

Battleship



El joc



1.- Introducció.

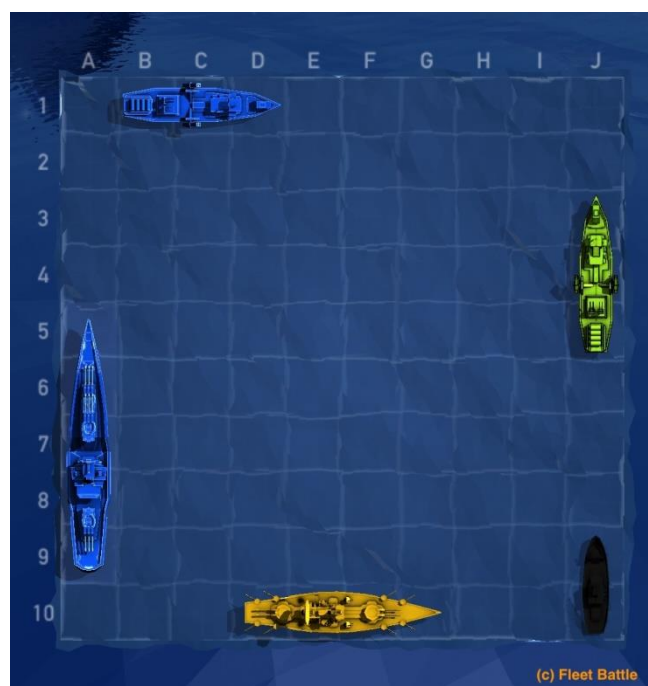
L'objectiu del projecte és crear un programa en Java que permeti jugar a una versió simple del joc **Battleship** "Hundir la Flota".

En el joc original, cada jugador maneja dos taulers dividits en caselles. Cada tauler representa una zona diferent de la mar oberta: la pròpia i la contrària.

En un dels taulers el jugador col·loca els seus vaixells i registra els «tirs» de l'oponent; en l'altre, es registren els tirs propis, al mateix temps que es dedueix la posició dels vaixells del contrincant.

En començar, cada jugador posiciona els seus vaixells en el primer tauler, de manera secreta, invisible a l'oponent.

Una vegada tots els vaixells han sigut posicionats, s'inicia una sèrie de rondes. En cada ronda, cada jugador, en el seu torn, «dispara» cap a la flota del seu oponent, indicant una posició (les coordenades d'una casella), la que registra en el segon tauler.



*Si eixa posició està ocupada per part d'un vaixell contrari, l'oponent cantarà **Tocat!** si encara queden parts del vaixell (caselles) sense danyar, o **Afonat!** si amb aquest tret el vaixell ha quedat totalment destruït.*

*Si la posició indicada no correspon a cap part d'un vaixell, cantarà **Aigua!**.*

Guanya el jugador que destrueixca primer tots els vaixells del seu oponent.

[https://es.wikipedia.org/wiki/Batalla_naval_\(juego\)](https://es.wikipedia.org/wiki/Batalla_naval_(juego))



2.- Descripció del Projecte.

- En la nostra versió simplificada només hi haurà un jugador humà (usuari) l'objectiu del qual serà afonar tots els vaixells de l'ordinador en un número determinat d'intents.
- L'ordinador tindrà un tauler amb vaixells i l'usuari dispararà per a intentar afonar-los.
- Es jugarà sobre un **tauler de 10x10 posicions** que s'enumeraran del **0 al 9** en les **columnes** i de la **A a la J** en les **files**.
- Els **tipus de vaixells** són:

- Llanxa (L)**: ocupa una casella del tauler.
- Vaixell (B)**: ocupa 3 caselles **horitzontals** consecutives del tauler.
- Cuirassat (Z)**: ocupa 4 caselles **horitzontals** consecutives del tauler.
- Portaavions (P)**: ocupa 5 caselles **verticals** consecutives del tauler.

- El videojoc tindrà quatre **nivells de dificultat** :

- Fàcil**: L'ordinador col·locarà 10 vaixells (5 llanxes, 3 vaixells, 1 cuirassat i 1 portaavions) en el tauler i el jugador tindrà 50 intents per a afonar-los tots.
- Mitjà**: L'ordinador col·locarà 5 vaixells (2 llanxes, 1 vaixell, 1 cuirassat i 1 portaavions) en el tauler i el jugador tindrà 30 intents per a afonar-los tots.
- Difícil**: L'ordinador col·locarà 2 vaixells (1 llanxa i 1 vaixell) en el tauler i el jugador tindrà 10 intents per a afonar-los tots.
- Personalitzat**: Se li preguntarà a l'usuari la grandària del tauler, el nombre de vaixells de cada tipus i el nombre d'intents.

- A l'inici de cada joc se li preguntarà al jugador en quin **nivell de dificultat vol jugar** i una vegada seleccionat l'ordinador col·locarà **aleatòriament** els vaixells en el tauler (al principi aquest estarà ocult a l'usuari).
- Els vaixells del tauler podran tocar-se però **cal assegurar-se que no se solapen**.



3.- El Tauler de Joc.

La nomenclatura del **tauler** serà la següent:

- Llanxa: **L**
- Vaixell: **B**
- Cuirassat: **Z**
- Portaavions: **P**
- Aigua: **A**
- Posició no disparada: -
- Posició tocada o afonada: **X**

Un exemple de tauler **de l'ordinador (ocult)** en el nivell fàcil podria ser aquest:

	0	1	2	3	4	5	6	7	8	9
A	-	-	-	-	-	-	-	B	B	B
B	-	-	B	B	B	-	-	-	-	-
C	-	-	-	-	-	-	P	-	-	-
D	-	-	L	-	-	-	P	-	-	-
E	-	-	-	-	-	-	P	-	-	L
F	-	B	B	B	-	-	P	-	-	-
G	Z	Z	Z	Z	L	-	P	-	-	-
H	-	-	-	L	-	-	-	-	-	-
I	-	-	-	-	-	-	-	-	-	-
J	-	-	-	-	-	-	L	-	-	-

Al principi el **tauler visible al jugador** mostrarà totes la seua posicions a - (no disparat), així:

	0	1	2	3	4	5	6	7	8	9
A	-	-	-	-	-	-	-	-	-	-
B	-	-	-	-	-	-	-	-	-	-
C	-	-	-	-	-	-	-	-	-	-
D	-	-	-	-	-	-	-	-	-	-
E	-	-	-	-	-	-	-	-	-	-
F	-	-	-	-	-	-	-	-	-	-
G	-	-	-	-	-	-	-	-	-	-
H	-	-	-	-	-	-	-	-	-	-
I	-	-	-	-	-	-	-	-	-	-
J	-	-	-	-	-	-	-	-	-	-



3.- Funcionament del Joc.

- Per a la col·locació dels vaixells s'utilitzaran números aleatoris, recorda que el més senzill és utilitzar el mètode **Math.random()**
- Una vegada l'ordinador ha creat el seu tauler, es mostrarà el tauler visible al jugador i se li preguntarà a quina posició (fila i columna) vol disparar.
- Si en eixa posició hi ha una part del vaixell, es mostrarà el missatge **“Tocat”**. En el cas que no hi haja cap vaixell en aqueixa posició, es mostrarà el missatge **“Aigua”**.
- En tots dos casos es mostrarà el tauler visible al jugador actualitzat amb el tret realitzat. Cal tindre en compte que en la posició disparada només es mostrarà una **A** (si era aigua) o **X** (si un vaixell ha sigut tocat). No s'haurà d'indicar el tipus de vaixell. Per tant, el tauler visible al jugador només contindrà caselles amb -, **A** o **X**.
- L'usuari continuarà realitzant trets fins que afone tots els vaixells, i en aquest cas li apareixerà el missatge de **“Has guanyat!”**, o bé, si el jugador no ha aconseguit afonar tots els vaixells en els intents que tenia, amb el que li semblarà el missatge de **“Has perdut!”**.
- En tots dos casos es mostrarà el tauler ocult de l'ordinador i el joc finalitzarà.
- Durant el joc caldrà manejar les possibles **entrades errònies** (per exemple si l'usuari introdueix una coordenada de tret incorrecta).



4.- Us de Funcions.

- Un aspecte important d'aquest treball són les funcions, que us permetran crear un codi més estructurat, menys repetitiu i senzill d'entendre.
- Haureu de dissenyar i crear les funcions que necessiteu per a cadascuna de les característiques o funcionalitats del joc. A baix teniu la descripció d'algunes **funcions recomanades a implementar**:

- **crearTauler**. Crea el tauler de l'ordinador amb tots els vaixells necessaris.
- **inserixVaixell**. Inserida un vaixell en el tauler.
- **tret**. Controla el tret de l'usuari i actualitza el tauler.
- **mostrarTauler**. Mostra per pantalla el tauler a l'usuari.

- Segurament us és d'ajuda crear algunes funcions més. Per exemple, podrien ser útil crear *triar_dificultat*, *numero_aleatori*, *inserir_llanxa*, *comprovar_???*, etc.

És obligatori escriure a dalt de cada funció un comentari explicant què fa.

5.- Recomanacions.

- **Una bona pràctica és que les funcions no tinguin més de 30-40 línies de codi**, inclosa la funció **main()**. No és obligatori, és un consell. Si una funció té més línies tal vegada és convenient posar part d'aqueix codi en una funció.
- **Abans de començar** a programar llig bé l'enunciat, subratlla l'important, fes-te notes, escriu un esborrany de les possibles funcions, etc. Intenta entendre el problema i com realitzar-lo abans de començar a escriure codi.
- **No intentes fer-ho tot de colp**. Una estratègia clau en programació és dividir un problema gran en diversos problemes xicotets i després resoldre'ls un a un (és a dir, programa i prova les funcions una a una).
- **Comença fent el funcionament bàsic** descrit en l'apartat 6 i a poc a poc anar afegint els funcionalitats. És molt aconsellable fer **còpies de seguretat** del codi després de cada etapa.



5.- Desenvolupament per etapes i criteris de qualificació.

- 🎯 **Videojoc Bàsic (5 punts):** Es pot jugar però només hi ha 10 llanxes (no hi ha altres tipus de vaixells) i sense nivells de dificultat (el joc acaba després de 50 trets).
- 🎯 **Afegir tots els vaixells (2 punts):** Afegir tots els tipus de vaixells (Llanxa, Vaixell, Cuirassat i Portaavions). Segueixen sense haver-hi nivells de dificultat, es juga en nivell fàcil.
- 🎯 **Afegir 3 nivells de dificultat (2 punts):** Afegir l'opció de triar el nivell de dificultat (només fàcil, mitjà i difícil).
- 🎯 **Afegir joc personalitzat (1 punt):** S'afegirà l'opció de joc personalitzat.

En cada etapa desenvolupada es valorarà:

- Correcte funcionament del videojoc.
- Correcte ús de funcions i evitar repetir codi (en la mesura que siga possible).
- Codi ordenat i ben estructurat.
- Noms de variables apropiats i autoexplicatives.
- Comentaris útils i breus que ajuden a entendre el codi.