Pages 19 et 20

EXERCICE 33 — Comprendre un algorithme

1. Compléter le tableau ci-dessous et en déduire la valeur que contient la variable A en fin d'algorithme.

```
A ← 100
Pour I variant de 1 à 4
    A ← A + 25
Fin Pour
```

	I	A
Avant le début de la boucle		100
Fin de la 1re itération	1	125
Fin de la 2e itération	2	150
Fin de la 3e itération	3	175
Fin de la 4e itération	4	200

La valeur que contient la variable A en fin d'algorithme est : 200

2. Si l'instruction « **Pour** I variant de 1 à 4 » est remplacée par « **Pour** I variant de 0 à 3 », quelle valeur contient la variable A en fin d'algorithme ?

La valeur que contient la variable A ne dépend pas des valeurs successives prises par la variable I, mais seulement du nombre de valeurs que cette variable I a prise. Il y a quatre entiers entre 1 et 4 et quatre entiers entre 0 et 3, donc la valeur que contient la variable A est également 200 dans ce cas.

EXERCICE 34 — Comprendre la fonction range

Cet exercice est un QCM. Pour chaque question, choisir la ou les bonne(s) réponse(s).

1. En langage Python, lors de l'instruction **for k in range(2,7)**, k prend les valeurs :

☐ 2 et 7 ☑ 2 ; 3 ; 4 ; 5 et 6 ☐ 2 ; 3 ; 4 ; 5 ; 6 et 7 ☐ 2 ; 3 ; 4 ; 5 ; 6 ; 7 et 8

2. En langage Python, lors de l'instruction **for k in range(5)**, k prend les valeurs :

☑ 0 ; 1 ; 2 ; 3 et 4 ☐ 0 ; 1 ; 2 ; 3 ; 4 et 5 ☐ 1 ; 2 ; 3 et 4 ☐ 1 ; 2 ; 3 ; 4 et 5

3. En langage Python, lorsque l'on veut que la variable k prenne successivement toutes les valeurs entières de 0 à 33, on peut utiliser l'instruction :

☐ for k in range(0,33) ☑ for k in range(0,34) ☐ for k in range(33) ☑ for k in range(34)

4. En langage Python, lors de l'instruction **for k in range(2,29)**, k prend :

☐ 29 valeurs ☐ 28 valeurs ☑ 27 valeurs ☐ 26 valeurs

EXERCICE 35 — Comprendre le rôle de l'indentation

Déterminer la valeur de la variable d après exécution du programme dans chacun des cas ci-dessous.

```
a=1;b=1;c=1;d=0
for k in range(2):
    a=a+b
    b=b+a
    c=c+b
    d=d+c
```

d = 16

```
a=1;b=1;c=1;d=0
for k in range(2):
    a=a+b
    b=b+a
    c=c+b
d=d+c
```

d = 12

```
a=1;b=1;c=1;d=0
for k in range(2):
    a=a+b
    b=b+a
c=c+b
d=d+c
```

d = 9

```
a=1;b=1;c=1;d=0
for k in range(2):
    a=a+b
b=b+a
c=c+b
d=d+c
```

d = 5

EXERCICE 36

Comprendre un programme

On considère le programme ci-contre, écrit en langage Python.

```
A="Je dis:"
for i in range(1,5):
    A=A+" bravo"
A=A+" !"
```

1. Que contient la variable A après exécution du programme ?

A contient la phrase : Je dis : bravo bravo bravo bravo !

2. Comment modifier ce programme pour que la phrase que contient la variable A après exécution du programme comporte huit fois le mot « bravo » ?

On peut remplacer l'instruction range(1,5) par range(1,9) puisque *i* prend alors successivement les valeurs entières de 1 à 8.

EXERCICE 37

Compléter et programmer un algorithme

Mme Langlois effectue un unique versement de 20 000 euros sur un livret d'épargne.

Le capital est augmenté chaque année de 100 euros par le versement d'intérêts.

1. Déterminer le capital disponible au bout d'un an sur le livret d'épargne de Mme Langlois.

Le capital disponible au bout d'un an est 20 100 €.

2. Compléter l'algorithme ci-contre pour que la variable C contienne, en fin d'algorithme, le capital disponible sur le livret au bout de 15 ans.

3. En utilisant l'algorithme ci-dessus, programmer en langage Python une fonction sans argument retournant le capital disponible au bout de 15 ans.

$C \longleftarrow$ 20 000

Pour *I* variant de 1 à 15

| $C \longleftarrow C + 100$

Fin Pour

```
def capital():
    C=20000
    for i in range (1,16):
        C=C+100
    return(C)
```

Le capital disponible au bout de 15 ans est 21 500 €.

EXERCICE 38

Chaînes de caractères

Compléter et modifier un programme

On veut programmer en langage Python une fonction qui a pour argument un mot ou une phrase et qui renvoie en sortie le nombre de fois où la lettre « e » se trouve dans ce mot ou cette phrase.
Dans la fonction ci-contre, len(ph) calcule la longueur de la chaîne de caractères ph.

```
def edansphrase(ph):
    s=len(ph)
    n=0
    for i in range(0,s):
        if ph[i]=="e":
            n=n+1
    return(n)
```

1. Compléter le programme ci-contre.

2. Modifier cette fonction afin qu'elle renvoie la fréquence de la lettre « a » dans la phrase ou le mot.

On change le texte « ph[i]=="e" » en « ph[i]=="a" ». Avant la dernière ligne, on introduit la ligne : f=n/s, puis on change l'instruction return(n) en return(f).