

DS01

Algorithmique et programmation

Exercice 1 : Architecture matérielle et logicielle

Q 1 : Rappeler les trois fonctions qui permettent de définir un ordinateur.

Q 2 : Parmi les propositions suivantes, décrire quelles sont celles présentes dans l'architecture de Von Neumann

- (a) Mémoire virtuelle;
- (b) Processus;
- (c) Processeur;
- (d) Canal de communication;
- (e) Mémoire vive;
- (f) Mémoire de masse.

Q 3 : Donner la signification du mot OS. Décrire brièvement à quoi ça sert et donner 3 exemples.

Exercice 2 : Fonction mystère

On donne la fonction `fonctionMystere(n)` définie comme suit.

■ Python

```
def fonctionMystere(n) :  
    if n==0 or n==1:  
        return 1  
    else :  
        res = 1  
        for i in range (2,n+1) :  
            res = res * i  
        return res
```

Q 4 : Si $n = 5$ quelles sont les valeurs que va prendre la variable i ?

Q 5 : Si $n = 4$ donner les valeurs successives que vont prendre les variables i et res lorsqu'on exécute l'algorithme.

Q 6 : Quel est le nom mathématique usuel donné à la fonction `fonctionMystere` ?

Exercice 3 : Evolution des variables

■ Python

```
def fonction(x) :
    y=1.1
    i=0
    while x!=y and i <10:
        x=x+0.1
        i=i+1
    return i
```

Q 7 : On exécute l'instruction `fonction(0.1)`. À chaque itération, donner la valeur de *i* et de *x*.

Q 8 : On exécute l'instruction `fonction(0.3)`. À chaque itération, donner la valeur de *i* et de *x*.

Exercice 4 : Manipulation de listes

Q 9 : Écrire une fonction `ajouteUnFor(L)` qui prend comme argument une liste *L* de flottants et qui ajoute 1 à chaque élément de la liste. On utilisera une boucle `for`.

Q 10 : Écrire une fonction `ajouteUnWhile(L)` qui prend comme argument une liste *L* de flottants et qui ajoute 1 à chaque élément de la liste. On utilisera une boucle `while`.

Q 11 : Expliquer pourquoi il n'est pas indispensable que la fonction renvoie la liste modifiée.

Exercice 5 : Maximum de listes

Pour les deux questions suivantes, les fonctions `max` et `min` ne sont pas autorisées.

Q 12 : Écrire une fonction `chercheMax(L)` qui prend comme argument une liste *L* d'entiers `int` et qui renvoie le plus grand élément de la liste.

Q 13 : Écrire une fonction `chercheMaxIndice(L)` qui prend comme argument une liste *L* d'entiers `int` et qui renvoie l'indice du plus grand élément de la liste.