

Pelota Juguetona

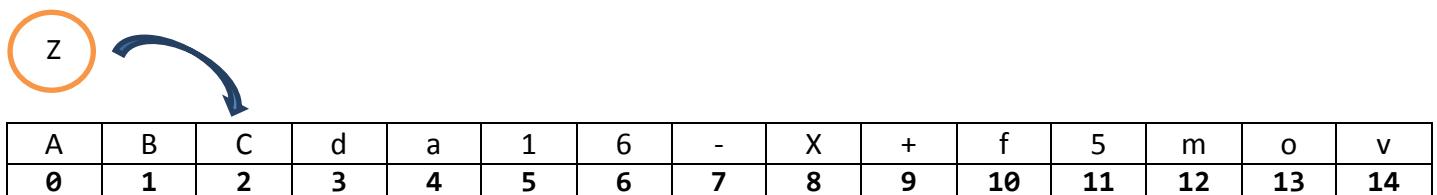
Implemente una DLL de nombre `PelotaJuguetona.dll` que tenga la siguiente clase

```
namespace PelotaJuguetona
{
    public class Prueba
    {
        public static char[] Brinca(char[] array, int[] saltos, char caracterInicial)
        {
            // ... Aquí va su implementación
        }
    }
}
```

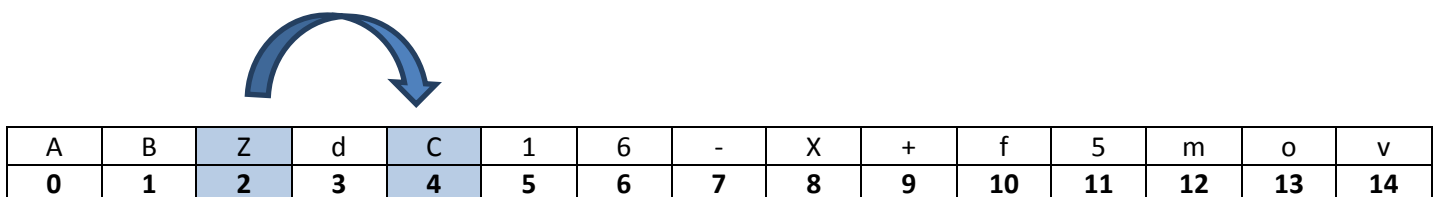
El método `Brinca` recibe varios parámetros, el primero de ellos es una array de `char` que representa el estado inicial del array, el segundo parámetro es un array de `int` que representa los saltos que va a dar la pelota en cada momento y el tercer parámetro es el carácter inicial con el que comienza brincando la pelota.

Veamos el siguiente ejemplo para los siguientes parámetros:

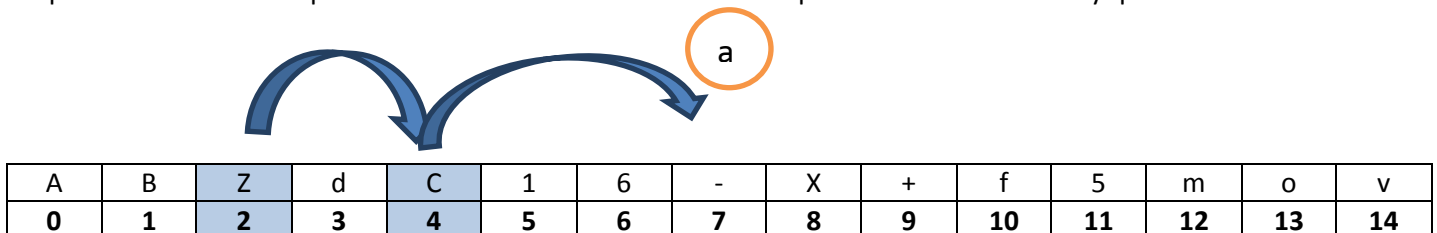
```
char[] array = {'A','B','C','d','a','1','6','-','X','+','f','5','m','o','v'};
int[] saltos = {2,2,3,8,1,5};
char caracterInicial = 'Z';
```



Como puede ver en la imagen anterior la pelota da un primer salto de 2 y cae por tanto en la posición 2 del array y ahí debe dejar el carácter 'Z' luego sigue recorriendo el array de saltos, que en este ejemplo es de nuevo 2 y realiza un nuevo salto llevando ahora la 'C', cae ahora en la posición 4 y deja ahí la 'C' para llevarse la 'a'.



El próximo salto es de 3 por tanto debe de cambiar el carácter '-' por el carácter 'a' el array queda:



Ahora debe llevarse el '-' y saltar 8 lugares, pero como puede apreciar se le acabó el array, si esto ocurre entonces la pelota comienza de nuevo por el inicio del array.

El proceso termina cuando la pelota tuviese que saltar a una posición por la que ya ha pasado o cuando se haya recorrido todo el array de saltos. .

El método que usted implemente no debe afectar el array sino retornar un nuevo array de caracteres resultante de las operaciones que la pelota hizo.

El array final resultante del ejemplo es:

-	A	Z	d	C	1	B	a	X	+	f	5	m	o	v
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Para simplificar considere (de modo que no tiene que verificarlo) que:

- el array de caracteres nunca es nulo y tiene al menos un elemento.
- el array de saltos nunca es nulo y tiene al menos un elemento.

Sin embargo, sí debe controlar que un salto no sea negativo en cuyo caso deberá lanzar una excepción con el siguiente mensaje: "Un salto no puede ser negativo".

Le recomendamos tratar de compilar su proyecto lo más frecuentemente posible desde que empiece a trabajar. Entre otros beneficios, esto disminuirá la probabilidad de que pierda su trabajo si ocurre alguna falla o interrupción.