

# Bataille navale en réseau

Le fichier bataille navale qui vous est fourni vous permet de jouer seul, à partir d'un terminal.

Le jeu fonctionne ainsi :

- phase d'initialisation :
  - le plateau de jeu est créé ;
  - les bateaux sont générés et placés aléatoirement ;
- phase d'itération :
  - le plateau est affiché
  - on demande à l'utilisateur de saisir un coup ;
    - si le coup a déjà été joué, on lui demande une nouvelle saisie ;
  - on joue le coup :
    - la case précédemment masquée devient occupée ou est marquée comme vide ;
    - si tous les navires sont découverts, la partie est gagnée : phase de conclusion
    - s'il reste encore une case occupée par un navire à découvrir, on refait une itération ;
- phase de conclusion :
  - on demande au joueur s'il veut rejouer :
    - si oui, on refait une phase d'initialisation et on recommence les itérations
    - si non, on quitte le programme.

**Phase 1 :** L'idée principale du projet est de rendre le jeu jouable en réseau.

On aurait donc un client qui aurait les caractéristiques suivantes :

- serait utilisé uniquement pour l'affichage et la saisie ;
- ne connaît pas l'état du jeu ni ses règles ;
- valide cependant le format attendu (gère la saisie d'une case et de oui/non).

Dans ce qui vous est livré :

- un package saisie qui contient les fonctions permettant de gérer les saisies
- un module composants qui contient des classes permettant de jouer
- un module jeu qui gère l'algorithme de jeu
- un module lanceur qui permet de lancer le jeu : **\$ python3 lanceur.py.**

Voici la procédure proposée pour jouer en réseau :

- démarrage du serveur :
  - le serveur se lance et affiche IP et port (# 1) ;
- démarrage du client :
  - le client demande l'IP et le port du serveur, le joueur le saisit (cf #1) ;
  - le client se connecte au serveur en lui envoyant un message HELLO :
    - en cas d'échec, s'éteint avec un message d'erreur ;
    - en cas de succès, le serveur lance la procédure d'initialisation (identique).

Itérations :

- le client demande le plateau ;
- le serveur envoie le plateau ;
- le client envoie le plateau et demande la saisie d'une case ;
- le joueur saisit une case ;
- le client envoie la case au serveur ;
- le serveur répond « déjà joué », « touché », « coulé » ou « manqué ».
- le client demande s'il a gagné ;
- le serveur répond oui ou non :
  - si non, le client refait une itération ;
  - si oui, le client demande si le joueur veut faire une nouvelle partie :
    - si oui, le client lance à nouveau un HELLO au serveur ;
    - si non, le client quitte.

**Phase 2 :** Rajouter une option permettant de compter le score à la fin (score = nombre de cases non encore découvertes). Cela vous obligera à rentrer *un peu* dans le code du jeu, mais ce score se calcule en une simple soustraction, ne cherchez pas trop loin.

A la fin de chaque partie : permettre au joueur d'entrer son nom et enregistrer le nombre score courant.

A chaque début et à chaque fin de partie (après que le joueur ait entré son score) afficher le TOP 10 des scores (s'il n'y a pas encore 10 scores, afficher uniquement les premiers, classés par ordre du plus gros au plus petit).

Phase 3 : Utiliser le server multi-threadé pour permettre à deux joueurs de jouer ensemble, au tour par tour. Afficher les deux tableaux cote à cote chez chaque client.

