|  |  |  |
| --- | --- | --- |
| Numérique et Sciences Informatiques | | |
| 4h | **Projet : mastermind** |  |
| Objectif : boucles, double boucles, listes, tuple, fonctions, variables locales et globales. | | |
| **Matériel :** Python | | |

Le dossier (nom : **NOM1\_NOM2\_NOM3)** du projet est à déposer avant **lundi 12 décembre 2022 à minuit** sur : https://tinyurl.com/projetsNSI

Le dossier sera constitué :

* du programme python.
* du rapport.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Groupe 1 :** | **Groupe 2 :** | **Groupe 3 :** | **Groupe 4 :** | **Groupe 5 :** | **Groupe 6 :** |
| Master\_dur()  Difficulté : \*  Master\_faible()  Difficulté : \*\* |  |  |  |  |  |  |
| joueur()  Difficulté : \*\* |  |  |  |  |  |  |
| bien\_et\_mal\_placees()  Difficulté : \*\*\* |  |  |  |  |  |  |

* Réalisez le jeu du **mastermind** grâce aux fonctions définies.
* Vous pouvez utiliser replit.com en ligne pour travailler sur le même code en même temps.
* Chaque groupe réalisera un rapport technique qui sera évalué (évaluation commune du code + évaluation individuelle du rapport).

Critères d'évaluation :

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Programme commun :** | | | **Rapport technique élève 1 :** | | | | **Retard :** |
| Qualité du code | Respect du cahier des charges | Qualité des commentaires | Partie 1 | Partie 2 | Partie 3 | Partie 4 |  |
| 3 points | 1 point | 1 point | 1 point | 1 point | 1.5 point | 1.5 point | - 1 point |

**Cahier des charges**

L’ordinateur choisit une série de 4 couleurs au hasard (parmi 6 couleurs possibles 'A', 'B', 'C', 'D', 'E', 'F') puis le joueur doit deviner la série de 4 couleurs qu'on appellera le **Master**.

Pour aider le joueur, après chaque essai, on affiche le nombre de couleurs **bien placées** et le nombre de couleurs **mal placées**.

La partie s'arrête lorsque le joueur aura trouvé le **Master**.

L'ordinateur indiquera le nombre d'essais.

Le programme permettra au joueur de faire 12 essais maximum.

Note : on peu se dire que 'A' est la couleur rouge, 'B' la couleur marron, 'C' la couleur orange ...

**Exemples**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **Master** à trouver | | | |  |  |
|  |  | **A** | **B** | **C** | **C** |  |  |
| **Joueur** | 1er essai : | B | B | D | A | Bien placées : 1 | Mal placées : 1 |
| 2ème essai : | C | A | B | B | Bien placées : 0 | Mal placées : 3 |
| 3ème essai : | C | B | B | C | Bien placées : 2 | Mal placées : 1 |
| 4ème essai : | A | B | B | C | Bien placées : 3 | Mal placées : 0 |

**Testez le jeu du en ligne : https://www.ordi-netfr.com/mastermind/jeux\_mastermind.php**

**Fonctions et variables utiles pour programmer le mastermind**

La fonction master\_dur(L\_couleurs) crée et renvoie une liste de 4 caractères correspondant aux 4 couleurs du **master** choisies au hasard parmi les 6 couleurs de la L\_couleurs.

Attention, une couleur peut être présente plusieurs fois dans ce master.

La fonction master\_facile(L\_couleurs) crée et renvoie une liste de 4 caractères correspondant aux 4 couleurs du **master** choisies au hasard parmi les 6 couleurs de la L\_couleurs.

Attention, toutes les couleurs de ce master seront différentes les unes des autres.

La fonction joueur(L\_couleurs) invite le **joueur** à faire une proposition et renvoie une liste de 4 caractères correspondant aux 4 couleurs proposées par le **joueur**.

Si le joueur saisit les couleurs en minuscule la fonction devra les transformer en majuscule.

Vous n'utiliserez pas la méthode upper().

La fonction vérifiera que la couleur rentrée par le joueur appartient bien à la L\_couleurs, sinon une autre saisie sera demandée.

La fonction bien\_et\_mal\_placees() renvoie deux arguments :

* Un entier correspondant au nombre de couleurs, proposées par le **joueur**, bien placées.
* Un entier correspondant au nombre de couleurs, proposées par le **joueur**, mal placées.

Attention, si une couleur est déjà bien placée et qu’elle apparaît une seconde fois dans la proposition du **joueur** il ne faudra pas la compter mal placée.

Attention : Si vous transformer la liste L\_master dans votre fonction, le joueur ne pourra faire qu'un essai → pensez à faire une **copie de liste**.

|  |  |  |
| --- | --- | --- |
| **Nom des fonctions** | **Variable(s) en argument : entrée(s)** | **Variable(s) renvoyée : sortie(s)** |
| Master\_dur()  Master\_facile() | L\_couleurs | L\_master |
| joueur() | L\_couleurs | L\_joueur |
| bien\_et\_mal\_placees() | L\_master  L\_joueur | nb\_bien\_placees  nb\_mal\_placees |
| **Plus besoin de rajouter \_loc sur les variables locales aux fonctions** | | |

|  |  |
| --- | --- |
| **Programme principal :** | |
| **Variables** | **Description** |
| L\_couleurs | Liste de 7 caractères représentants les 6 couleurs du mastermind. ['A','B','C','D','E','F'] |
| L\_master | Liste de 4 caractères représentants les 4 couleurs du mastermind à trouver. |
| L\_joueur | Liste de 4 caractères représentants les 4 couleurs du mastermind que le joueur propose à chaque essai. |
| nb\_bien\_placees | Variable contenant le nombre de couleurs bien placées. |
| nb\_mal\_placees | Variable contenant le nombre de couleurs mal placées (sans compter les couleurs bien placées). |
| nb\_essais | Variable contenant le nombre d'essais. |

**Contenu du rapport**

**Un rapport technique par groupe → chaque élève fera le travail suivant dans le rapport :**

**Partie 1**

Donnez les pré-conditions et les post-conditions de votre fonction.

**Partie 2**

Décrivez, sous forme de tableau, les tests que vous avez utilisés pour vérifier que votre fonction fonctionne correctement dans tous les cas.

**Partie 3**

Décrivez au moins un bug que vous avez rencontré (pas des erreurs de programmation) au cours du développement de votre fonction.

Si vous n'avez pas rencontré de bug, créez-en un et décrivez-le.

**Donnez le code avec le bug (copie d'écran) et expliquez ce qu'il se passe et comment vous avez réussi à le résoudre si c'est le cas.**

**Partie 4**

Concluez sur ce projet (ce qui vous a plu dans ce projet, vos difficultés, vos réussites).

**5 lignes minimum par élève**, la qualité de l'expression sera prise en compte dans l'évaluation.