

DS01

Algorithmique et programmation

Proposition de corrigé

Exercice 1 : Architecture matérielle et logicielle

Q 1 :

- Universelle;
- Servant à traiter de l'information;
- Programmable.

Q 2 :

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

Q 3 :

Un OS est un système d'exploitation (Operating System en Anglais). Ce programme a pour but de gérer les accès aux processeurs et à la mémoire et aux périphériques, ce qui permet effectivement que l'ordinateur exécute plusieurs programmes à la fois.

Par ailleurs l'OS permet de manière générale de gérer l'organisation des données sur le disque dur ainsi que leur droit d'accès. Il gère aussi les différentes ressources et sert de garde-fou en cas de tentative de mauvaise utilisation des ressources de l'ordinateur.

Pour finir, l'OS permet à un ou plusieurs utilisateurs de s'identifier leur permettant ainsi d'utiliser un seul ordinateur sans nécessairement partager les données et les programmes.

Exercice 2 : Fonction mystère

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

i va prendre les valeurs 2, 3, 4, 5.

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

$i=2$ $res=2$

$i=3$ $res=6$

$i=4$ $res=24$

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

`fonctionMystere(n)` retourne la factorielle de n .

Exercice 3 : Evolution des variables

Q 7 :

i=1 x=0.2
i=2 x=0.3
i=3 x=0.4
i=4 x=0.5
i=5 x=0.6
i=6 x=0.7
i=7 x=0.8
i=8 x=0.9
i=9 x=1.0
i=10 x=1.1

Q 8 :

i=1 x=0.4
i=2 x=0.5
i=3 x=0.6
i=4 x=0.7
i=5 x=0.8
i=6 x=0.9
i=7 x=1.0
i=8 x=1.1
i=9 x=1.2
i=10 x=1.3

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`.

```
def ajouteUnFor(L):
    '''prend comme argument une liste L de flottants et ajoute 1
    à chaque élément de la liste avec un boucle for'''
    for i in range(len(L)):
        L[i] += 1
```

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`.

```
def ajouteUnWhile(L):
    '''prend comme argument une liste L de flottants et ajoute 1
    à chaque élément de la liste avec une boucle while'''
    i = 0
    while i < len(L):
        L[i] += 1
        i += 1
```

Q 11 : Expliquer pourquoi il n'est pas indispensable que la fonction renvoie la liste modifiée.
Un liste est un objet mutable ça veut dire qu'elle est modifiable.

Exercice 5 : Maximum de listes

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.

```
def chercheMax(L):
    '''prend comme argument une liste L d'entiers int et
    qui renvoie le plus grand élément de la liste'''
    max = L[0]
    for i in range(1, len(L)):
        if L[i] > max:
            max = L[i]
    return max
```

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 et s'il y en plusieurs renvoie le plus petit.

```
def chercheMaxIndice(L):
    '''prend comme argument une liste L d'entiers int et
    qui renvoie l'indice du plus grand élément de la liste'''
    max = L[0]
    indmax = 0
    for i in range(1, len(L)):
        if L[i] > max:
            max = L[i]
            indmax = i
```

```
return indmax
```