

Partie 1 - Tri à bulles (8 points)

« Le **tri à bulles** ou **tri par propagation** est un algorithme de tri. Il consiste à comparer répétitivement les éléments consécutifs d'un tableau, et à les permuter lorsqu'ils sont mal triés. Il doit son nom au fait qu'il déplace rapidement les plus grands éléments en fin de tableau, comme des bulles d'air qui remonteraient rapidement à la surface d'un liquide.

Le tri à bulles est souvent enseigné en tant qu'exemple algorithmique, car son principe est simple. Mais c'est le plus lent des algorithmes de tri communément enseignés, et il n'est donc guère utilisé en pratique. »

Extrait de Wikipédia FR, fin octobre 2019.

Les différents exercices de cette partie partent du principe que nous disposons d'un tableau (liste python) déjà rempli de valeurs, nommé `myTable`, d'une taille quelconque.

1/ Nous allons avoir besoin de **permuter** deux valeurs d'un tableau à partir de leurs indices. Écrire un programme permettant de **permuter** deux valeurs du tableau `myTable`.

2/ Lors d'**une** itération, l'algorithme du tri à bulles parcourt le tableau, et compare les éléments consécutifs. Lorsque deux éléments consécutifs ne sont pas dans l'ordre, ils sont échangés. Par conséquent, à l'issue d'une itération (et donc, d'un parcours du tableau), le plus grand élément est systématiquement déplacé en fin de tableau ; comme s'il s'agissait d'une bulle qui remonte à la surface.

Écrire un programme permettant le **parcours** du tableau au cours d'une itération du tri à bulles. On pourra se servir de la **permutation** définie dans l'exercice précédent.

3/ Écrire un programme implémentant le tri à bulles complet, permettant de trier totalement un tableau grâce à l'algorithme du tri à bulles. On pourra se servir de la **permutation**, ainsi que de la réponse à la question précédente.

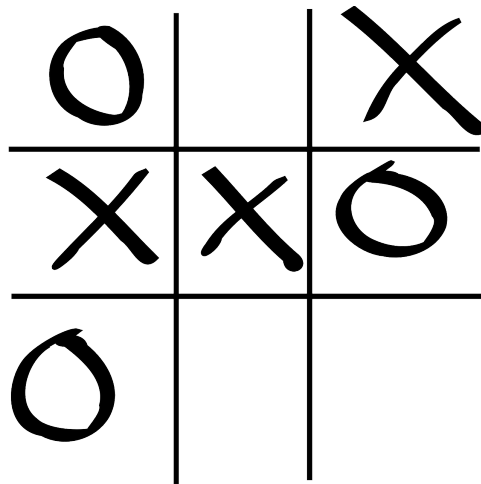
4/ Le tri à bulles est considéré comme très lent. Pourquoi ? Peut-on avoir une idée du temps nécessaire à son exécution, estimer son ordre de grandeur ? Détaillez votre réponse en commentaire de votre programme.

Partie 2 - Tic tac toe (12 points)

Règles du jeu

« Deux joueurs s'affrontent. Ils doivent remplir chacun à leur tour une case de la grille avec le symbole qui leur est attribué : O ou X. Le gagnant est celui qui arrive à aligner trois symboles identiques, horizontalement, verticalement ou en diagonale. Il est coutume de laisser le joueur jouant X effectuer le premier coup de la partie. »

Extrait de Wikipédia FR, fin octobre 2020.



1. Écrire la fonctionnalité permettant d'afficher la grille de jeu.
2. Écrire la fonctionnalité permettant de jouer un O ou un X.
3. Écrire la fonctionnalité vérifiant si oui ou non l'un des joueurs a réussi à aligner 3 symboles sur une ligne verticale, horizontale, diagonale.
4. Écrire la fonctionnalité vérifiant si la grille est complète.
5. Écrire un programme permettant de jouer à deux au Tic tac toe.
6. Qu'aura-t-on besoin de faire, si on souhaite désormais programmer un jeu de Puissance 4 ?
Répondez en commentaire.

Nomenclature repository git : JV1[A/B]_ExamBasesAlgo_NOMPRENOM