

Projet Pierre-Feuille-Ciseau

⚠ principe du projet

- **LA NOTE MAXIMALE DE CE PROJET EST DE 13 SANS LA PARTIE FACULTATIVE !**
- Vous travaillerez par binomes sur ce projet (ou exceptionnellement par trinome).
- Vous rendrez le code complet + votre dossier personnel pour le 09 novembre 2022 au plus tard.
- Vous aurez un temps en classe pour réaliser le projet, mais ce temps ne sera pas suffisant ! Vous devrez vous coordonner pour arriver à vos fins !

1. Description du projet

L'objectif est d'implémenter un jeu de « Pierre - Feuille - Ciseaux » entre un joueur/une joueuse humaine et l'ordinateur.

Le joueur/la joueuse choisit parmi les trois possibilités, puis l'ordinateur choisit au hasard une des trois possibilités.

Le gagnant gagne 1 point, si il y a égalité aucun ne marque.


Le jeu se termine dès qu'un des participant·e·s atteint 5 points.

2. Plan de travail

Le jeu étant *très simple* à créer, le plan de travail est moins avancé que pour les projets `Pendu` ou `Bandit Manchot`. Il faudra cependant créer les assertions et/ou les tests nécessaires pour chacune des fonctions suivantes.

Il faudra créer :

- une fonction `choix_joueur`, qui renvoie le choix du joueur / de la joueuse ;
- une fonction `choix_ordinateur`, qui renvoie le choix de l'ordinateur ;
- une fonction `qui_gagne`, qui renvoie l'indice du joueur/ de la joueuse gagnant·e, selon les deux arguments passés. Par exemple `qui_gagne('Feuille', 'Ciseaux')` renvoie `1`, alors que `qui_gagne('Pierre', 'Ciseaux')` renvoie `0` ;
- une fonction `une_manche` qui gère une manche de jeu ;
- une fonction `main_game` qui gère une partie complète.

 **Partie facultative**

Il est possible de complexifier le jeu avec 5 possibilités, comme dans le jeu de Pierre - Feuille - Ciseaux - Lezard - Spock , présenté [ici](#) par Sheldon Cooper.

Pour les allergiques à l'audio en anglais, le voici en explication [texte](#)

3. Grille de notation

intitulé	barème	Détails
fonction <code>choix_joueur</code>	2 pts	Les assertions sont explicites
fonction <code>choix_ordinateur</code>	2 pts	Les assertions sont explicites
fonction <code>qui_gagne</code>	2 pts	Des tests exhaustifs sont écrits
fonction <code>une_manche</code>	1 pt	L'ensemble est cohérent
fonction <code>main_game</code>	1 pt	Une partie est jouable
Code clair et commenté	2 pts	
Cohérence des choix de noms de variables	2 pts	
Esthétique du jeu	1 pt	
RCPLS	2pts	Jeu selon les règlmes TBBT