

Suites numériques – Série 1 – Correction

CONSIGNE

Pour chaque suite, calculer le terme demandé et préciser la valeur donnée à n .

CORRECTION

N°1

$$\begin{aligned}v_{n+1} &= 2v_n - 3 \\v_4 &= 2 \\n &= 4 \\v_5 &= 2v_4 - 3 = 1\end{aligned}$$

N°2

$$\begin{aligned}v_n &= 2n - 5 \\n &= 5 \\v_5 &= 2 \times 5 - 5 = 5\end{aligned}$$

N°3

$$\begin{aligned}u_n &= u_{n-1} + 5 \\u_2 &= 3 \\n &= 3 \\u_3 &= u_2 + 5 = 8\end{aligned}$$

N°4

$$\begin{aligned}u_n &= 2u_{n-1} + 2n + 1 \\u_3 &= 1 \\n &= 4 \\u_4 &= 2 \times u_3 + 2 \times 4 + 1 = 11\end{aligned}$$

N°5

$$\begin{aligned}u_{n+1} &= 5n + 3 \\n &= 5 \\u_6 &= 5 \times 5 + 3 = 28\end{aligned}$$

N°6

$$\begin{aligned}w_n &= 5n - 2w_{n-1} \\w_5 &= 3 \\n &= 6 \\w_6 &= 5 \times 6 - 2 \times w_5 = 24\end{aligned}$$

N°7

$$\begin{aligned}u_n &= 3(n + 1) - 5 \\n &= 7 \\u_7 &= 3 \times 8 - 5 = 19\end{aligned}$$

N°8

$$\begin{aligned}u_{n+1} &= n^2 + 2(n + 1) \\n &= 4 \\u_5 &= 4^2 + 2 \times 5 = 26\end{aligned}$$

N°9

$$\begin{aligned}v_n &= nv_{n-1} \\v_7 &= 3 \\n &= 8 \\v_8 &= 8 \times v_7 = 24\end{aligned}$$

N°10

$$\begin{aligned}v_{n+1} &= v_{n-1} + 3n \\v_2 &= -5 \\n &= 3 \\v_4 &= v_2 + 3 \times 3 = 4\end{aligned}$$

FIN

Suites numériques – Série 2 – Correction

CONSIGNE

Une suite étant définie par une expression algébrique, dire s'il s'agit d'une expression explicite ou d'une formule de récurrence.

CORRECTION

N°1

Pour tout entier $n \geq 0$,
 $u_n = -5n + 2$

Formule explicite

N°2

Pour tout entier $n \geq 0$,
 $u_{n+1} = -5u_n + 2$

Relation de récurrence

N°3

Pour tout entier $n \geq 0$,
 $u_{n+1} = n^2 + 2n - 1$

Formule explicite

N°4

Pour tout entier $n \geq 0$,
 $u_n = \frac{1}{n+1}$

Formule explicite

N°5

Pour tout entier $n \geq 1$,
 $u_n = u_{n-1} + 1$

Relation de récurrence

N°6

Pour tout entier $n \geq 0$,
 $u_{n+1} = \frac{u_n}{2}$

Relation de récurrence

N°7

Pour tout entier $n \geq 0$,
 $u_n = -5$

Formule explicite

N°8

Pour tout entier $n \geq 0$,
 $u_{n+1} = \frac{4}{3}(n-1)$

Formule explicite

N°9

Pour tout entier $n \geq 0$,
 $u_n = 2(u_{n-1})^2 - 5$

Relation de récurrence

N°10

Pour tout entier $n \geq 0$,
 $u_n = 3(n-1) + 2u_n$
 $u_n = -3(n-1)$

Formule explicite

FIN

Suites numériques – Série 3 – Correction

CONSIGNE

Une suite étant définie par une forme explicite, calculer le terme demandé.

CORRECTION

N°1

$$n = 5$$

Pour tout entier $n \geq 0$,

$$u_n = -5n + 2$$

$$u_5 = -5 \times 5 + 2 = -23$$

N°2

$$n = 7$$

Pour tout entier $n \geq 0$,

$$u_n = -n^2 + 2$$

$$u_7 = -7^2 + 2 = -47$$

N°3

$$n = 0$$

Pour tout entier $n \geq 0$,

$$u_{n+1} = n^2 + 2n - 1$$

$$u_{0+1} = 0^2 + 2 \times 0 - 1$$

$$u_{0+1} = -1$$

N°4

$$n = 3$$

Pour tout entier $n \geq 1$,

$$u_n = (n-1)^2 + 1$$

$$u_3 = (3-1)^2 + 1 = 5$$

N°5

$$n = 2$$

Pour tout entier $n \geq 0$,

$$u_n = \frac{1}{n+1}$$

$$u_2 = \frac{1}{2+1} = \frac{1}{3}$$

N°6

$$n = 8$$

Pour tout entier $n \geq 0$,

$$u_n = \frac{1}{2}n + 3$$

$$u_8 = \frac{1}{2} \times 8 + 3 = 7$$

N°7

$$n = 4$$

Pour tout entier $n \geq 0$,

$$u_{n+1} = \frac{4}{3}(n-1)$$

$$u_{4+1} = \frac{4}{3}(4-1) = 4$$

N°8

$$n = 3$$

Pour tout entier $n \geq 0$,

$$u_n = 6$$

$$u_3 = 6$$

N°9

$$n = 0$$

Pour tout entier $n \geq 0$,

$$u_n = (2n-1)^2$$

$$u_0 = (2 \times 0 - 1)^2 = 1$$

N°10

$$n = 1$$

Pour tout entier $n \geq 1$,

$$u_{n-1} = n(n+1)^2$$

$$u_{1-1} = 1(1+1)^2 = 4$$

FIN

Suites numériques– Série 4 – Correction

CONSIGNE

Une suite étant définie par une formule algébrique, adapter cette formule à l'indice indiqué.

CORRECTION

N°1

Pour tout entier $n \geq 0$,
 $u_n = -5n + 2$
 $u_{n+1} = -5(n+1) + 2$
 $u_{n+1} = -5n - 3$

N°2

Pour tout entier $n \geq 0$,
 $u_n = -n^2 + 2$
 $u_p = -p^2 + 2$

N°3

Pour tout entier $n \geq 1$,
 $u_{n+1} = 2u_n + 1$
 $u_n = 2u_{n-1} + 1$

N°4

Pour tout entier $n \geq 1$,
 $u_n = (n-1)^2 + 1$
 $u_{n+1} = (n+1-1)^2 + 1$
 $u_{n+1} = n^2 + 1$

N°5

Pour tout entier $n \geq 1$,
 $u_n = \frac{1}{n+1}$
 $u_{n-1} = \frac{1}{(n-1)+1} = \frac{1}{n}$

N°6

Pour tout entier $n \geq 0$,
 $u_n = \frac{1}{2}n + 3$
 $u_{p+1} = \frac{1}{2}(p+1) + 3$
 $u_{p+1} = \frac{1}{2}p + \frac{7}{2}$

N°7

Pour tout entier $n \geq 1$,
 $u_{n+1} = \frac{4}{3}(u_n - 1)$
 $u_n = \frac{4}{3}(u_{n-1} - 1)$

N°8

Pour tout entier $n \geq 1$,
 $u_n = n + u_{n-1}$
 $u_{p+1} = p+1 + u_{p+1-1}$
 $u_{p+1} = p+1 + u_p$

N°9

Pour tout entier $n \geq 0$,
 $u_n = (2n-1)^2$
 $u_{2n} = (2 \times 2n-1)^2$
 $u_{2n} = (4n-1)^2$

N°10

Pour tout entier $n \geq 1$,
 $u_{n-1} = n(n+1)^2$
 $u_{n+1} = (n+2)((n+2)+1)^2$
 $u_{n+1} = (n+2)(n+3)^2$

FIN

Suites numériques – Série 5– Correction

CONSIGNE

Une suite u étant arithmétique ou géométrique, calculer un terme ou la raison de la suite.

CORRECTION

N°1

La suite u est arithmétique de raison -2 et $u_0 = 5$.

$$u_3 = 5 + 3 \times (-2)$$

$$u_3 = -1$$

N°2

La suite u est géométrique de raison -2 et $u_0 = 3$.

$$u_4 = 3 \times (-2)^4$$

$$u_4 = 48$$

N°3

La suite u est arithmétique de raison 5 et $u_1 = 4$.

$$u_4 = 4 + 3 \times 5$$

$$u_4 = 19$$

N°4

La suite u est arithmétique

$$u_1 = 8 \text{ et } u_5 = -8.$$

Calculer sa raison.

$$u_5 - u_1 = 4r = -16$$

$$\text{donc } r = -4$$

N°5

La suite u est arithmétique de raison 4 et $u_2 = -10$.

$$u_5 = -10 + 3 \times 4 = 2$$

N°6

La suite u est géométrique de raison -2 et $u_1 = 5$.

$$u_4 = 5 \times (-2)^3 = -40$$

N°7

La suite u est géométrique à termes positifs.

$$u_1 = 20 \text{ et } u_3 = 500.$$

Calculer sa raison.

$$\frac{u_3}{u_1} = q^2 = 25 \text{ donc } q = 5$$

N°8

La suite u est arithmétique de raison 3 et $u_6 = 12$.

$$u_8 = 12 + 2 \times 3 = 18$$

N°9

La suite u est arithmétique.

$$u_0 = 5 \text{ et } u_3 = -1.$$

Calculer sa raison.

$$u_3 - u_0 = 3r = -6$$

$$\text{donc } r = -2$$

N°10

La suite u est géométrique de raison -1 et $u_1 = -7$.

$$u_{10} = -7 \times (-1)^9$$

$$u_{10} = 7$$

FIN

Suites numériques – Série 6– Correction

CONSIGNE

Déterminer le sens de variation de chaque suite.

Correction

Activités mentales et automatismes en classe de première
IREM de Clermont-Ferrand

Question 1

$$\begin{cases} u_1 = -2 \\ u_{n+1} = u_n + \frac{3}{n} \text{ pour tout } n \in \mathbb{N}^* \end{cases}$$

Pour tout $n \in \mathbb{N}^*$, $u_{n+1} - u_n = \frac{3}{n}$ et $\frac{3}{n} > 0$

donc la suite u est strictement croissante.

Question 2

$$\begin{cases} u_0 = -1 \\ u_{n+1} = u_n + e^{-n} \text{ pour tout } n \in \mathbb{N} \end{cases}$$

Pour tout $n \in \mathbb{N}$, $u_{n+1} - u_n = e^{-n}$ et $e^{-n} > 0$

donc la suite u est strictement croissante.

Question 3

$$\begin{cases} u_0 = 3 \\ u_{n+1} = u_n - \frac{2}{n+1} \text{ pour tout } n \in \mathbb{N} \end{cases}$$

Pour tout $n \in \mathbb{N}$, $u_{n+1} - u_n = -\frac{2}{n+1}$

et $-\frac{2}{n+1} < 0$

donc la suite u est strictement décroissante.

Question 4

Pour tout $n \in \mathbb{N}$, $u_n = 9 - 7n$

Pour tout $n \in \mathbb{N}$, $u_n = f(n)$ avec f la fonction définie sur $[0; +\infty[$ par $f(x) = -7x + 9$

f est strictement décroissante sur $[0; +\infty[$

donc la suite u est strictement décroissante.

Question 5

Pour tout $n \in \mathbb{N}$, $u_n = \left(\frac{1}{2}\right)^n$

Pour tout $n \in \mathbb{N}$, $u_n = q^n$ avec $q = \frac{1}{2}$

$$0 < \frac{1}{2} < 1$$

donc la suite u est strictement décroissante.

Question 6

Pour tout $n \in \mathbb{N}$, $u_n = (\sqrt{2})^n$

Pour tout $n \in \mathbb{N}$, $u_n = q^n$ avec $q = \sqrt{2}$

$$\sqrt{2} > 1$$

donc la suite u est strictement croissante.

Question 7

Pour tout $n \in \mathbb{N}$, $u_n = 3^{-n}$

Pour tout $n \in \mathbb{N}$, $u_n = q^n$ avec $q = \frac{1}{3}$

$$0 < \frac{1}{3} < 1$$

donc la suite u est strictement décroissante.

Question 8

Pour tout $n \in \mathbb{N}^*$, $u_n = \frac{1}{n}$

Pour tout $n \in \mathbb{N}^*$, $u_n = f(n)$ avec f la fonction définie sur $]0; +\infty[$ par $f(x) = \frac{1}{x}$

f est strictement décroissante sur $]0; +\infty[$

donc la suite u est strictement décroissante.

Question 9

Pour tout $n \in \mathbb{N}^*$, $u_n = -\frac{4}{n} - 1$

Pour tout $n \in \mathbb{N}^*$, $u_n = f(n)$ avec f la fonction définie sur $]0; +\infty[$ par $f(x) = -\frac{4}{x} - 1$

f est strictement croissante sur $]0; +\infty[$

donc la suite u est strictement croissante.

Question 10

Pour tout $n \in \mathbb{N}$, $u_n = e^{-n+2}$

Pour tout $n \in \mathbb{N}$, $u_n = f(n)$ avec f la fonction définie sur $[0; +\infty[$ par $f(x) = e^{-x+2}$

f est strictement décroissante sur $[0; +\infty[$

donc la suite u est strictement décroissante.

Fin

Activités mentales et automatismes en classe de première
IREM de Clermont-Ferrand

Suites numériques – Série 7 – Correction

CONSIGNE

Reconnaître une suite définie par un programme Python. Donner son expression (explicite ou récurrence).

Comparer deux suites définies par des programmes Python Déterminer le sens de variation de chaque suite.

Correction

Activités mentales et automatismes en classe de première
IREM de Clermont-Ferrand

Question 1

L'algorithme suivant définit-il une suite ?

```
U ← 2
U ← 3U - 5
```

L'algorithme ne calcule qu'une seule valeur de U .

Donc : **NON**

Question 2

Lorsque n désigne un entier naturel, quelle suite définit l'algorithme suivant ?

```
U ← 1
Pour i allant de 0 à n :
    U ← 2U2 - 1
```

Pour tout entier naturel n ,

$$\begin{cases} u_0 = 1 \\ u_{n+1} = 2u_n^2 - 1 \end{cases}$$

Question 3

Lorsque n désigne un entier naturel, quelle suite définit l'algorithme suivant ?

```
U ← 1
Pour i allant de 1 à n :
    U ← 5√U
```

Pour tout entier naturel n ,

$$\begin{cases} u_0 = 1 \\ u_{n+1} = 5\sqrt{u_n} \end{cases}$$

Question 4

Lorsque n désigne un entier naturel, ces deux algorithmes définissent-ils la même suite ?

```
U ← 2
Pour i allant de 1 à n :
    U ← U + 6

Pour i allant de 0 à n :
    V ← 2 + 6i
```

(u_n) est une suite arithmétique de 1^{er} terme $u_0 = 2$ et de raison 6.

Donc : $u_n = u_0 + nr = 2 + 6n = v_n$

Question 5

Lorsque n désigne un entier naturel, ces deux algorithmes définissent-ils la même suite ?

```
U ← -3
Pour i allant de 0 à n :
    U ← U - 2

Pour i allant de 1 à n :
    V ← -3 - 2i
```

$$u_0 = -3$$

$$v_0 = -3 - 2 \times 1 = -5.$$

$$u_n \neq v_n$$

Question 6

Lorsque n désigne un entier naturel, ces deux algorithmes définissent-ils la même suite ?

```
U ← -1
Pour i allant de 0 à n :
    U ← 3 × U

Pour i allant de 0 à n :
    V ← -1 × 3i
```

(u_n) est une suite géométrique de 1^{er} terme $u_0 = -1$ et de raison 3.

Donc : $u_n = u_0 \times q^n = (-1) \times 3^n = v_n$

Question 7

Lorsque n désigne un entier naturel, ces deux algorithmes définissent-ils la même suite ?

```
U ← 7
Pour i allant de 1 à n :
    U ← 0,5 × U

Pour i allant de 1 à n :
    V ← 7 × 0,5i
```

$u_0 = 7$ et v_0 n'existe pas !

Donc : $u_n \neq v_n$

Question 8

Compléter l'algorithme suivant pour qu'il définisse la suite (u_n) telle que :

$$u_1 = 3 \text{ et } u_{n+1} = u_n^2 - 5 \text{ pour tout } n \in \mathbb{N}$$

```
U ← ...3...
Pour i allant de ...1... à ...n... :
    U ← ...U2 - 5...
```

Question 9

L'algorithme suivant définit-il (u_n) telle que : $u_0 = 1$ et $u_{n+1} = 5u_n - 3$ pour tout $n \in \mathbb{N}$?

```
U ← 0
Pour i allant de 1 à n :
    U ← 5U - 3
```

$$u_0 = 1$$

Le 1^{er} terme de l'algorithme est 0.

Donc l'algorithme ne convient pas.

Question 10

L'algorithme suivant définit-il (u_n) telle que : $u_0 = 1$ et $u_{n+1} = 6u_n + n - 2$ pour tout $n \in \mathbb{N}$?

```
U ← 1
Pour i allant de 1 à n :
    U ← 6U + n - 2
```

$$u_0 = 1 \text{ et } u_1 = 6 \times u_0 + 0 - 2 = 4$$

Algorithme :

le 1^{er} terme est 1, le 2^{ème} $6 \times 1 + 1 - 2 = 5$.

Donc l'algorithme ne convient pas.

Fin

Activités mentales et automatismes en classe de première
IREM de Clermont-Ferrand

Suites numériques – Série 8 – Correction

CONSIGNE

Déterminer les valeurs renvoyées par un programme Python qui définit une suite.
Reconnaître une suite définie par un programme Python.
Compléter un programme Python afin qu'il définisse une suite donnée.

Correction

Activités mentales et automatismes en classe de première
IREM de Clermont-Ferrand

Question 1

On considère le programme écrit ci-dessous

```
def u(n):  
    u = 2 * n ** 2 - 3  
    return u
```

$$u_n = 2n^2 - 3$$

Question 2

On considère le programme écrit ci-dessous

```
def u(n):  
    if n%2 == 0:  
        u = n/2  
    else:  
        u = 3 * n  
    return u
```

4 est pair donc $u(4) = \frac{4}{2} = 2$
5 est impair donc $u(5) = 3 \times 5 = 15$

Question 3

On considère le programme écrit ci-dessous :

```
u = 3  
for k in range(2):  
    u = u ** 2 - 1  
print(u)
```

On répète deux fois l'instruction $u = u^2 - 1$
 $u = 3$ puis $u = 9 - 1 = 8$
puis $u = 64 - 1 = 63$
la valeur affichée à l'exécution du programme est 63.

Question 4

On considère le programme écrit ci-dessous :

```
v = 2  
for k in range(1,3):  
    v = 3 - 2 * v  
print(v)
```

On répète deux fois l'instruction $v = 3 - 2v$
 $v = 2$ puis $v = 3 - 4 = -1$
puis $v = 3 + 2 = 5$
la valeur affichée à l'exécution du programme est 5.

Question 5

On considère le programme écrit ci-dessous :

```
u = 5  
for k in range(3):  
    u = 2 * u + k  
print(u)
```

$k = 0$ $u = 10 + 0 = 10$
 $k = 1$ $u = 20 + 1 = 21$
 $k = 2$ $u = 42 + 2 = 44$
la valeur affichée à l'exécution du programme est 44.

Question 6

On considère le programme écrit ci-dessous :

```
def v(n):  
    v = -1  
    for k in range(n):  
        v = 2 * v - 1  
    return v
```

$v_0 = -1$
et pour tout entier naturel n :
 $v_{n+1} = 2v_n - 1$

Question 7

On considère le programme écrit ci-dessous :

```
def u(n):  
    u = 1  
    for i in range(n):  
        u = 4 * u - 3 * i  
    return u
```

$u_0 = 1$
et pour tout entier naturel n : $u_{n+1} = 4u_n - 3n$

Question 8

On considère le programme écrit ci-dessous :

```
def u(n):  
    u = 1  
    for i in range(1, n + 1):  
        u = u + 1/i  
    return u
```

$u_0 = 1$
et pour tout entier naturel n : $u_{n+1} = u_n + \frac{1}{n+1}$

Question 9

Compléter le programme Python ci-dessous afin qu'il définisse la suite définie par :

$$\begin{cases} u_0 = 5 \\ u_{n+1} = 3u_n - u_n^2 \end{cases}$$

```
def u(n):  
    u = 5  
    for k in range(n):  
        u = 3 * u - u ** 2  
    return u
```

Question 10

Compléter le programme Python ci-dessous afin qu'il définisse la suite définie par :

$$\begin{cases} u_0 = -2 \\ u_{n+1} = u_n - n^2 - 1 \end{cases}$$

```
def u(n):  
    u = -2  
    for k in range(n):  
        u = u - k ** 2 - 1  
    return u
```

Fin

Activités mentales et automatismes en classe de première
IREM de Clermont-Ferrand

Suites numériques – Série 9– Correction

CONSIGNE

Les premiers termes d'une suite sont définis dans un programme Python, et stockés dans une liste. Répondre aux questions concernant la suite, ou le liste.

Correction

Activités mentales et automatismes en classe de première
IREM de Clermont-Ferrand

Question 1

```
L = []  
for i in range(10):  
    L.append( 2*i + 1 )
```

Écrire l'expression de la suite (u_n) .

$$u_n = 2n + 1, n \in \mathbb{N}$$

Question 2

```
L = []  
for i in range(10):  
    L.append( 2*i + 1 )
```

A quel terme de la suite $L[6]$ correspond-il et quelle est sa valeur ?

$L[6]$ correspond au 7^{ème} terme, u_6
et $u_6 = 13$

Question 3

```
L = []  
for i in range(1,10):  
    L.append( 2*i + 1 )
```

A quel terme de la suite $L[6]$ correspond-il et quelle est sa valeur ?

$L[6]$ correspond au 7^{ème} terme, u_7
et $u_7 = 15$

Question 4

```
L = []  
for i in range(10):  
    L.append( 2*i + 1 )
```

Que contient la liste L à la fin de l'exécution du programme ?

L contient les termes de $i=0$ à $i=9$
 $L=[1,3,5,7,9,11,13,15,17,19]$

Question 5

```
L = [ 3 ]  
for i in range(1,15):  
    L.append ( 3*L[i-1] + 1 )
```

A quoi correspond la première ligne du programme pour la suite ?

La première ligne indique le 1^{er} terme de la suite, par exemple $u_0 = 3$

Question 6

```
L = [ 3 ]  
for i in range(1,15):  
    L.append ( 3*L[i-1] + 1 )
```

Compléter la phrase :
d'après ce programme,

$$u_n = 3u_{n-1} + 1 \text{ pour } n \geq 1$$

Question 7

```
L = [ 3 ]  
for i in range(1,15):  
    L.append ( 3*L[i-1] + 1 )
```

Compléter la phrase :
d'après ce programme,

$$u_{n+1} = 3u_n + 1 \text{ pour tout } n \in \mathbb{N}$$

Question 8

```
L = [ ]  
for i in range(1,10):  
    L.append ( 1 / i )
```

A quel terme de la suite, $L[2]$ correspond-il et quelle est sa valeur ?

$L[2]$ correspond au 3^{ème} terme, donc $i=3$
et $L[3] = 0.333 \left(\frac{1}{3} \right)$

Question 9

```
L = [ ]  
for i in range(3):  
    L.append ( 1+3 / i )
```

Que contient $L[2]$?

Rien !

La boucle commence à $i=0$; on ne peut pas calculer le 1^{er} terme de la liste.

Le programme est en erreur.

Question 10

On a, pour tout $n \geq 1$, $u_n = \frac{2}{n}$

Compléter le programme pour que la liste contienne les 10 premiers termes de la suite (u_n) :

```
L = [ ]  
for i in range (1,11):  
    L.append ( 2 / i )
```

Fin

Activités mentales et automatismes en classe de première
IREM de Clermont-Ferrand

Suites numériques – Série 10 – Correction

CONSIGNE

On s'intéresse à la somme de termes consécutifs d'une suite numérique. Répondre à la question posée puis pour les cinq dernières questions, déterminer la ou les bonne(s) réponse(s).

Correction

Activités mentales et automatismes en classe de première
IREM de Clermont-Ferrand

Question 1

On considère une suite u .
Combien y a-t-il de termes dans la somme
 $u_0 + u_1 + \dots + u_n$?

Cette somme comporte $n + 1$ termes.

Question 2

On considère une suite u .
Soit n un entier supérieur ou égal à 2.
Combien y a-t-il de termes dans la somme
 $u_2 + u_3 + \dots + u_n$?

Cette somme comporte $n - 1$ termes.

Question 3

Calculer la somme
 $S = 1 + 2 + 3 + \dots + 20$

On utilise la propriété :
 $1 + 2 + \dots + n = \frac{n(n+1)}{2}$

$$S = \frac{20 \times 21}{2} = 210$$

Question 4

Calculer la somme
 $S = 2 + 3 + 4 + \dots + 99$

$$S = (1 + 2 + \dots + 99) - 1$$

$$S = \frac{99 \times 100}{2} - 1 = 4950 - 1 = 4949$$

Question 5

Calculer la somme
 $10 + 11 + 12 + \dots + 30$

$$S = (1 + 2 + \dots + 30) - (1 + 2 + \dots + 9)$$

$$S = \frac{30 \times 31}{2} - \frac{9 \times 10}{2} = 465 - 45 = 420$$

Question 6

Soit n un entier naturel.
La somme $S = 1 + 2 + 4 + \dots + 2^n$ est égale à

- a) $2^n - 1$ b) $\frac{1-2^{n+1}}{-1}$ c) $1 - 2^{n+1}$ d) $2^{n+1} - 1$

On utilise la propriété :

$$\text{Soit } q \neq 1, 1 + q + \dots + q^n = \frac{1 - q^{n+1}}{1 - q}$$

$$S = \frac{1 - 2^{n+1}}{1 - 2} = \frac{1 - 2^{n+1}}{-1} = 2^{n+1} - 1$$

Question 7

Soit n un entier naturel non nul.
La somme $S = 1 + 4 + 16 + \dots + 4^{n-1}$ est égale à

- a) $\frac{1-4^n}{3}$ b) $\frac{4^{n+1}-1}{3}$ c) $\frac{4^n-1}{3}$ d) $\frac{4^{n+1}+1}{3}$

$$S = \frac{1 - 4^{n-1+1}}{1 - 4} = \frac{4^n - 1}{3}$$

Question 8

Soit n un entier naturel.
La somme $S = 1 + \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{2^n}$ est égale à

- a) $2 \left(1 - \frac{1}{2^{n+1}}\right)$ b) $\frac{1}{2} \left(1 - \frac{1}{2^{n+1}}\right)$ c) $\frac{1}{2} - \frac{1}{2^{n+2}}$ d) $2 - \frac{1}{2^n}$

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

Question 9

Soit n un entier naturel.
La somme $S = 1 - 3 + 9 + \dots + (-3)^n$ est égale à

- a) $\frac{1-3^{n+1}}{2}$ b) $\frac{1+3^{n+1}}{4}$ c) $\frac{1-(-3)^{n+1}}{4}$ d) $\frac{1+3(-3)^n}{4}$

$$S = \frac{1 - (-3)^{n+1}}{1 - (-3)} = \frac{1 - (-3)^{n+1}}{4}$$

$$S = \frac{1 - (-3)(-3)^n}{4} = \frac{1 + 3(-3)^n}{4}$$

Question 10

On considère la suite géométrique de premier terme 4 et de raison 2. La somme de ses 10 premiers termes est égale à

- a) $4(2^{11} - 1)$ b) $4(2^{10} - 1)$ c) $2^{12} - 4$ d) $2^{10} - 1$

On utilise la propriété :

$$S = 1^{\text{er}} \text{ terme de la somme} \times \frac{1 - q^{\text{nombre de termes}}}{1 - q}$$

$$S = 4 \frac{1 - 2^{10}}{1 - 2} = 4(2^{10} - 1) = 2^{12} - 4$$

Fin

Activités mentales et automatismes en classe de première
IREM de Clermont-Ferrand

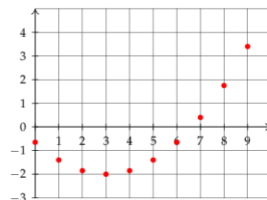
Suites numériques – Série 11 – Correction

CONSIGNE

On conjecture graphiquement ou à partir d'un tableau de valeurs la limite, si elle existe, d'une suite.

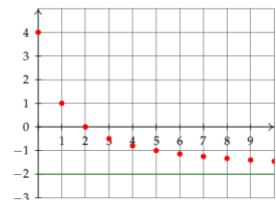
CORRECTION

Question (1) – Correction



La suite semble tendre vers $-\infty$.

Question (2) – Correction



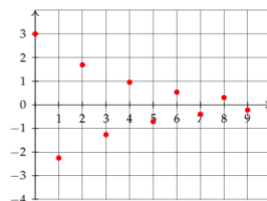
La suite semble tendre vers $-\infty$.

Question (3) – Correction

A	B	C	D	E
n	u_n	n	u_n	
1	0	10	40	
2	-2,75	20	-720	
3	0	30	-3780	
4	6,75	40	-10640	
5	16	50	-22800	
6	26,25	60	-41760	
7	36	70	-69020	
8	43,75	80	-109080	
9	48	90	-154440	
10	47,25	100	-215600	

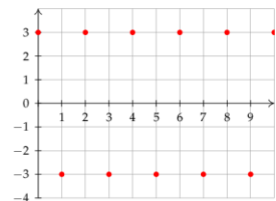
La suite semble tendre vers $-\infty$.

Question (4) – Correction



La suite semble tendre vers 0.

Question (5) – Correction



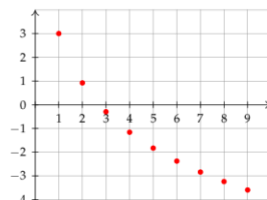
La suite semble ne pas avoir de limite.

Question (6) – Correction

A	B	C	D	E
n	u_n	n	u_n	
1	0	10	-495	
2	-500	20	-490	
3	-499,5	30	-485	
4	-499	40	-480	
5	-498,5	50	-475	
6	-498	60	-470	
7	-497,5	70	-465	
8	-497	80	-460	
9	-496,5	90	-455	
10	-496	100	-450	

La suite semble tendre vers $-\infty$.

Question (7) – Correction



La suite semble tendre vers $-\infty$.

Question (8) – Correction

A	B	C	D	E
n	u_n	n	u_n	
1	0	10	-3	
2	1	20	-3	
3	2	30	-3	
4	3	40	-3	
5	4	50	-3	
6	5	60	-3	
7	6	70	-3	
8	7	80	-3	
9	8	90	-3	
10	9	100	-3	

La suite semble ne pas avoir de limite.

Question (9) – Correction

A	B	C	D	E
n	u_n	n	u_n	
1	3	11	2,096096091	
2	2,5	12	2,083333333	
3	2,333333333	13	2,076923077	
4	2,25	14	2,071428571	
5	2,2	15	2,066666667	
6	2,166666667	16	2,0625	
7	2,142857143	17	2,058823529	
8	2,125	18	2,055555556	
9	2,111111111	19	2,052631579	
10	2,1	20	2,05	

La suite semble tendre vers 2.

Question (10) – Correction

A	B	C	D	E
n	u_n	n	u_n	
1	0	10	0,107374102	
2	-0,8	20	0,011529215	
3	0,64	30	0,00123794	
4	-0,512	40	0,000132923	
5	0,4096	50	1,42725E-05	
6	-0,32768	60	1,5325E-06	
7	0,262144	70	1,6455E-07	
8	-0,2097152	80	1,76685E-08	
9	0,16777216	90	1,89714E-09	
10	-0,13421773	100	2,03704E-10	

La suite semble tendre vers 0.

FIN

Suites numériques – Série 12 – Correction

CONSIGNE

On s'intéresse aux algorithmes de seuils. Répondre à la question.

CORRECTION

Question (1) – Correction

On considère l'algorithme ci-dessous :

```
N ← 0
U ← 1
Tant que U < 10 faire :
    U ← U + 2
    N ← N + 1
Fin Tant que
Renvoyer N
```

Quelle valeur contient la variable N à la fin de l'exécution de cet algorithme ?

N contient la valeur 5.

Question (2) – Correction

On cherche le plus petit entier N à partir duquel la suite (u_n) prend des valeurs supérieures ou égales à 10. On considère l'algorithme ci-dessous :

```
U ← 1
Tant que U < 10 faire :
    U ← U + 1/2
    N ← N + 1
Fin Tant que
Renvoyer N
```

Pour quelle(s) raison(s) cet algorithme ne répond-il pas à la question ?

Cet algorithme ne convient pas car N n'est pas initialisée.

Question (3) – Correction

On cherche le plus petit entier N à partir duquel la suite (u_n) prend des valeurs supérieures ou égales à 10. On considère l'algorithme ci-dessous :

```
N ← 0
U ← 1
Tant que U < 10 faire :
    U ← U - 3
    N ← N + 1
Fin Tant que
Renvoyer N
```

Pour quelle(s) raison(s) cet algorithme ne répond-il pas à la question ?

U est initialisé à 1 et ses valeurs de U décroissent à chaque tour de boucle donc U ne peut atteindre la valeur de 10.

Question (4) – Correction

On considère l'algorithme ci-dessous :

```
N ← 0
U ← 1
Tant que U < 10 faire :
    U ← U × 2
    N ← N + 1
Fin Tant que
Renvoyer N
```

Quelle valeur contient la variable N à la fin de l'exécution de cet algorithme ?

N contient la valeur 4.

Question (5) – Correction

On cherche le plus petit entier N à partir duquel la suite (u_n) prend des valeurs supérieures ou égales à 10. On considère l'algorithme ci-dessous :

```
N ← 0
U ← 1
Tant que U < 10 faire :
    U ← U × 2
    N ← N + 1
Fin Tant que
Renvoyer N
```

Pour quelle(s) raison(s) cet algorithme ne répond-il pas à la question ?

Cet algorithme ne convient pas car N n'est pas incrémenté à chaque tour de boucle.

Question (6) – Correction

Compléter l'algorithme suivant afin qu'il détermine le plus petit entier n tel que $u_n \geq 85$.

```
N ← 0
U ← 65
Tant que u < 85
    N ← N + 1
    U ← U × 0,8 + 18
Fin Tant que
Renvoyer N
```

Question (7) – Correction

Déterminer l'algorithme qui permet de calculer le plus petit entier à partir duquel A dépasse 0,5.

<pre>A ← 0,3 N ← 0 Tant que A < 0,5 A ← 0,92 × A + 0,05 N ← N + 1 Fin Tant que Renvoyer N</pre> <p>Algorithme 1</p>	<pre>A ← 0,3 N ← 0 Tant que A > 0,5 A ← 0,92 × A + 0,05 N ← N + 1 Fin Tant que Renvoyer N</pre> <p>Algorithme 2</p>
<pre>A ← 0,3 N ← 0 Tant que A < 0,5 A ← 0,92 × A + 0,05 N ← N + 1 Fin Tant que Renvoyer N</pre> <p>Algorithme 3</p>	

L'algorithme correct est l'algorithme 1.

Question (8) – Correction

Que renvoie l'algorithme suivant lorsque $\epsilon < \frac{1}{3}$?

```
n ← 1
Tant que n × (1/3)^n > epsilon
    n ← n + 1
Fin Tant que
Renvoyer n
```

Cet algorithme renvoie le plus petit entier n à partir duquel $n \left(\frac{1}{3}\right)^n$ devient inférieur ou égal à epsilon.

Question (9) – Correction

Soit (u_n) la suite définie par $u_0 = 5000$ et, pour tout entier $n \geq 0$, $u_{n+1} = 1,02u_n$. Compléter l'algorithme suivant, de telle sorte qu'il renvoie le plus petit entier n à partir duquel u_n dépasse 10 000.

```
n ← 0
u ← 5000
Tant que u < 10000
    n ← n + 1
    u ← 1,02 × u
Fin Tant que
Renvoyer n
```

Question (10) – Correction

Que représente la valeur de la variable n à la fin de l'exécution de cet algorithme ?

```
n ← 0
u ← 1
Tant que u ≤ 10^6 :
    n ← n + 1
    u ← 2u + 1
Fin Tant que
Renvoyer n
```

Cet algorithme renvoie le plus petit entier n à partir duquel la suite définie par $u_0 = 1$ et, pour tout entier $n \geq 1$, $u_{n+1} = 2u_n + 1$ prend des valeurs strictement supérieures à 10^6 .

FIN

Suites numériques – Série 13 – Correction

CONSIGNE

On s'intéresse aux algorithmes de seuils mais écrits en python. Répondre à la question.

CORRECTION

Question (1) – Correction

On considère le programme ci-dessous :

```
1 def q1():
2     N = 0
3     U = 1
4     while U < 10:
5         U = U*2
6         N = N+1
7     return N
```

Quelle valeur contient la variable N à la fin de l'exécution de ce programme ?

N contient la valeur 5.

Question (2) – Correction

On cherche le plus petit entier N à partir duquel la suite (u_n) prend des valeurs supérieures ou égales à 10. On considère le programme ci-dessous :

```
1 def q2():
2     U = 1
3     while U < 10:
4         U = U+1/2
5         N = N+1
6     return N
```

Pour quelle(s) raison(s) ce programme ne répond-il pas à la question ?

Ce programme ne convient pas car N n'est pas initialisée.

Question (3) – Correction

On cherche le plus petit entier N à partir duquel la suite (u_n) prend des valeurs supérieures ou égales à 10. On considère le programme ci-dessous :

```
1 def q3():
2     N = 0
3     U = 1
4     while U < 10:
5         U = U-3
6         N = N+1
7     return N
```

Pour quelle(s) raison(s) ce programme ne répond-il pas à la question ?

U est initialisé à 1 et ses valeurs de U décroissent à chaque tour de boucle donc U ne peut atteindre la valeur de 10.

Question (4) – Correction

On considère le programme ci-dessous :

```
1 def q4():
2     N = 0
3     U = 1
4     while U < 10:
5         U = U*2
6         N = N+1
7     return N
```

Quelle valeur contient la variable N à la fin de l'exécution de ce programme ?

N contient la valeur 4.

Question (5) – Correction

On cherche le plus petit entier N à partir duquel la suite (u_n) prend des valeurs supérieures ou égales à 10. On considère le programme ci-dessous :

```
1 def q5():
2     N = 0
3     U = 1
4     while U < 10:
5         U = U*2
6     return N
```

Pour quelle(s) raison(s) ce programme ne répond-il pas à la question ?

Ce programme ne convient pas car N n'est pas incrémenté à chaque tour de boucle.

Question (6) – Correction

Compléter le programme suivant afin qu'il détermine le plus petit entier n tel que $u_n \geq 85$.

```
1 def q6():
2     N = 0
3     U = 65
4     while U < 85:
5         N = N+1
6         U = U*0.8+18
7     return N
```

Question (7) – Correction

Déterminer le programme qui permet de calculer le plus petit entier à partir duquel A dépasse 0,5.

```
1 def q7_algo1():
2     A = 0.3
3     N = 0
4     while A <= 0.5:
5         A = 0.92*A+0.05
6         N = N+1
7     return N
```

```
1 def q7_algo2():
2     A = 0.3
3     N = 0
4     while A > 0.5:
5         A = 0.92*A+0.05
6         N = N+1
7     return N
```

Le programme correct est le programme 1.

Question (8) – Correction

Que renvoie le programme suivant ?

```
1 def q8():
2     n = 1
3     while n*(1/3)**n > epsilon:
4         n = n+1
5     return n
```

Ce programme renvoie le plus petit entier n à partir duquel $n \left(\frac{1}{3}\right)^n$ devient inférieur ou égal à epsilon.

Question (9) – Correction

Soit (u_n) la suite définie par $u_0 = 5000$ et, pour tout entier $n \geq 0$, $u_{n+1} = 1,02u_n$. Compléter le programme suivant, de tel sorte qu'il renvoie le plus petit entier n à partir duquel u_n dépasse 10 000.

```
1 def q9():
2     n = 0
3     u = 5000
4     while u <= 10000:
5         n = n+1
6         u = 1.02*u
7     return n
```

Question (10) – Correction

Que représente la valeur de la variable n à la fin de l'exécution de ce programme ?

```
1 def q10():
2     n = 0
3     u = 1
4     while u <= 10**6:
5         n = n+1
6         u = 2*u+1
7     return n
```

Cet algorithme renvoie le plus petit entier n à partir duquel la suite définie par $u_0 = 1$ et, pour tout entier $n \geq 1$, $u_{n+1} = 2u_n + 1$ prend des valeurs strictement supérieures à 10^6 .

FIN