

Juego de Cartas.

Se tiene un juego de cartas numeradas de 1 a N, sin que existan dos cartas con el mismo número. Se “baraja” el mazo de cartas de manera que quedan desordenadas. Se comienza por la primera carta del mazo hasta que se encuentra la carta de número 1, se elimina y luego se busca la carta 2, se elimina y se busca la próxima carta repitiendo el mismo proceso hasta que se elimina la carta N, que es cuando el juego termina. Si se llega a la última carta del mazo y quedan cartas por eliminar no se ha terminado de eliminar cartas del juego se comienza por la primera carta del mazo a buscar la próxima carta.

Por ejemplo si se tiene el juego de cartas de 1 a 5 y un el orden es: {2, 3, 1, 5, 4}. Se busca primero la carta 1, esta se encuentra en la posición 2, se elimina quedando el mazo de cartas {2, 3, 5,4} ahora se elimina la carta de número 2 del mazo que se encuentra en la posición 0 por lo que queda {3, 5,4}, ahora se elimina la carta 3 que se encuentra en la posición 0, quedando el mazo {5, 4}, luego se elimina la carta 4 que se encuentra en la posición 1, y luego se elimina la carta 5 que está en la única posición que queda, que evidentemente será en la posición 0.

Lo que se quiere en este caso como respuesta es un array con las posiciones que hemos ido eliminando, lo que en este ejemplo es el array {2, 0, 0, 1,0}

Implemente el método `JuegoCartas` que recibe como parámetro el array de int con las cartas y retorne un array con las posiciones en las que se hallaron las cartas en los mazos que fueron quedando.

Para el ejemplo:

```
int[] juego = new int[5]{ 2, 3, 1, 5,4};  
int[] respuesta = JuegoCartas(juego);  
  
// Respuesta es igual al array {2, 0, 0, 1,0}
```

Suponga que el mazo inicial tiene al menos una carta de modo que no tiene que controlar esto.