**¿Cuál es la dificultad computacional del problema, por qué es difícil de computar?**

* El problema es difícil de computar puesto que se recibe una cantidad N de números, sobre los cuales se aplica una cantidad K de operaciones. Estas operaciones se realizan sobre rangos determinados y no sobre la totalidad de números, produciendo que no sólo los números sean importantes, sino que también sus posiciones. Esto hace que el problema sea difícil de computar.

**Complejidad del algoritmo más básico que se puede hacer**

* El algoritmo más básico que se puede hacer es cada vez que se necesite determinar un valor dentro de un rango determinado de los números, es calcularlo cada vez iterando sobre estos. Esto hace que la complejidad de actualizar o de obtener un valor dentro de un rango, tienda a ser n en el peor caso (cuando se opere sobre el rango completo de números), mientras que por cada operación k, se hacen estas actualizaciones. Esto nos lleva a una complejidad de N\*K. Si consideramos que K puede tener un valor cercano a N, hablamos de una complejidad del orden **.**

**Complejidad que se necesita para resolver el problema**

* Para resolver este problema, lo ideal sería utilizar la estructura de lazy segment tree, dado que cuando se hacen las operaciones, no se hace totalmente necesario actualizar todo el arreglo inmediatamente, sino que cuando se haga la query, se actualizan los valores necesarios. Al implementar esta estructura en el problema, se reduce la complejidad considerablemente en comparación al algoritmo más básico que resuelve el problema.

**Explicar la solución implementada**

* La solución implementada utiliza la estructura de segment tree. Dado que esta estructura funciona con arreglos cuyo largo debe ser una potencia de 2, cada vez que se ingresa una cantidad que no cumpla esta condición, el árbol es completado con neutros multiplicativos (1). Una vez hecho esto, se realizan las operaciones sobre el segment tree.