3)$ , dependiendo si dentro de los ciclos el código es de complejidad lineal o no.

Pregunta **10**

Respuesta  
guardada

Puntúa como 5

Para un vector  $v$ , decimos que un **valle** es un subarreglo de valores que cumple que

- todos los valores del subarreglo son iguales
- Los vecinos inmediatos del subarreglo (si existen) son de valor mayor estricto que los valores del subarreglo.

El siguiente programa calcula **el valle más largo** de un vector.

```
int valleMasLargo(vector<int> &v) {
    int i = 0;
    int maxValle = 0;
    int valle;
    while (i < v.size()) {
        int j = i + 1;
        while (j < v.size() && v[i] == v[j]) {
            j++;
        }
        if ( (i == 0 || v[i] < v[i-1])
            && (j == v.size() || v[j] > v[j-1]) ) {
            valle = j - i;
        }
        i = j;
        if (valle > maxValle) {
            maxValle = valle;
        }
    }
    return maxValle;
}
```

Seleccione su complejidad de peor caso en función de  $n$  donde  $n = |v|$ .

Si además este problema se puede resolver en una complejidad mejor, también seleccione la mejor complejidad de la lista.

Seleccione una o más de una:

- ☒ a.  $O((\log n)^2)$
- ☐ b.  $O(1)$
- ☐ c.  $O(n)$
- ☒ d.  $O(n \log n)$