ALGO RÉVISIONS - SPÉCIALITÉ MATHS

Exercice 1

On considère l'algorithme suivant :

```

U ← 5
N ← 0
Tant que U ≤ 5000
    U ← 3 × U - 8
    N ← N + 1
Fin Tant que
```

Affirmation 3 : À la fin de l'exécution, la variable U contient la valeur 5 000.

Exercice 2

On considère la suite (u_n) définie, pour tout entier naturel non nul n , par :

$$u_n = \frac{n(n+2)}{(n+1)^2}$$

La suite (v_n) est définie par :

$v_1 = u_1$, $v_2 = u_1 \times u_2$ et pour tout entier naturel $n \geq 3$, $v_n = u_1 \times u_2 \times \dots \times u_n = v_{n-1} \times u_n$.

On considère l'algorithme incomplet ci-contre. Recopier et compléter sur la copie cet algorithme afin que, après son exécution, la variable V contiennent la valeur v_n où n est un nombre entier naturel non nul définie par l'utilisateur. Aucune justification n'est attendue.

Algorithme	
1.	$V \leftarrow 1$
2.	Pour i variant de 1 à n
3.	$V \leftarrow \frac{\dots(\dots+2)}{(\dots+1)^2}$
4.	$V \leftarrow \dots$
5.	Fin Pour

Exercice 3

En 2020, une influenceuse sur les réseaux sociaux compte 1 000 abonnés à son profil. On modélise le nombre d'abonnés ainsi : chaque année, elle perd 10% de ses abonnés auxquels s'ajoutent 250 nouveaux abonnés.

Pour tout entier naturel n , on note u_n le nombre d'abonnés à son profil en l'année $(2020+n)$, suivant cette modélisation. Ainsi $u_0 = 1000$.

La fonction Python nommée « suite » est définie ci-dessous. Dans le contexte de l'exercice, interpréter la valeur renvoyée par suite(10).

```

def suite(n) :
    u = 1000
    for i in range(n) :
        u = 0,9*u + 250
    return u
```

Exercice 4

On note T_n la température des gâteaux en degré Celsius, au bout de n minutes après leur sortie du congélateur; ainsi $T_0 = -19$. On admet que pour modéliser l'évolution de la température, on doit avoir la relation suivante pour tout entier naturel n , $T_{n+1} - T_n = -0,06 \times (T_n - 25)$.

On a $T_{n+1} = 0,94T_n + 1,5$

Le programme suivant, écrit en langage Python, doit renvoyer après son exécution la plus petite valeur de l'entier n pour laquelle $T_n \geq 10$.

```

def seuil() :
    n=0
    T= .....
    while T .....
        T= .....
        n=n+1
    return
```

Recopier ce programme sur la copie et compléter les lignes incomplètes afin que le programme renvoie la valeur attendue.

Exercice 5

Une action est cotée à 57 €. Sa valeur augmente de 3% tous les mois.

La fonction python seuil() qui renvoie le nombre de mois à attendre pour que sa valeur dépasse 200 € est :

a. <pre>def seuil() : m=0 v=57 while v < 200 : m=m+1 v = v*1.03 return m</pre>	b. <pre>def seuil() : m=0 v=57 while v > 200 : m=m+1 v = v*1.03 return m</pre>	c. <pre>def seuil() : v=57 for i in range (200) : v = v*1.03 return v</pre>	d. <pre>def seuil() : m=0 v=57 if v < 200 : m=m+1 else : v = v*1.03 return m</pre>
---	---	---	---

Exercice 6

pour tout entier naturel n , $u_{n+1} = 0,9u_n + 0,25$.

On estime que le médicament est réellement efficace lorsque sa quantité dans le sang du patient est supérieure ou égale à 1,8 mg.

a. Recopier et compléter le script écrit en langage Python suivant de manière à déterminer au bout de combien de périodes de trente minutes le médicament commence à être réellement efficace.

```

def efficace() :
    u=1
    n=0
    while ..... :
        u=.....
        n = n+1
    return n
```

b. Quelle est la valeur renvoyée par ce script? Interpréter ce résultat dans le contexte de l'exercice.

Exercice 7

$p_{n+1} = 0,3 + 0,7p_n^2$.

La fonction suivante, écrite en langage Python, a pour objectif de renvoyer les n premiers termes de la suite (p_n) .

```

1 def suite(n) :
2     p= ...
3     s=[p]
4     for i in range (...) :
5         p=...
6         s.append(p)
7     return (s)
```

Recopier, sur votre copie, cette fonction en complétant les lignes 2, 4 et 5 de façon à ce que la fonction suite (n) retourne, sous forme de liste, les n premiers termes de la suite.

Exercice 8

On définit deux suites (a_n) et (b_n) par :

$$\begin{cases} a_0 = \frac{1}{10} \\ b_0 = 1 \end{cases} \text{ et, pour tout entier naturel } n, \begin{cases} a_{n+1} = e^{-b_n} \\ b_{n+1} = e^{-a_n} \end{cases}$$

a. Calculer a_1 et b_1 . On donnera des valeurs approchées à 10^{-2} près.

b. On considère ci-dessous la fonction termes, écrite en langage Python.

```

def termes (n) :
    a=1/10
    b=...
    for k in range(0,n) :
        c= ...
        b = ...
        a = c
    return(a,...)
```

Recopier et compléter sans justifier le cadre ci-dessus de telle sorte que la fonction termes calcule les termes des suites (a_n) et (b_n) .

ALGO RÉVISIONS - SPÉCIALITÉ MATHS - CORRECTION

Exercice 1

À la fin de l'exécution, la variable U contient la première valeur strictement supérieure à 5 000. En effet l'algorithme ne s'arrête pas tant que $U \leq 5000$. En utilisant la calculatrice, on obtient la valeur 6565. L'affirmation 3 est donc fausse.

Exercice 2

On complète l'algorithme ci-contre afin que, après son exécution, la variable V contiennent la valeur v_n où n est un nombre entier naturel non nul défini par l'utilisateur.

Algorithme	
1.	$V \leftarrow 1$
2.	Pour i variant de 1 à n
3.	$U \leftarrow \frac{i(i+2)}{(i+1)^2}$
4.	$V \leftarrow V \times U$
5.	Fin Pour

Exercice 3

$u(10)$ donne le nombre d'abonnés au bout de 10 ans; une calculatrice donne ≈ 1977 .

Exercice 4

On complète le programme suivant, écrit en langage Python, qui doit renvoyer après son exécution la plus petite valeur de l'entier n pour laquelle $T_n \geq 10$.

```
def seuil() :  
    n = 0  
    T = -19  
    while T < 10  
        T = 0,94T + 1,5  
        n = n+1  
    return n
```

Exercice 5

Une action est cotée à 57 €. Sa valeur augmente de 3% tous les mois. Pour que la fonction seuil fonctionne, il faut que la boucle while s'exécute tant que $v < 200$ et que le nombre de mois soit augmenté de 1 à chaque exécution de la boucle.
Réponse a

Exercice 6

a. Recopier et compléter le script écrit en langage Python suivant de manière à déterminer au bout de combien de périodes de trente minutes le médicament commence à être réellement efficace.

```
def efficace() :  
    u=1  
    n=0  
    while u < 1.8 :  
        u=0.9*u+0.25  
        n = n+1  
    return n
```

b. Le script renvoie $n = 8$, car $u_8 \approx 1,854 > 1,8$.
Le médicament est réellement efficace après 4 heures.

Exercice 7

La fonction suivante, écrite en langage Python, a pour objectif de renvoyer les n premiers termes de la suite (p_n) .

On complète, les lignes 2, 4 et 5 de cette fonction de façon à ce que la fonction suite(n) retourne, sous forme de liste, les n premiers termes de la suite.

```
1 def suite(n) :  
2     p = 0.3  
3     s = [p]  
4     for i in range (n - 1)  
5         p = 0.3+0.7*p**2  
6         s.append(p)  
7     return (s)
```

Exercice 8

```
def termes(n) :  
    a = 1/10  
    b = 1  
    for k in range(0, n) :  
        c = exp(-b)  
        b = exp(-a)  
        a = c  
    return(a, b)
```