

Tarea UVA7: Listas

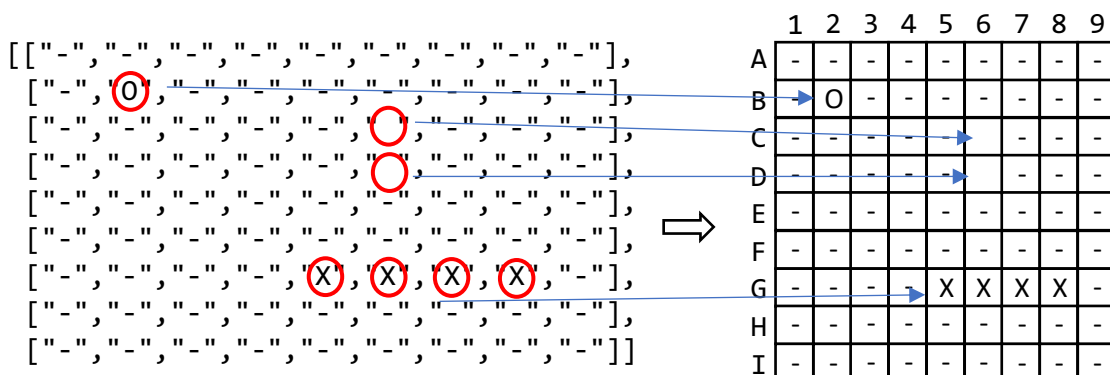


La famosa desarrolladora de videojuegos Pyntendo está por lanzar un novedoso juego nunca antes visto: *Solitary Battleship*. En este juego, la computadora ubica barcos de distintos tamaños en posiciones aleatorias dentro de un tablero de tamaño 9x9. Dichos barcos están ocultos para el jugador, quien deberá realizar disparos a ciegas hasta destruir todas las naves.

La mayor parte del código del juego ya está listo, pero a los desarrolladores les faltan algunas funciones que se encarguen de actualizar el tablero, luego de cada disparo. Podrá encontrar un archivo de código incompleto junto al enunciado de esta tarea. Descárguelo y desarrolle en él las funciones solicitadas, en el área indicada.

1.- Desarrolle la función `disparo(tablero, barcos, fila, columna)` que recibe los siguientes 4 parámetros:

- **tablero:** corresponde a una lista de listas de tamaño 9x9 que representa las casillas del tablero del juego. Cada casilla puede contener 4 posibles valores de tipo *string*
 - "-" (guión): Indica que la casilla es desconocida para el jugador, es decir, aún no ha recibido disparos. Todas las casillas presentan este valor al iniciar el juego.
 - " " (espacio): Indica que la casilla solo contiene agua. Se debe mostrar cuando el jugador ha disparado a esta casilla, pero no ha encontrado un barco.
 - "O": Indica que la casilla está ocupada por un barco, pero aún no ha sido destruido por completo.
 - "X": Indica que la casilla está ocupada por un barco que ya fue destruido. Un barco se considera destruido cuando ya se ha disparado a cada una de las casillas que ocupa, según su tamaño y orientación.



Por ejemplo, la lista presentada arriba, representa un tablero en donde se han realizado 7 disparos. Se ha descubierto que en B2 hay un barco que aún no ha sido destruido por completo, en C6 y D6 no hay barcos, y en G5, G6, G7 y G8 había un barco de largo 4 que ya fue destruido.

- **barcos:** Una lista de listas con la información de los barcos. Contiene listas, cada una conformada por 4 números enteros que indican el **largo del barco (número de casillas)**, la **orientación (vertical = 1 y horizontal = 2)**, y la **fila y columna (valores entre 0 y 8) de la primera casilla (la de más arriba o más hacia la izquierda)** ocupada por el barco. A modo de ejemplo, la lista barcos correspondiente al tablero mostrado anteriormente podría ser:

```
[ [2, 1, 4, 8], # 2 es el largo del barco, 1 indica vertical, ubicado en fila E columna 9
  [3, 2, 1, 0],
  [3, 1, 6, 1],
  [4, 2, 6, 4],
  [5, 1, 1, 4] ]
```

Revisando el ejemplo para un barco en particular, vemos que la sublista `[4, 2, 6, 4]` representa el barco destruido: un barco de largo 4, con orientación horizontal que comienza en la casilla G5.

- **fila:** Entero entre 0 y 8 que representa la fila en la que se realiza el disparo.
- **columna:** Entero entre 0 y 8 que representa la columna en la que se realiza el disparo.

Esta función debe actualizar el tablero, modificando el valor en la casilla a la cual se disparó. El nuevo valor en dicha casilla será "O" si hay un barco en esa ubicación, o " " en caso de haber solo agua. La función **disparo** no retorna ningún valor.

2. La función **destruidos(tablero, barcos)** recibe como parámetros las listas **tablero** y **barcos** mencionadas anteriormente. Esta función **debe verificar los barcos que estén completamente destruidos** y en esos casos, **cambiar el valor de cada una de sus casillas a "X"**. La función **retorna el número total de barcos destruidos en el tablero**.

El programa muestra el estado del tablero luego de cada disparo, por lo que puede utilizar esta información para comprobar que la función esté correcta. A modo de ejemplo, consideremos que **barcos** corresponde a la lista `[2, 1, 1, 1]` y el tablero actualmente se encuentra como en el ejemplo mostrado anteriormente. Entonces, luego de disparar en “C2”, su función debería retornar 2 y el tablero que mostraría el programa sería el siguiente:

	1	2	3	4	5	6	7	8	9
A	-	-	-	-	-	-	-	-	-
B	-	X	-	-	-	-	-	-	-
C	-	X	-	-	-		-	-	-
D	-	-	-	-	-		-	-	-
E	-	-	-	-	-	-	-	-	-
F	-	-	-	-			-	-	-
G	-		-	-	X	X	X	X	-
H	-	-	-	-	-	-	-	-	-
I	-	-	-	-			-	-	-