

Práctica

Tenemos un juego en el que se sitúan barcos en un tablero para dos jugadores y posteriormente los jugadores efectúan lanzamientos hasta hundir todos los barcos de los jugadores contrarios.

Un tablero tiene coordenadas desde la A a las J para las columnas y desde el 1 hasta el 10 para las filas, siendo A1 la coordenada de arriba a la izquierda.

Un barco está definido por el tipo de barco (Submarino, Fragata, Dragaminas, Portaaviones), por la posición de su proa (una casilla del tablero) y por su orientación (Horizontal o Vertical).

Un barco horizontal tiene la proa a la izquierda. Uno vertical tiene la proa arriba.

Los submarinos ocupan 1 casilla, las fragatas 2, los dragaminas 3, y los portaaviones 4.

En la entrada, tenemos datos para un dos jugadores, y, para cada jugador, tenemos datos de un número indeterminado de barcos.

Cada jugador en la entrada tiene barcos (quizá ninguno) seguidos por la palabra FIN.

Para cada barco, tenemos en este orden el tipo de barco, la coordenada de la proa y la orientación.

Puede haber cualquier cantidad de espacio en blanco en la entrada entre distintas palabras.

Además, la entrada puede ser incorrecta y en lugar de tener lo que se espera tener, podríamos tener cualquier otra cosa.

Además, los barcos podrían estar mal situados, de tal forma que alguno salga del tablero o de forma que alguno esté sobre otro parcial o totalmente.

Los barcos mal situados deben ignorarse.

Tras los datos de los jugadores, disponemos de disparos realizados alternativamente por los jugadores, hasta el fin del fichero o hasta que uno de los jugadores hunda la flota del jugador contrario.

Cada disparo de un jugador opera sobre el tablero del jugador contrario, y se considera Agua si no hay barco en la casilla, Tocado si se acierta en un barco pero dicho barco tiene casillas sin tocar y Hundido si se acierta en la última casilla sin tocar de un barco.

Un ejemplo de entrada correcta sería este.

Submarino

A1

Horizontal Fragata B

```

3 Vertical Submarino C5 Horizontal FIN Submarino
B
1
Vertical Fragata B 5
    Vertical
Fragata
C
5
Horizontal
FIN B5 A
3
C2 D 3 B
1

```

Se desea un programa en Pascal que

1. Lea estos datos de la entrada. Los barcos mal situados deben ignorarse, y el programa debe escribir un mensaje cuando lea un barco mal situado, indicando qué tipo de barco se ignora. Los disparos fuera del tablero simplemente hacen que el jugador desperdicie el turno.
2. Si hay errores en la entrada, escriba un mensaje informando de la entrada incorrecta y termine la ejecución ordenadamente sin leer nada mas (no se permite utilizar halt ni operaciones similares para terminar la ejecución).
3. Tras leer los datos de los barcos de los jugadores, dibuje el tablero para cada jugador, tal y como se muestra en el ejemplo, indicando el número del jugador antes de dibujar su tablero.

Para cada barco se utiliza la inicial del tipo de barco para marcarlo en la casilla.

A	B	C	D	E	F	G	H	I	J	
S	1
.	2
.	F	3
.	F	4
.	.	S	5
.	6
.	7
.	8
.	9
.	10

Para dibujar el tablero se sugiere buscar para cada casilla si dicha casilla forma parte de algún barco.

4. Tras leer cada disparo, escriba el disparo realizado y dibuje el tablero del jugador contrario (al que se dispara). Esta vez, además de dibujar la posición de los barcos como en el dibujo anterior, hay que marcar con una X las casillas de barcos que han sido tocadas, y con “,” las casillas que corresponden a disparos que han sido agua. Por ejemplo, como en esta salida

Disparo al jugador 2: E3 Agua

A	B	C	D	E	F	G	H	I	J	
S	1
.	2
.	X	3
.	X	4
.	.	S	5
.	6
.	7
.	8
.	9
.	10

que muestra el tablero del jugador 2 tras un disparo al agua (y dos disparos que han hundido un barco).

5. El programa debe terminar sin leer nada más de la entrada en el momento en que un jugador hunde todos los barcos del jugador contrario. En dicho momento debe escribir “Gana el jugador X”, donde X es el número del jugador que gana.

6. Si se termina la entrada y no hay ganador, el programa debe escribir “Empate” y terminar.

7. Antes de notificar un ganador o un empate, el programa debe escribir los movimientos efectuados por ambos jugadores, con el siguiente formato:

Jugador 1: A3 Agua
Jugador 2: B5 Tocado
Jugador 1: A4 Agua
Jugador 2: B6 Hundido
...

8. Puede suponerse un límite razonable para el número de barcos y el número de movimientos de cada jugador, pero se valorará que la implementación acepte un número ilimitado de barcos y movimientos.

9. Los datos de entrada proceden del fichero llamado datos.txt

10. El programa no puede escribir en la salida estándar nada salvo lo que se indica en el enunciado.

11. El programa no puede leer de la entrada nada que no se indique en el enunciado.

12. Ni que decir tiene que hay que utilizar los procedimientos, funciones y tipos de datos que sean necesarios para tener una implementación limpia.