Choix des variables

Constantes

Des constantes permettent de paramétrer le jeu :

- COULEURS = [“J”,“B”,“R”,“V”,“O”,“N”]
- NB_PIONS = 4
- NB_MAX_ESSAIS = 10
- AVEC_REPETITION = True
- COMPTAGE_GLOBAL = False
- SYMBOLE_BIEN_PLACE = “#”
- SYMBOLE_MAL_PLACE = “X”
- SYMBOLE_ABSENT = “0”

Variables du jeu

- la combinaison à deviner est une liste de chaînes de caractères
- la proposition de combinaison est également une liste de chaînes de caractères
- l’évaluation d’une proposition est rendue sous la forme d’une liste de symboles, par exemple “[‘#’, ‘X’, ‘X’, ‘0’]” si la combinaison contient un “bien placé”, deux “mal placés” puis un “non présent”.

Découpage des fonctions

Fonctions utilitaires

- combi_to_str(combinaison : list) -> str
- nb_occurrences(element : str, combinaison : list) -> int
- trouve(element : str, combinaison : list) -> int

Fonctions propres au projet

- genere_combinaison_secrete() -> list
- saisir_combinaison() -> list
- evalue_proposition(proposition : list, secret : list) -> list
  Compare la proposition avec la combinaison secrète pour évaluer les pions bien placés et les mal placés dans la proposition.
  Retourne une liste de symboles représentant “bien placé”, “mal placé”, ou “pas présent”.
- manche_de_mastermind() -> int
  Jeu complet d’une manche de Mastermind à un joueur contre l’ordinateur.
  - on génère la combinaison avec genere_combinaison_secrete()
  - le joueur saisit sa proposition avec saisir_combinaison()
  - la combinaison est évaluée avec evalue_proposition(proposition : list, secret : list) -> list
  - et on répète le jeu jusqu’à ce que le joueur gagne (combinaison correcte) ou perde (nombre maximal de propositions atteint)
- on pourra prévoir un menu pour paramétrer la difficulté et choisir de rejouer ou non.