TD 2 MPSI 2020 – 2021

Lycée M@RIETTE
BOULOGNE - SUR - MER

On utilisera un seul fichier Python nommé TD2.py pour l'ensemble du TD

EXERCICE N°1 :

Une entreprise souhaite renouveler ses imprimantes. Le prix unitaire est de 75 €. Celui – ci passe à 70 € dès la deuxième et à 65 € à partir de la sixième.

Par exemple, pour 7 imprimantes, le prix total sera :
p = 75 + 4 x 70 + 2 x 65 = 485 €
l'imprimante 1 est facturée 75€
les imprimantes 2,3,4,5 sont facturées 70€
les imprimantes 6 et 7 sont facturées 65€

On souhaite connaître le prix total en fonction du nombre n d'imprimantes achetées.

1°) Compléter les algorithmes ci – dessous :

prix en fonction de n

si n=1

alors prix = ...

sinon si n . . . . . et n . . . .

alors prix = ...

sinon prix = ...

renvoyer prix

```
def prix(n) :
    if n==1:
        p= . . . . . .
    elif n . . . . .  and  n . . . . . :
        p=. . . . . . . . . . . . . . . . .
    else :
        p = . . . . . . . . . . . . . . . . .
    return p
```

2°) Implanter cet algorithme sous Python et le tester avec n = 1 , n = 4 , n = 8

Formulaire :		
	n=1	n prend la valeur 1
	n==1	pour une condition : n est-il égal à 1 ?
	then	n'existe pas en Python ! !
	elif	else if = sinon si

EXERCICE N°2 :

Formulaire :

L=[]	L est une liste « vide »
L=[2,7,5,13]	L est une liste de 4 nombres
L[0]	renvoie le 1^{er} terme de L : ici 2
L.append(8)	ajoute le nombre 8 en dernière position
L	affiche la liste L (modifiée)
len(L)	renvoie la longueur de L : ici 5
L[4]	renvoie le $5^{ème}$ terme de L : le dernier ! ici 8
L[5]	renvoie erreur ! !

1°) Tester, sur la page d'accueil de Python, les commandes précédentes.

2°) On considère la suite (U_n) : $\begin{cases} U_0 = 3 \\ U_{n+1} = 4\,U_n + 2 \end{cases}$

Ecrire un algorithme nommé listeU(n), qui à un entier n fixé, renvoie la liste L des n premières valeurs de la suite (U_n). Tester avec n = 1 , n = 4 , n = 10

EXERCICE N°3 :

On considère la suite de Fibonacci (F_n) : $\begin{cases} F_0 = 0 \quad F_1 = 1 \\ F_{n+2} = F_{n+1} + F_n \end{cases}$

1°) Calculer, à la main, les dix premiers termes de la suite (F_n).

2°) Ecrire un algorithme nommé listefibo(n), qui à un entier n fixé ($n \geq 2$), renvoie la liste L des n premières valeurs de la suite (F_n). Tester avec n = 2 , n = 4 , n =10

3°) En utilisant une récurrence « double », montrer l'égalité :

$$F_n = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n$$

Principe de la récurrence double :

On montre que l'égalité est vraie pour n = 0 et n = 1. On suppose que la formule est valable pour F_n et F_{n+1} (pour un n fixé), on montre qu'elle reste vraie pour F_{n+2}