

BASES DE L'ALGORITHMIQUE

EXAMEN TP

20 MAI 2022

Modalités :

- Durée : 1h30
- L'accès à l'espace Bases de l'algorithmique sur Community est autorisé.
- **Toute autre utilisation d'internet et du mail est interdite.**

Exercice 1 (9 points)

1. (3 points) Écrivez une fonction qui prend en paramètre un entier n et qui génère et retourne une permutation aléatoire de l'ensemble des entiers de 1 à n , c'est à dire une liste contenant l'ensemble de ces entiers, chacun apparaissant une seule fois, et dont les positions sont aléatoires.

Par exemple, si $n = 10$ une permutation aléatoire de l'ensemble des entiers de 1 à 10 peut être la suivante : $[6, 5, 10, 1, 2, 4, 9, 7, 3, 8]$.

Important : Si une fonction Python permet de générer directement une permutation aléatoire vous ne devez pas l'utiliser. Vous devez tirer vos valeurs aléatoires une par une.

Bonus : Un bonus de 1 point sera accordé si votre façon de générer la permutation ne nécessite que n tirages aléatoires.

2. (3 points) Écrivez une fonction qui prend une liste d'entiers en paramètre et qui retourne la longueur de la plus longue sous-séquence croissante incluse dans cette liste.

Par exemple, la plus longue sous-séquence croissante de la liste :

$[16, 5, 10, 31, 2, 4, 19, 27, 3, 18]$ est $[2, 4, 19, 27]$ et est donc de longueur 4.

3. (3 points) Écrivez un programme qui utilise les fonctions précédentes pour générer deux permutations aléatoires de même taille, qui affiche la longueur de la plus longue sous-séquence croissante de chacune, et qui calcule et affiche le nombre de positions collisionnantes, c'est à dire pour lesquelles la valeur est la même dans les deux permutations.

Par exemple, si les permutations aléatoires sont $[6, 5, \mathbf{7}, 1, 2, \mathbf{4}, \mathbf{3}, 8]$ et $[5, 1, \mathbf{7}, 6, 8, \mathbf{4}, \mathbf{3}, 2]$, le nombre de positions collisionnantes est 3.

Exercice 2 (11 points)

Vous allez programmer un jeu de cartes appelé *Blackjack* dont les règles sont les suivantes :

Dans (une version simple de) ce jeu, les couleurs (pique, cœur, carreau, trèfle) n'ont pas d'importance. Seules les valeurs faciales des cartes importent. Les cartes de 2 à 10 ont pour valeur ce nombre. Les cartes valet, dame et roi valent chacune 10. L'as vaut soit 1, soit 11, en fonction de ce qui est le plus avantageux pour le joueur.

Le joueur tire 2 cartes dans un jeu classique de 52 cartes. Le but du joueur est d'avoir un total des valeurs des cartes qui est le plus grand possible sans toutefois dépasser 21. Si le total des cartes dépasse 21, la partie est terminée et le joueur gagne 0 point.

À tout moment le joueur peut décider de tirer une carte supplémentaire, ou de terminer la partie avec les cartes qu'il possède. Il gagne alors le nombre de points total de ses cartes.

Exemple 1 : Le joueur tire deux cartes qui sont 3 de carreau et as de pique. Il décide de s'arrêter et gagne 14 points.

Exemple 2 : Le joueur tire deux cartes qui sont 3 de carreau et as de pique. Il décide de tirer une autre carte qui est un 8 de pique. Cela lui fait 12 points (car si on comptait l'as à 11 on aurait 22 qui dépasse 21). Il décide de s'arrêter et gagne 12 points.

Exemple 3 : Le joueur tire deux cartes qui sont 3 de carreau et as de pique. Il décide de tirer une autre carte qui est un 8 de pique. Cela lui fait 12 points (car si on comptait l'as à 11 on aurait 22 qui dépasse 21). Il décide de tirer une autre carte qui est un 5 de trèfle. Cela lui fait 17 points. Il décide de s'arrêter et gagne 17 points.

Exemple 4 : Le joueur tire deux cartes qui sont 3 de carreau et as de pique. Il décide de tirer une autre carte qui est un 8 de pique. Cela lui fait 12 points (car si on comptait l'as à 11 on aurait 22 qui dépasse 21). Il décide de tirer une autre carte qui est un 5 de trèfle. Cela lui fait 17 points. Il décide

de tirer une autre carte qui est une dame de carreau. Cela fait un total de 27 qui est supérieur à 21. La partie s'arrête donc et le joueur gagne 0 point.

Vous représenterez chaque carte comme une liste de deux chaînes de caractères, l'une étant la valeur faciale de la carte, l'autre la couleur. Par exemple une carte pourra valoir ["valet", "carreau"], une autre ["3", "coeur"].

Le main courante du joueur sera représentée par une liste de cartes. Par exemple : [["valet", "carreau"], ["3", "coeur"]].

1. (3 points) Écrivez une fonction qui retourne une carte (liste de taille 2) dont la valeur faciale et la couleur sont aléatoires.
2. (4 points) Écrivez une fonction qui prend en paramètre une main (liste des cartes du joueur) et qui retourne le nombre de points correspondant à cette main. Si un ou plusieurs as sont présents dans la main alors vous les compterez chacun à 1 ou 11 selon ce qui est le plus avantageux pour le joueur (il ne faut pas dépasser 21).
3. (4 points) Écrivez un programme qui simule une partie complète. Initialement le joueur se voit attribuer deux cartes. Puis répétitivement, le joueur doit décider s'il est satisfait et arrête la partie, ou si il souhaite tirer une carte supplémentaire. À chaque tour de la partie vous devez afficher la main ainsi que le nombre de points qu'elle représente.

La partie s'arrête, soit d'autorité lors la main est supérieure à 21, soit par la décision du joueur. Lorsque la partie est finie vous devez afficher le nombre de points gagnés par le joueur.