

Examen du module d'ingénierie informatique

Durée : 2 heures

Documents autorisés : aucun

Exercice 1 (3 points) — Question de cours

On considère la fonction suivante :

```
def fonction (tab) :  
    somme = 0  
  
    y = 0  
    x = 0  
    while y < 3:  
        while x < 3:  
            somme = somme + tab[y,x]  
            x = x + 1  
        y = y + 1  
  
    return somme
```

tab =

2	3	4
1	3	7
1	2	1

Indiquez ce que cette fonction retournera si on lui passe en paramètre le tableau numpy `tab` indiqué ci-dessus.

Exercice 2 (3 points) — Question de cours

On considère les fonctions suivantes :

```
def initialise (tab) :  
    tab = np.full((3,3), 0)  
  
def rempli_inc (tab) :  
    initialise (tab)  
    x = 0  
    while x < 3:  
        tab[0,x] = x  
        x = x + 1
```

tab =

2	2	2
2	2	2
2	2	2

Après avoir exécuté la fonction `rempli_inc` en lui passant en paramètre le tableau numpy `tab` indiqué ci-dessus, quel est le contenu de ce dernier.

Exercice 3 (3 points) — Recherche dans un tableau

Écrivez une fonction python `recherche` prenant en paramètres un tableau 1D numpy `tab` ainsi qu'un entier `tab.length` représentant la taille de ce tableau. On supposera que le tableau `tab` contient uniquement des entiers positifs ou nuls. Votre fonction `recherche` renverra l'index **le plus petit** des cellules de `tab` dont la valeur est 0, si de telles cellules existent, sinon elle renverra la valeur `-1`. Par exemple, si `tab = [2 3 0 4 0 0 6]`, la fonction renverra 2, tandis que si `tab = [2 3 1 4]`, la fonction renverra `-1`.

Exercice 4 (3 points) — Contenu d'un tableau

Q 4.1 Indiquez ce qu'affiche l'instruction `print` du programme python ci-dessous (ne donnez pas d'explications, indiquez juste l'affichage).

```
import numpy as np

def myfunc (tab, tab_length):
    """tab_length = longueur du tableau"""
    i = 0
    while i < tab_length // 2:
        t = tab[i]
        tab[i] = tab[tab_length-i-1]
        tab[tab_length-i-1] = t
        i = i+1
```

```
tableau = np.full(5,0)
i = 0
while i < 5:
    tableau[i] = i
    i += 1

myfunc (tableau,5)
print (tableau)
```

Q 4.2 Qu'afficherait le programme si la boucle `while` de `myfunc` allait jusqu'à `tab_length` au lieu de `tab_length // 2`.

Exercice 5 (3 points) — Transposée

Écrivez une fonction qui prend en paramètres un tableau numpy carré ainsi que la taille (nombre de lignes = nombre de colonnes) du tableau, et qui transpose ce dernier. Par exemple, l'application de la fonction au tableau

1	2	3
4	5	6
7	8	9

transformera ce dernier en

1	4	7
2	5	8
3	6	9

. Bien entendu, il est interdit d'utiliser la fonction `transpose` de numpy.

Exercice 6 (3 points) — Déplacement

Écrivez une fonction `direction` qui, étant donné les coordonnées `x_souris`, `y_souris`, `x_bonbon`, `y_bonbon`, exprimées en lignes/colonnes du `tab_jeu` de candy crush, renvoie la direction dans laquelle le bonbon a été déplacé (`DIRECTION_DROITE`, `DIRECTION_GAUCHE`, `DIRECTION_HAUT`, `DIRECTION_BAS`) si le bonbon a été déplacé à la souris horizontalement ou verticalement et si elle se trouve dans l'espace de jeu (on rappelle que `tab_jeu` est un tableau de `NB_LIGNES_JEU` lignes et `NB_COLONNES_JEU` colonnes). Dans le cas contraire, la fonction renverra `DIRECTION_AUCUNE`.

Exercice 7 (2 points) — Dernière sommation

Écrivez une fonction qui prend en argument un nombre réel `x` ainsi qu'un entier `n` dont on supposera qu'il est strictement positif, et qui renvoie la valeur de l'expression :

$$\sum_{i=1}^n i \times x^i.$$

Attention : il est interdit d'utiliser d'autres opérateurs arithmétiques que les additions (+) et multiplications (*).