

Plaquette 4 - Variations, seuils et algorithmes

Suites arithmétiques et géométriques — Terminale STI2D

Corrections détaillées

Boîte à outils

Sens de variation.

Suite	Croissante si	Décroissante si
arithmétique $u_n = u_0 + nr$	$r > 0$	$r < 0$
géométrique à termes positifs	$q > 1$	$0 < q < 1$

— Une suite géométrique de raison $q = 1$ est **constante**.

Comportement à long terme d'une géométrique à termes positifs.

- $q > 1$: les termes deviennent aussi grands qu'on veut.
- $0 < q < 1$: les termes se rapprochent de 0.

Recherche de seuil, le squelette du programme.

```
def seuil():
    u = premier_terme
    n = 0
    while condition_non_atteinte:
        u = ...      # on passe au terme suivant
        n = n + 1
    return n
```

- La condition porte sur la variable **modifiée** dans la boucle.
- Le compteur n donne le **rang** du premier terme qui franchit le seuil.
- Pour une suite décroissante vers 0, la boucle s'arrête toujours ; pour une suite croissante cherchant un seuil plus bas, elle ne s'arrêterait jamais.

Seuil d'une suite géométrique. Le premier terme et la raison suffisent : on se ramène à une puissance de q , que l'on encadre par essais.

$$u_0 \times q^n > S \Leftrightarrow q^n > \frac{S}{u_0}$$

Contrôle sans programme. On calcule u_{n-1} et u_n à la main pour vérifier l'encadrement du seuil.

⚠ *tant que $u \leq S$ et tant que $u < S$ ne donnent pas le même résultat quand un terme vaut exactement S : on lit l'énoncé (« dépasser » ou « atteindre »).*

Correction de l'exercice 1 - Sens de variation

- $u_n = 40 - 3n$ est arithmétique de raison $r = -3$. On retire 3 à chaque étape, donc la suite est **décroissante**.
- $v_n = 250 \times 1,07^n$ est géométrique, à termes positifs, de raison $q = 1,07 > 1$: chaque terme est plus grand que le précédent, la suite est **croissante** (hausse de 7 % par étape).
- $w_n = 900 \times 0,88^n$ est géométrique, à termes positifs, de raison $0 < q = 0,88 < 1$: la suite est **décroissante** (baisse de 12 % par étape) et ses termes se rapprochent de 0.

Pour une suite géométrique à termes positifs, tout se lit sur la raison : croissante si $q > 1$,

décroissante si $0 < q < 1$.

Réponse. a) décroissante · b) croissante · c) décroissante.

Correction de l'exercice 2 - Croissance linéaire ou exponentielle

a) Dans le scénario A, on ajoute chaque année le même nombre : suite arithmétique de raison 150, $a_n = 2\,000 + 150n$, croissante car $150 > 0$. Dans le scénario B, gagner 6 % revient à multiplier par 1,06 : suite géométrique à termes positifs, $b_n = 2\,000 \times 1,06^n$, croissante car $1,06 > 1$.

b) Une suite arithmétique traduit une **croissance linéaire** (même gain chaque année) : c'est A. Une suite géométrique traduit une **croissance exponentielle** (même pourcentage chaque année, donc un gain de plus en plus grand) : c'est B.

c) Au bout de 10 ans, $a_{10} = 3\,500$ et $b_{10} = 2\,000 \times 1,06^{10} \approx 3\,582$: les deux scénarios sont proches. Au bout de 20 ans :

$$a_{20} = 5\,000 \quad b_{20} = 2\,000 \times 1,06^{20} \approx 6\,414$$

L'écart s'est creusé : la première année, B ne gagne que 120 habitants, mais son gain annuel grandit sans cesse, alors que celui de A reste à 150.

Réponse. a) $a_n = 2\,000 + 150n$ et $b_n = 2\,000 \times 1,06^n$, toutes deux croissantes · b) A linéaire, B exponentielle · c) 3 500 contre environ 3 582, puis 5 000 contre environ 6 414.

Correction de l'exercice 3 - Un premier seuil arithmétique

La production augmente d'un montant fixe : la suite est arithmétique, $u_n = 450 + 35n$. On cherche le premier rang où elle dépasse 1 000 :

$$450 + 35n > 1\,000 \Leftrightarrow 35n > 550 \Leftrightarrow n > 15,71\dots$$

Le premier entier qui convient est $n = 16$. **Contrôle** : $u_{15} = 450 + 525 = 975$, encore en dessous, et $u_{16} = 450 + 560 = 1\,010$, au-dessus. Réponse : au bout de 16 **mois**.

Réponse. Au bout de 16 mois.

Correction de l'exercice 4 - Un seuil géométrique croissant

Le capital forme une suite géométrique, $u_n = 8\,000 \times 1,05^n$. Il dépasse 12 000 € lorsque

$$8\,000 \times 1,05^n > 12\,000 \Leftrightarrow 1,05^n > 1,5$$

On procède par essais à la calculatrice. Pour $n = 8$, $1,05^8 \approx 1,4775$: le capital vaut environ 11 820 €, c'est insuffisant. Pour $n = 9$, $1,05^9 \approx 1,5513$: le capital vaut environ 12 411 €.

La suite étant croissante, le seuil reste dépassé ensuite. Réponse : au bout de 9 **ans**.

Réponse. Au bout de 9 ans.

Correction de l'exercice 5 - Un seuil décroissant

Une baisse de 12 % revient à multiplier par 0,88 : $u_n = 5\,000 \times 0,88^n$. Le stock passe sous 2 000 pièces lorsque

$$5\,000 \times 0,88^n < 2\,000 \Leftrightarrow 0,88^n < 0,4$$

Pour $n = 7$, $0,88^7 \approx 0,4087$: il reste environ 2 044 pièces, encore au-dessus. Pour $n = 8$, $0,88^8 \approx 0,3596$: il reste environ 1 798 pièces.

La suite est décroissante, donc le stock reste ensuite sous le seuil. Réponse : au bout de 8 **mois**.

Réponse. Au bout de 8 mois.

Correction de l'exercice 6 - Dérouler un algorithme de seuil

On déroule la boucle à la main. Au départ, $u = 200$ et $n = 0$; à chaque tour, u est multiplié par 1,2 et n augmente de 1.

n	1	2	3	4	5
u	240	288	345,6	414,72	497,66

Après le cinquième tour, $u \approx 497,66$ est encore inférieur à 500 : la condition est vraie, la boucle continue. Au sixième tour, $u \approx 597,20$, qui n'est plus inférieur à 500 : la boucle s'arrête et le programme renvoie

$$n = 6$$

Contrôle : $200 \times 1,2^5 \approx 497,66 < 500$ et $200 \times 1,2^6 \approx 597,2 > 500$. Il faut six étapes pour dépasser 500.

Réponse. 6.

Correction de l'exercice 7 - Écrire un algorithme de seuil

Le stock est multiplié par 0,88 chaque mois. On répète cette multiplication **tant que** le stock n'est pas passé sous le seuil, c'est-à-dire tant que $u \geq 2\,000$, en comptant les mois dans n :

```
def seuil():
    u = 5000
    n = 0
    while u >= 2000:
        u = u * 0.88
        n = n + 1
    return n
```

La boucle s'arrête forcément, car la suite décroît vers 0. Elle renvoie 8, comme le calcul direct : $5\,000 \times 0,88^7 \approx 2\,044$ et $5\,000 \times 0,88^8 \approx 1\,798$.

Écrire `while u > 2000` ne changerait le résultat que si un terme valait exactement 2 000, ce qui n'arrive pas ici.

Réponse. Une boucle `while u >= 2000` avec $u = u * 0.88$, qui renvoie 8.

Correction de l'exercice 8 - Compléter un algorithme

Ligne 1. La variable u contient la production ; au départ, elle vaut 450 : $u = 450$.

Ligne 2. On veut continuer tant que le seuil n'est **pas encore dépassé**, c'est-à-dire tant que $u \leq 1\,000$: `while u <= 1000:`. Comme aucun terme ne vaut exactement 1 000, `while u < 1000:` donnerait le même résultat.

Ligne 3. La production augmente de 35 pièces par mois, la suite est arithmétique : $u = u + 35$. Écrire $u = u * 35$ serait une erreur, car cela multiplierait la production.

La fonction renvoie 16. **Contrôle :** $450 + 15 \times 35 = 975$ et $450 + 16 \times 35 = 1\,010$, comme dans le calcul direct.

Réponse. $u = 450$ · `while u <= 1000:` · $u = u + 35$.

Correction de l'exercice 9 – Une boucle qui ne s'arrête pas

La boucle tourne **tant que** $u > 500$. Au départ, $u = 3\,000$, la condition est vraie. À chaque tour, u est multiplié par 1,05 : la suite **croît**, u s'éloigne de 500 au lieu de s'en approcher. La condition reste vraie indéfiniment et le programme ne s'arrête jamais.

Deux corrections sont possibles, selon ce qu'on cherche :

- chercher un seuil **au-dessus** du départ, par exemple *while* $u < 5000$;
- faire décroître u , par exemple $u = u * 0.95$, pour une baisse de 5 %.

À retenir : une suite croissante ne peut franchir qu'un seuil situé au-dessus de son départ, une suite décroissante un seuil situé en dessous. Ajouter une sécurité comme *and* $n < 10000$ évite de bloquer la machine.

Réponse. La suite croît alors que la condition attend une baisse : il faut chercher un seuil supérieur, ou faire décroître u .

Correction de l'exercice 10 – Comparer deux seuils

Le capital double lorsque $10\,000 \times (1 + t)^n \geq 20\,000$, c'est-à-dire lorsque le coefficient multiplicateur $(1 + t)^n$ dépasse 2. Le capital de départ n'intervient pas.

À 3 %. $1,03^{23} \approx 1,9736$ est encore inférieur à 2, alors que $1,03^{24} \approx 2,0328$ le dépasse : il faut 24 ans.

À 6 %. $1,06^{11} \approx 1,8983$ et $1,06^{12} \approx 2,0122$: il faut 12 ans.

Doubler le taux divise ici la durée par deux. Cela rejoint la « règle de 72 » des financiers : la durée de doublement vaut environ $\frac{72}{\text{taux}}$, soit $\frac{72}{3} = 24$ et $\frac{72}{6} = 12$. Réponses : 24 **ans** et 12 **ans**.

Réponse. 24 ans à 3 %, 12 ans à 6 %.

Correction de l'exercice 11 – Un seuil sur une somme

La production du mois n est arithmétique : $u_n = 500 + 30(n - 1)$. La production cumulée sur n mois est la somme des n premiers termes, soit le nombre de termes fois la moyenne des extrêmes :

$$S_n = n \times \frac{500 + u_n}{2}$$

On procède par essais. Pour $n = 14$, $u_{14} = 890$ et $S_{14} = 14 \times 695 = 9\,730$ pièces, c'est insuffisant. Pour $n = 15$, $u_{15} = 920$ et $S_{15} = 15 \times 710 = 10\,650$ pièces.

Le cumul augmente à chaque mois, donc le seuil reste dépassé ensuite. Réponse : au bout de 15 **mois**.

Réponse. Au bout de 15 mois.

Correction de l'exercice 12 – Un algorithme de cumul

Il faut deux variables : u pour la production du mois et s pour le cumul, qui vaut 0 avant le premier mois. À chaque tour, on ajoute la production du mois au cumul, **puis** on passe au mois suivant :

```
def seuil():
    u = 500
    s = 0
    n = 0
    while s <= 10000:
```

```

s = s + u
u = u + 30
n = n + 1
return n

```

La condition porte sur s , la quantité qu'on compare à 10 000. Après 14 tours, $s = 9\,730$; après 15 tours, $s = 10\,650$: la fonction renvoie 15, comme le calcul de l'exercice précédent.

Si l'on inversait les deux lignes de la boucle, on cumulerait 530 dès le premier mois au lieu de 500, et le résultat serait faux.

Réponse. Une boucle `while s <= 10000` cumulant u puis l'augmentant de 30 ; elle renvoie 15.

Correction de l'exercice 13 - Somme des carrés en Python

a) La variable s est l'**accumulateur** : elle vaut 0 au départ et, à chaque tour, on lui ajoute k^2 . Le **compteur** k prend les valeurs entières de 1 à n (`range(1, n + 1)` s'arrête juste avant $n + 1$). La fonction calcule donc

$$\sum_{k=1}^n k^2 = 1^2 + 2^2 + \dots + n^2$$

Les bornes de la somme correspondent aux valeurs du compteur, et le terme k^2 à ce qu'on ajoute dans la boucle.

b) `somme(4)` renvoie $1 + 4 + 9 + 16 = 30$. `somme(10)` renvoie $1 + 4 + 9 + \dots + 100 = 385$; on peut le vérifier avec la formule $\frac{n(n+1)(2n+1)}{6} = \frac{10 \times 11 \times 21}{6} = 385$.

c) Il suffit de remplacer la ligne de la boucle par $s = s + k**3$. Par exemple, la somme des cinq premiers cubes vaut $1 + 8 + 27 + 64 + 125 = 225$.

Réponse. a) $\sum_{k=1}^n k^2$ · b) 30, puis 385 · c) remplacer $k**2$ par $k**3$.

Correction de l'exercice 14 - Un seuil impossible

a) Baisser de 20 % revient à multiplier par 0,8 : les productions annuelles forment une suite géométrique de premier terme 1 000 et de raison 0,8. Le cumul des n premières années est une somme de n termes :

$$S_n = 1\,000 \times \frac{1 - 0,8^n}{1 - 0,8} = 5\,000 \times (1 - 0,8^n)$$

b) Le nombre $0,8^n$ est toujours strictement positif, donc $1 - 0,8^n < 1$ et $S_n < 5\,000$ pour tout n . Le cumul augmente ($S_5 \approx 3\,362$, $S_{10} \approx 4\,463$, $S_{20} \approx 4\,942$) mais ne dépasse jamais 5 000 kWh, donc **jamais** 6 000 kWh.

c) La condition $s \leq 6000$ reste toujours vraie : la boucle tournerait indéfiniment. Avant de lancer un algorithme de seuil, il faut donc vérifier que le seuil peut être atteint, ou ajouter une sécurité comme `and n < 1000`.

Réponse. a) $S_n = 5\,000 \times (1 - 0,8^n)$ · b) jamais : le cumul reste sous 5 000 kWh · c) il ne s'arrêterait jamais.

Correction de l'exercice 15 - Deux suites qui se croisent

On calcule quelques termes pour situer le croisement. Au départ, $u_0 = 2\,000$ dépasse $v_0 = 1\,200$. Au rang 10, $u_{10} = 3\,500$ et $v_{10} \approx 2\,591$; au rang 15, $u_{15} = 4\,250$ et $v_{15} \approx 3\,807$: u est toujours devant, mais l'écart se réduit.

On affine autour du rang 18 :

$$u_{17} = 4\,550 > v_{17} \approx 4\,440 \quad u_{18} = 4\,700 < v_{18} \approx 4\,795$$

Au-delà, l'écart en faveur de v ne fait que grandir, car une croissance en pourcentage finit toujours par l'emporter sur une croissance fixe. Réponse : v dépasse u à partir du **rang 18**.

Réponse. À partir du rang 18.

Correction de l'exercice 16 - Synthèse : réduire les déchets d'un atelier

a) Réduire de 8 % revient à multiplier par 0,92 : la suite est géométrique de raison 0,92. Pour aller de l'année 1 à l'année n , on applique $n - 1$ baisses, donc $u_n = 12 \times 0,92^{n-1}$. Les termes sont positifs et $0 < 0,92 < 1$: la suite est **décroissante**.

b) On calcule les termes autour du seuil : $u_5 = 12 \times 0,92^4 \approx 8,60$ tonnes, encore au-dessus, puis $u_6 = 12 \times 0,92^5 \approx 7,91$ tonnes. La suite étant décroissante, l'atelier reste ensuite sous les 8 tonnes : c'est à partir de la **sixième année**.

c) On part de l'année 1 et on applique la baisse tant que la masse reste d'au moins 8 tonnes :

```
def annee():
    u = 12
    n = 1
    while u >= 8:
        u = u * 0.92
        n = n + 1
    return n
```

La fonction renvoie 6. Attention à l'initialisation : le compteur part de 1, puisque $u_1 = 12$.

d) La masse totale est la somme des 10 premiers termes d'une suite géométrique :

$$u_1 + \dots + u_{10} = 12 \times \frac{1 - 0,92^{10}}{1 - 0,92} \approx 84,84 \text{ tonnes}$$

Sans effort de réduction, l'atelier aurait produit 120 tonnes : il en évite environ 35.

Réponse. a) $u_n = 12 \times 0,92^{n-1}$, décroissante · b) la sixième année · c) boucle *while* $u \geq 8$ · d) environ 84,84 tonnes.