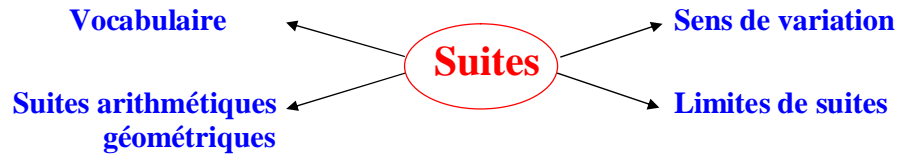


Généralités sur les suites



Il s'agit d'un chapitre sans formules.

On donne :

- des définitions et du vocabulaire ;
- des notations ;
- des techniques.

On effectue des changements de cadres fréquents.

Plan du chapitre :

I. Présentation du chapitre

II. Exemples introductifs

III. Définition et vocabulaire

IV. Différentes manières de définir une suite

V. Liens entre suites et fonctions

VI. Représentations graphiques

VII. Calculs d'indices

VIII. Sens de variation d'une suite

IX. Le « raisonnement de proche en proche »

X. Calcul d'un terme de rang donné d'une suite définie par récurrence

XI. Conjectures

I. Présentation du chapitre

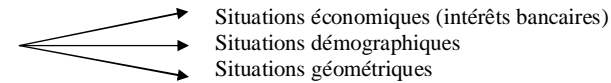
1°) Difficultés du chapitre

- le vocabulaire
- les notations
- les formules
- le calcul algébrique

Chapitre assez abstrait au début.

2°) Quelques utilisations des suites

Modélisation de situations :



On étudiera des phénomènes chronologiques.

Il y a beaucoup d'autres utilisations qui seront vues en exercices.

3°) Lien avec les fonctions

Les suites ressemblent beaucoup aux fonctions mais avec des spécificités propres.

II. Exemples introductifs

1°) Exemple 1 (suites logiques)

On considère les listes de nombres suivants construits sur un principe logique.

L_1	1 ; 4 ; 7 ; 10 ; 13 ; 16 ; 19 ; 22	On ajoute 3 à chaque fois.
L_2	2 ; 4 ; 8 ; 16 ; 32 ; 64 ; 128 ; 256	On multiplie par 2 à chaque fois.
L_3	15 ; 11 ; 7 ; 3 ; -1 ; -5 ; -9 ; -13	On enlève 4 à chaque fois.
L_4	40 ; 20 ; 10 ; 5 ; 2,5	On divise par 2 à chaque fois.
L_5	1 ; 4 ; 9 ; 16 ; 25 ; 36 ; 49 ; 64	Carrés parfaits
L_6	3 ; 4 ; 6 ; 9 ; 13 ; 18 ; 24 ; 31	+ 1, + 2, + 3, + 4, + 5...

Il s'agit d'un procédé de **construction algorithmique**.

La suite donnée dans L_1 est une suite arithmétique de raison 3.

La suite donnée dans L_2 est une suite géométrique de raison 2

On adopte une notation.

On désigne par :

u_0 le premier terme de la suite

u_1 le deuxième terme de la suite

u_2 le troisième terme de la suite

etc...

(notation indicielle ou indexée)

C'est une première bizarrerie que l'on observe sur les suites : u_0 est le premier terme de la suite, u_1 le deuxième terme de la suite, u_2 le troisième terme de la suite etc.

Dans la liste L_1 , $u_3 = 10$.

Dans la liste L_5 , $u_4 = 25$.

Les suites arithmétiques correspondent au procédé logique qui consiste à additionner ou soustraire un même nombre.

Les suites géométriques correspondent au procédé logique qui consiste à multiplier ou diviser par un même nombre.

Dans le cadre de l'étude d'une suite, utiliser le registre de la langue naturelle, le registre algébrique, le registre graphique, et passer de l'un à l'autre.

2°) Exemple 2 (phénomène chronologique)

L'étude d'une population fait apparaître une augmentation de 1 % par mois à partir du 1^{er} janvier 2000. On se propose d'étudier l'évolution de cette population chaque mois sachant que cette population est de 100 000 individus le 1^{er} janvier 2000.

Population au 1^{er} février 2000 : $100\,000 \times \boxed{1,01} = 101\,000$

Population au 1^{er} mars 2000 : $101\,000 \times \boxed{1,01} = 102\,010$

On note P_0 la population au 1^{er} janvier 2000

P_1 la population au 1^{er} février 2000

P_2 la population au 1^{er} mars 2000

etc.

Exemples :

La population au 1^{er} mars 2001 est donc P_{14} .

P_{25} désigne la population au 1^{er} février 2002.

$$P_{n+1} = P_n \times \boxed{1,01}$$

III. Définition et vocabulaire

1°) Définition mathématique d'une suite

Une **suite numérique** est une fonction u qui à tout entier naturel n associe un nombre réel noté $u(n)$ ou u_n (« u indice n »)
 $u : \mathbb{N} \rightarrow \mathbb{R}$
 $n \mapsto u_n$

Rappel : une fonction $f : \mathbb{R} \longrightarrow \mathbb{R}$

2°) Notations et vocabulaire

- La suite u est souvent notée $(u_n)_{n \in \mathbb{N}}$.
- Les images sont notées u_0, u_1, u_2 au lieu de $u(0), u(1), u(2) \dots$
- Ce sont les **termes** de la suite.

Attention : u_n désigne le terme d'indice n .

$u(n) : "u \text{ de } n"$
 $u_n : "u \text{ indice } n"$ } **même chose**

$u = (u_n)_{n \in \mathbb{N}}$ (les parenthèses sont obligatoires)

Quand il n'y a pas de parenthèses, c'est pour définir le nombre.
Quand il y a des parenthèses, c'est pour définir la fonction.

3°) Exemple

$\begin{cases} u \\ (u_n) \end{cases}$ est la suite définie par $\forall n \in \mathbb{N} \quad u_n = n^2 + 2n - 3$

$u : \mathbb{N} \rightarrow \mathbb{R}$

$$n \mapsto n^2 + 2n - 3$$

Calculons les trois premiers termes.

$$u_0 = u(0) = 0^2 + 2 \times 0 - 3 = -3$$

$$u_1 = u(1) = 0$$

$$u_2 = u(2) = 5$$

4°) Remarques sur les indices

u_n : terme d'indice n
(n est un entier naturel, u_n est un réel)

u_{n+1} : terme d'indice $n + 1$, **terme suivant**

u_{n-1} : terme d'indice $n - 1$, **terme précédent** ($n \geq 1$)

On dira que u_{n-1} , u_n , u_{n+1} sont des **termes consécutifs** (consécutif = consécutif, mot du langage courant qui signifie « qui se suivent ») de la suite.

De manière générale, on appelle termes consécutifs d'une suite des termes dont les indices sont des entiers naturels consécutifs).

Exemple :

3, 4, 5, 6, 7 sont des entiers consécutifs.
 u_3, u_4, u_5, u_6, u_7 sont des termes consécutifs.

Attention :

Pour une suite qui commence par l'indice 0
est définie à partir de l'indice 0

- le 100^e terme est u_{99}
- le 1000^e terme est u_{999}

Récapitulatif :

- suite : $u, (u_n)_{n \in \mathbb{N}}$ (2 notations)
- indice
- terme

Successeur d'un entier :

$n + 1$: **successeur** de n

(à voir comme une fonction non comme une addition)

$$\begin{aligned} n &\rightarrow n+1 \\ n+1 &\rightarrow n+2 \\ n-1 &\rightarrow n \\ n-2 &\rightarrow n-1 \end{aligned}$$

$(u_n), (u_{n-1}), (u_{n+1})$: expressions globales

5°) Nombre d'éléments d'une liste d'entiers consécutifs

• Propriété [Nombre d'entiers relatifs compris entre deux entiers au sens large]

Soit n et p deux entiers relatifs tels que $p \leq n$.
Le nombre d'entiers relatifs k tels $p \leq k \leq n$ est égal à $n - p + 1$.

• Exemple

Le nombre d'entiers relatifs k tels $-3 \leq k \leq 4$ est égal à $14 (10 - (-3) + 1)$.

• Application

La liste u_3, u_4, \dots, u_{20} comprend 18 termes.

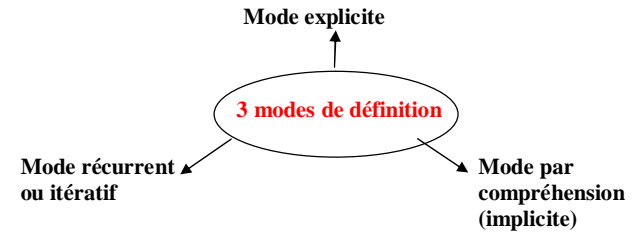
• Mise en garde

La liste u_0, u_1, \dots, u_n comprend $n+1$ termes.

La liste u_1, \dots, u_n comprend n termes.

IV. Différentes manières de définir une suite

Dans ce paragraphe, nous allons étudier les différents modes de génération d'une suite.
Il y en a trois principaux.



1°) Par une formule explicite : mode explicite ou fonctionnel

• Principe

On définit une suite $\begin{cases} u \\ (u_n) \end{cases}$ de façon explicite lorsque l'on exprime le terme général u_n en fonction de l'indice n .

(Même chose que pour les fonctions où l'on définit $f(x)$ en fonction de x).

• **Exemple 1**

$\begin{matrix} u \\ (u_n) \end{matrix}$ est la suite définie par $\forall n \in \mathbb{N} \quad u_n = n^2 + 2n - 3$.

Repasser en rouge le n qui est en indice à gauche et le n qui est dans l'expression à droite.

$$u : \mathbb{N} \rightarrow \mathbb{R}$$

$$n \mapsto n^2 + 2n - 3 \quad (n \text{ est la variable})$$

Calculons les trois premiers termes.

Il s'agit de simples calculs d'images.

$$u_0 = u(0) = 0^2 + 2 \times 0 - 3 = -3$$

$$u_1 = u(1) = 0$$

$$u_2 = u(2) = 5$$

La suite u est définie $\left\{ \begin{array}{l} \text{sur } \mathbb{N} \\ \text{à partir de l'indice } 0 \end{array} \right.$

Quand je choisis une valeur de n (par exemple $n=5$), il s'agit d'une valeur particulière de n (que j'ai choisi).
Je prends autant de valeurs particulières que je veux, 10, 20, ... 5000 ...
En remplaçant dans l'expression de la suite, j'obtiens des termes particuliers u_{10} , u_{20} , u_{5000} .

• **Exemple 2**

$\begin{matrix} u \\ (u_n) \end{matrix}$ est la suite définie par $\forall n \in \mathbb{N}^* \quad u_n = \frac{3-n}{n}$.

La suite u est définie $\left\{ \begin{array}{l} \text{sur } \mathbb{N}^* \\ \text{à partir de l'indice } 1 \end{array} \right.$

$$u_1 = 2$$

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

• **Exemple 3**

$\begin{matrix} u \\ (u_n) \end{matrix}$ est la suite définie par $\forall n \in \mathbb{N} \quad u_n = 5 \times 2^n - 3$.

La suite u est définie $\left\{ \begin{array}{l} \text{sur } \mathbb{N} \\ \text{à partir de l'indice } 0 \end{array} \right.$

$$u_0 = 5 \times 2^0 - 3 = 5 \times 1 - 3 = 2$$

$$u_1 = 5 \times 2^1 - 3 = 5 \times 2 - 3 = 7$$

On rencontrera des suites dont le terme général est exprimé en fonction de n à l'aide de puissances. (Il ne faut pas être choqué : une expression avec des puissances où n est en exposant est bien une expression du terme général en fonction de n).

2°) **Mode par compréhension (implicite)**

• **Exemple 1**

$u_n = n$ -ième décimale de π ($n \geq 1$)

$\pi = 3,141592654\dots$ (développement décimal de π)
La suite u est définie sur \mathbb{N}^* (à partir de l'indice 1).

$$u_1 = 1^{\text{ème}} \text{ décimale de } \pi = 1$$

$$u_2 = 2^{\text{e}} \text{ décimale de } \pi = 4$$

$$u_3 = 1$$

La recherche des décimales de π a été l'objet de nombreuses recherches en mathématiques et le reste même en recherche actuelle. On aimerait notamment trouver une formule donnant la n -ième décimale de π . Une telle formule a été trouvée pour la base deux en 1994. Elle s'appelle formule de Bailey-Borwyn-Plouffe. On ne sait pas si une telle formule existe en base dix.

À la place de π , on peut prendre n'importe quel réel.

Tout réel admet un développement décimal propre.

Exemples :

$\frac{1}{3} = 0,3333\dots$ (le 3 souligné indique qu'il se répète indéfiniment)
On a une suite constante (tous les termes sont égaux à 3).

$$\frac{1}{6} = 0,16666\dots \quad u_1=1, u_2=6, u_3=6$$

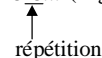
La suite u est stationnaire à partir de l'indice 2 (constante à partir d'un certain indice).

La suite des décimales d'un réel est un exemple fondamental de suite.

La suite des décimales d'un rationnel est toujours périodique. C'est même une caractérisation des nombres rationnels.

Exercice :

On pose $x = 3,151515\dots$ (le groupe 15 se répète indéfiniment).



Déterminer la valeur exacte de x .

On calcule $100x = 315,1515\dots$ puis $100x - x = 312$ soit $99x = 312$ d'où $x = \frac{312}{99} = \frac{104}{33}$.

On vérifie avec la calculatrice (commande pour passer d'une écriture décimale à une écriture fractionnaire).

● Exemple 2

$u_n = n$ -ième nombre pair ($n \geq 1$)

$$u_1 = 0$$

$$u_2 = 2$$

$$u_3 = 4$$

● Exemple 3

$u_n = n$ -ième nombre impair ($n \geq 1$)

$$u_1 = 1$$

$$u_2 = 3$$

$$u_3 = 5$$

● Exemple 4

$u_n =$ salaire au bout de n mois (voir exercices plus tard)

● Exemple 5

Pour tout entier naturel n , on pose $u_n =$ somme des entiers naturels inférieurs ou égaux à n .

$u_0 =$ somme des entiers naturels inférieurs ou égaux à $0 = 0$

$u_1 =$ somme des entiers naturels inférieurs ou égaux à $1 = 0 + 1 = 1$

$u_2 =$ somme des entiers naturels inférieurs ou égaux à $2 = 0 + 1 + 2 = 3$

On peut écrire $u_n = \sum_{k=0}^{k=n} k$.

Nous verrons plus tard qu'il est possible de trouver une formule donnant u_n en fonction de n .

● Exemple 6

Pour tout entier naturel $n \geq 1$, on pose $u_n =$ produit de tous les entiers naturels non nuls inférieurs ou égaux à n .

$u_1 =$ produit des entiers naturels non nuls inférieurs ou égaux à $1 = 1$

$u_2 =$ produit des entiers naturels non nuls inférieurs ou égaux à $2 = 1 \times 2 = 2$

$u_3 =$ produit des entiers naturels non nuls inférieurs ou égaux à $3 = 1 \times 2 \times 3 = 6$

Pour tout entier naturel $n \geq 1$, u_n est égal à la factorielle de n .

La **factorielle** d'un entier naturel n est noté $n!$ et est définie de la manière suivante :

- pour $n \geq 2$, $n!$ est égal au produit de tous les entiers de 1 à n c'est-à-dire $n! = 1 \times 2 \times \dots \times n$;
- $1! = 1$ et $0! = 1$ par convention.

On a ainsi par exemple $3! = 1 \times 2 \times 3 = 6$, $4! = 1 \times 2 \times 3 \times 4 = 24$ (on peut d'ailleurs observer que $4! = 3! \times 4$) etc.

Écrire un programme Python qui permet de calculer la factorielle d'un entier naturel quelconque.

Écrire un programme qui calcule la factorielle.

```
def fact(n):  
    """fact(n): calcule la factorielle de n (entier >= 0)"""  
    x=1  
    for i in xrange(2,n+1):  
        x*=i  
    return x
```

3°) Mode récurrent ou itératif

● La commande « rép » de la calculatrice

● Principe

On définit une suite $\left(u_n \right)$ de façon **récurrente** lorsque l'on donne une formule permettant de calculer chaque terme en fonction du ou des précédents.

On définit une suite $\left(u_n \right)$ de façon **récurrente** lorsque l'on exprime le terme u_n (d'indice n) en fonction du précédent u_{n-1} (d'indice $n-1$).

(Nouvelle définition par rapport aux fonctions)

$$\begin{cases} u_0 \\ u_n = f(u_{n-1}) \text{ ou } u_{n+1} = f(u_n) \end{cases}$$

$$\begin{matrix} f & f & f & f \\ u_0 \mapsto u_1 \mapsto u_2 \mapsto u_3 \mapsto u_4 \mapsto \dots \end{matrix}$$

• Exemples

Exemple 1 :

$\begin{pmatrix} u \\ (u_n) \end{pmatrix}$ est la suite définie sur \mathbb{N} par :

$u_0 = 4$ (1^{er} terme)

et

la relation de récurrence : $\forall n \in \mathbb{N} \quad u_{n+1} = 2u_n - 3$.

Le premier terme et la relation générale de récurrence sont donnés dans l'énoncé.

La relation de récurrence est générale (valable pour tout entier naturel).

Le 25-4-2020

$u_{n+1} = 2u_n - 3$ (j'avais noté $u_{n+1} = 2u_n - 3$)

Il s'agit d'une égalité orientée.

Recopier en rouge le n en indice à gauche et le n en indice à droite. C'est le même « n ».

$u(n+1) = 2u(n) - 3$

On notera que la relation de récurrence peut aussi s'écrire $\forall n \in \mathbb{N}^* \quad u_n = 2u_{n-1} - 3$.

Principe de calcul des termes

Chaque terme, sauf le premier, s'obtient en multipliant le terme précédent par 2 et en retirant 3 au résultat.

Le programme de calcul des termes se résume ainsi : « multiplier par 2 le terme précédent et retirer 3 au résultat ».

Calcul de premiers termes

$u_0 = 4$

$u_1 = u_{0+1} = 2 \times u_0 - 3 = 2 \times 4 - 3 = 5$

$u_2 = u_{1+1} = 2 \times u_1 - 3 = 2 \times 5 - 3 = 7$

$u_3 = u_{2+1} = 2 \times u_2 - 3 = 2 \times 7 - 3 = 11$

Programmes Python :

Sans récursivité Avec boucle For		Avec récursivité
<pre>def terme(n): u=4 for i in range(1, n+1): u=2*u-3 return u</pre>		<pre>def terme(n): if n==0: return 4 else: return 2*terme(n-1)-3</pre>
<pre>def terme(n): u=4 for i in range(0, n): u=2*u-3 return u</pre>		
Affiche n'importe quel terme d'indice n supérieur ou égal à 1		Affiche n'importe quel terme d'indice n .

Exemple 2 :

$\begin{pmatrix} u \\ (u_n) \end{pmatrix}$ est la suite définie sur \mathbb{N} par :

$u_0 = 3$ (1^{er} terme)

et

la relation de récurrence : $\forall n \in \mathbb{N} \quad u_{n+1} = 2u_n + n - 1$.

$u_1 = 2u_0 + 0 - 1 = 2 \times 3 - 1 = 5$

$u_2 = 2u_1 + 1 - 1 = 2 \times 5 = 10$

Programme Python :

Pour la programmation, on écrit la relation de récurrence sous la forme

$\forall n \in \mathbb{N}^* \quad u_n = 2u_{n-1} + n - 1 - 1 = 2u_{n-1} + n - 2$.

```
def terme(n):
    u=3
    for i in range(1, n+1):
        u=2*u+i-2
    return u
```

Exemple 3 :

$\begin{cases} u \\ (u_n) \end{cases}$ est la suite définie sur \mathbb{N} par :

$$u_0 = -1 \text{ (1}^{\text{er}} \text{ terme)}$$

$$u_1 = 3 \text{ (2}^{\text{e}} \text{ terme)}$$

la relation de récurrence : $\forall n \in \mathbb{N} \quad u_{n+2} = 2u_{n+1} + u_n$.

Il s'agit d'une relation de récurrence d'ordre 2 (suite récurrente double).

$$u_2 = 2u_1 + u_0 = 2 \times 3 - 1 = 5$$

$$u_3 = 2u_2 + u_1 = 2 \times 5 + 3 = 13$$

Sans récursivité Avec boucle For		Avec récursivité
<pre>def terme(n): a, b = -1, 3 for i in range(1, n): a, b = b, 2*b+a return b</pre>		<pre>def terme(n): if n==0: return -1 elif n==1: return 3 else: return 2*terme(n-1)+terme(n-2)</pre>
Affiche n'importe quel terme d'indice n supérieur ou égal à 2.		Affiche n'importe quel terme d'indice n .

Je laisse tomber :

```
def terme(n):
    v = -1
    u = 3
    for i in range(1, n):
        u, v = 2*u+v, u
    return u
```

```
u = range(0, N)
u[0] = 1
for i in u[2: N]:
    u[i] = u[i-1] + u[i-2]
```

print u

Le but de beaucoup d'exercices est souvent de passer d'un mode récurrent ou par compréhension à un mode explicite.

Le calcul des termes pour une suite définie par récurrence peut être automatisé sur ordinateur en utilisant un tableur.

4°) Autres moyens de définir une suite

- par un algorithme

- par somme ou produit

V. Liens entre suites et fonctions

1°) Mode explicite

- On peut associer une suite à une fonction.

Principe :

Étant donnée une fonction $f: \mathbb{R}_+ \rightarrow \mathbb{R}$, on peut définir une suite u définie sur \mathbb{N} de manière explicite en posant $u_n = f(n)$ pour tout entier naturel n .

$u_n = f(n)$ est l'image de n par la fonction f .

Exemple « concret » :

On considère la fonction avec $f: x \mapsto 2x + 3$.

La suite associée selon le principe donné dans le cadre précédent est la suite u définie sur \mathbb{N} par $u_n = f(n)$

c'est-à-dire $u_n = 2n + 3$. On a défini ainsi une suite u de manière explicite (puisque l'on a l'expression du terme général).

Plus généralement, étant donnée une fonction $f: E \rightarrow \mathbb{R}$ où E une partie de \mathbb{R} contenant \mathbb{N} , on peut définir une suite u définie sur \mathbb{N} de manière explicite en posant $u_n = f(n)$ pour tout entier naturel n .

Il s'agit de la restriction de f à \mathbb{N} .

Au début, on a pris $E = \mathbb{R}_+$.

- Dans l'autre sens, il est parfois possible d'associer une fonction à une suite définie en mode explicite.

Exemples :

① $\forall n \in \mathbb{N} \quad u_n = 2n^2 + 3n - 5$ (suite définie en mode explicite)

On peut considérer la fonction $f: x \mapsto 2x^2 + 3x - 5$ définie sur \mathbb{R} .

On peut écrire $\forall n \in \mathbb{N} \quad u_n = f(n)$.

② $\forall n \in \mathbb{N} \quad u_n = 2^n$ (suite définie en mode explicite)

On peut considérer la fonction $f: x \mapsto 2^x$ définie sur \mathbb{R} [cette fonction n'est pas connue en 1^{ère}].

On peut écrire $\forall n \in \mathbb{N} \quad u_n = f(n)$.

Nous verrons pourquoi il est intéressant d'associer une fonction à une suite définie en mode explicite.

2°) Mode récurrent

• On peut associer une suite récurrente à une fonction.

Principe :

Étant donnée une fonction $f: \mathbb{R} \rightarrow \mathbb{R}$, on peut définir une suite u sur \mathbb{N} de manière récurrente par la valeur de u_0 et telle que chaque terme, sauf le premier, soit égal à l'image du précédent par f .

Le procédé algorithmique sous jacent permet de calculer ainsi tous les termes de la suite de proche en proche.

On a ainsi :

$$u_1 = f(u_0) = \text{image de } u_0 \text{ par } f$$

$$u_2 = f(u_1) = \text{image de } u_1 \text{ par } f$$

etc.

On peut écrire $\forall n \in \mathbb{N} \quad u_{n+1} = f(u_n)$.

L'égalité quantifiée « $\forall n \in \mathbb{N} \quad u_{n+1} = f(u_n)$ » est appelée relation de récurrence.

Elle peut aussi s'écrire sous la forme « $\forall n \in \mathbb{N}^+ \quad u_n = f(u_{n-1})$ » ou « $u_n = f(u_{n-1})$ pour tout entier naturel $n \geq 1$ ».

Chaque terme de la suite, sauf le premier, est l'image du précédent par la fonction f .

Exemple « concret » :

$$f: x \mapsto 2x+3$$

On peut définir une suite u sur \mathbb{N} en choisissant une valeur pour le premier terme u_0 et par la relation de récurrence « $\forall n \in \mathbb{N} \quad u_{n+1} = f(u_n)$ » soit « $\forall n \in \mathbb{N} \quad u_{n+1} = 2u_n + 3$ ».

Dans le principe, on peut remplacer $f: \mathbb{R} \rightarrow \mathbb{R}$ par $f: D \rightarrow \mathbb{R}$ où D est une partie de \mathbb{R} stable par f c'est-à-dire telle que $\forall x \in D \quad f(x) \in D$. On écrit parfois $f(D) \subset D$.

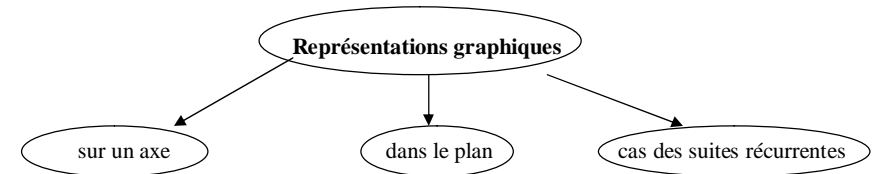
• Dans l'autre sens, il est parfois possible d'associer une fonction à une suite définie en mode récurrent.

$$\begin{cases} u_0 \in \mathbb{R} \\ u_{n+1} = (u_n)^2 - 3u_n + 1 \end{cases} \quad (\text{suite définie en mode récurrent})$$

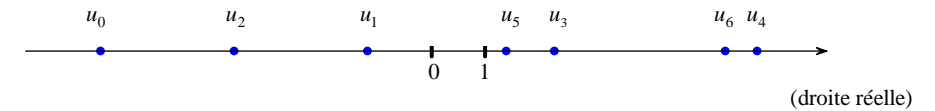
$$u_{n+1} = f(u_n) \text{ avec } f: x \mapsto x^2 - 3x + 1$$

VI. Représentations graphiques

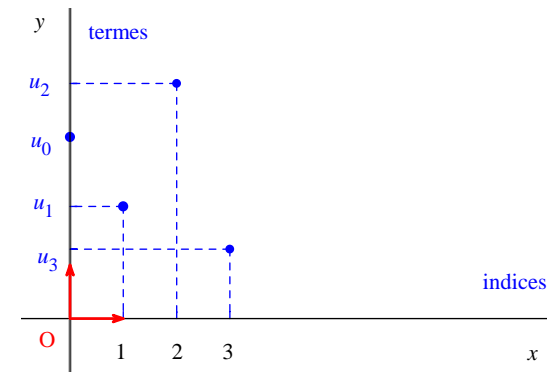
Schéma



1°) Représentation graphique sur un axe gradué



2°) Représentation graphique dans un repère du plan



La suite u est représentée graphiquement par les points d'abscisse entière n et d'ordonnée u_n .

La suite u est représentée par des points isolés (« nuage de points »).

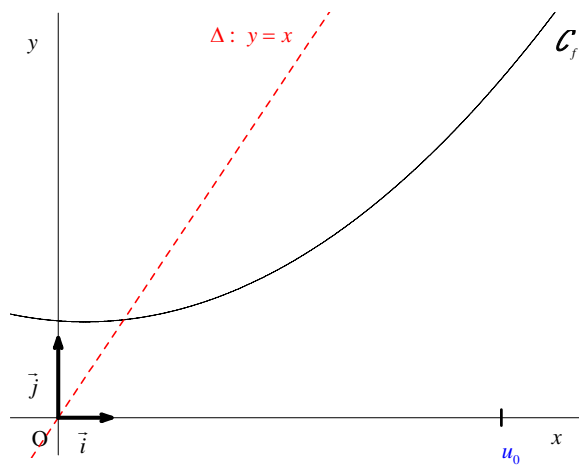
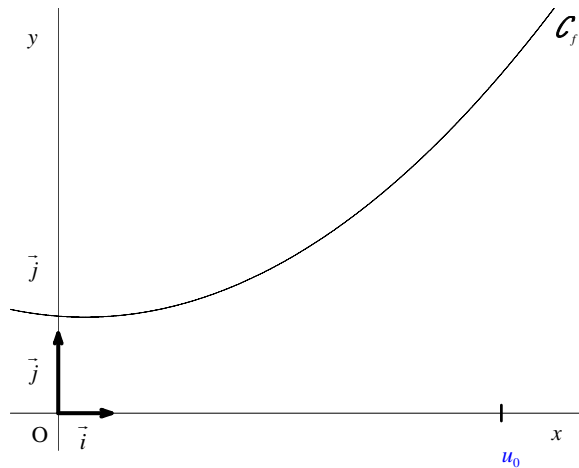
3°) Lecture graphique des termes d'une suite récurrente

f est une fonction.

(u_n) est la suite définie par $\begin{cases} u_0 \text{ donné (terme initial)} \\ \forall n \in \mathbb{N} \quad u_{n+1} = f(u_n) \end{cases}$.

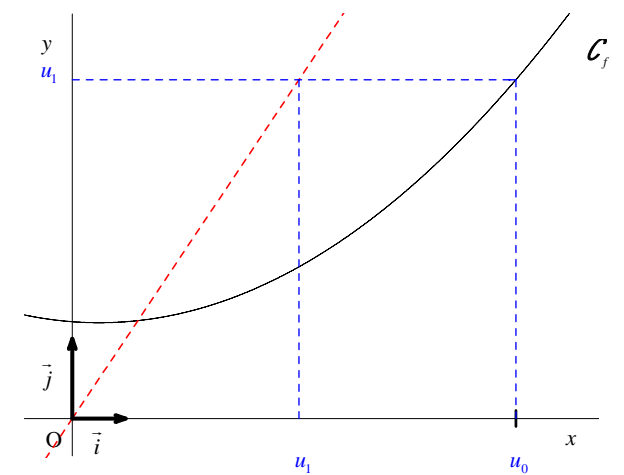
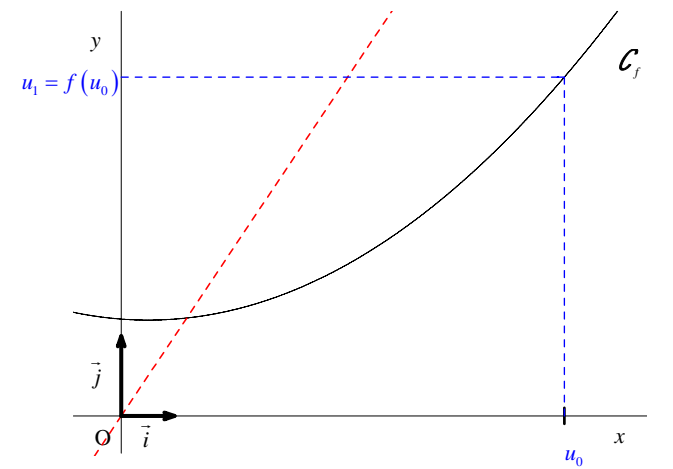
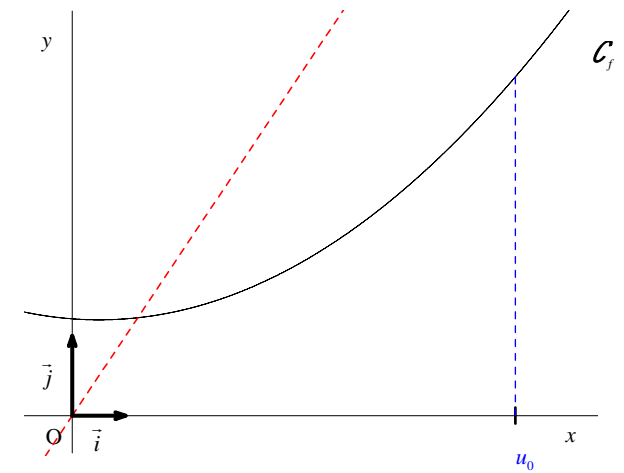
On veut représenter sans calculs les premiers termes de la suite.

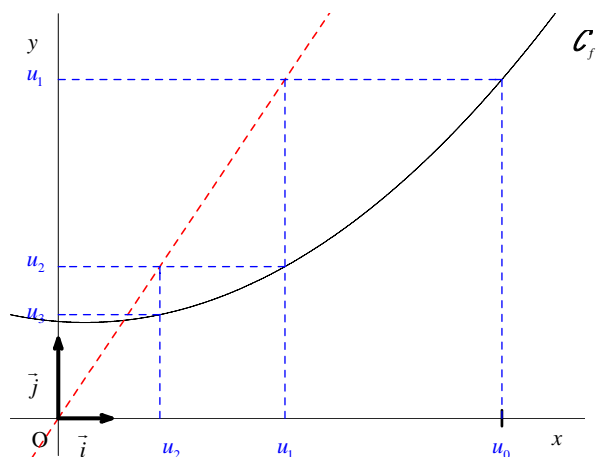
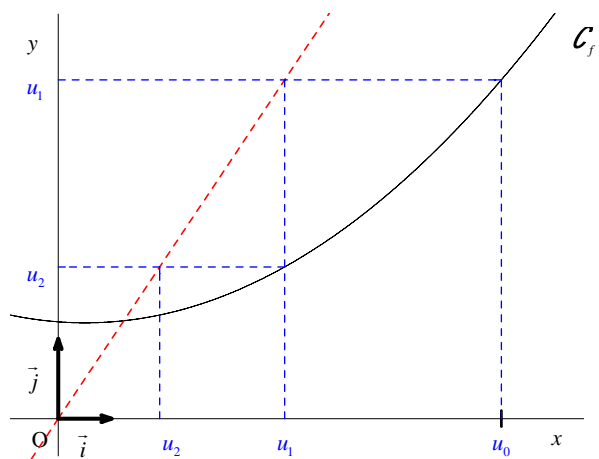
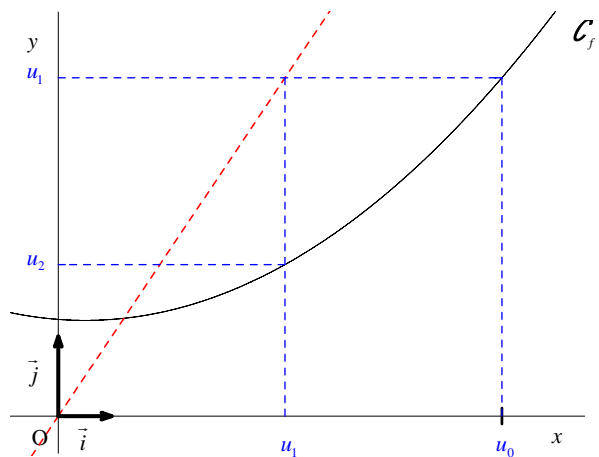
On trace d'abord la courbe représentative \mathcal{C}_f de f dans un repère.



On utilise un système de parallèles à l'axe des abscisses et à l'axe des ordonnées que l'on trace en pointillés.

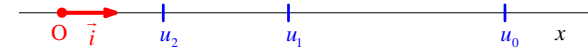
On n'est pas obligé de prolonger les parallèles à l'axe des abscisses jusqu'à l'axe des ordonnées.





On obtient ici une construction en « marches d'escalier ».

À la fin, on oublie l'axe des ordonnées pour ne conserver que l'axe des abscisses.



Méthode :

Dans un repère, on trace :

- la représentation graphique \mathcal{C}_f de f ;
 - la droite Δ d'équation $y = x$ (« 1^{ère} bissectrice » du repère lorsque les vecteurs de base ont la même norme ou « droite à 45° » lorsque le repère est orthonormé).
- En général, on prend le repère orthonormé mais cela n'est pas obligatoire.

Recette :

- On place u_0 sur l'axe des abscisses.
- On « monte » u_0 jusqu'à \mathcal{C}_f .
- On obtient u_1 en ordonnée car $u_1 = f(u_0)$.
- On rallonge jusqu'à Δ .
- On redescend en abscisse ; on obtient la valeur de u_1 .
- On recommence avec u_1 et ainsi de suite.

Il s'agit d'une construction des termes d'une suite récurrente **sans calcul**.

Suivant les cas, on obtient une construction en « marches d'escalier » ou « en spirale » (« en escargot » ou « en colimaçon »).

Cette construction des termes d'une suite récurrente est une **construction itérative**.

gestes associés à la construction

Il est aisé de faire la construction avec le doigt.

Le 4-3-2020

$u_1 =$ image de u_0 par f
 $u_2 =$ image de u_1 par f

Ce qui compte c'est l'axe des abscisses.
 On oublie l'axe des ordonnées pour ne conserver que l'axe des abscisses.
 On ne fait aucun calcul.

La droite Δ est une « droite de report » (report des valeurs de l'axe des ordonnées sur l'axe des abscisses).

Tous les points de la droite Δ ont une abscisse égale à l'ordonnée.

Δ passe par l'origine du repère et le point de coordonnées (1 ; 1).

VII. Calculs d'indices

1°) Suites définies par une formule explicite

• Exemple 1

$\begin{cases} u \\ (u_n) \end{cases}$ est la suite définie par $\forall n \in \mathbb{N} \quad u_n = n^2 - 2n + 3$.

$$\begin{aligned} u_{n+1} &= (n+1)^2 - 2(n+1) + 3 \\ &= n^2 + 2n + 1 - 2n - 2 + 3 \\ &= n^2 + 2 \end{aligned}$$

$$\begin{aligned} u_{n-1} &= (n-1)^2 - 2(n-1) + 3 \quad (n \geq 1) \\ &= n^2 - 2n + 1 - 2n + 2 + 3 \\ &= n^2 - 4n + 6 \end{aligned}$$

• Exemple 2

$\begin{cases} u \\ (u_n) \end{cases}$ est la suite définie par $\forall n \in \mathbb{N} \quad u_n = \frac{2n-1}{n+4}$.

$$\forall n \in \mathbb{N} \quad u_{n+1} = \frac{2(n+1)-1}{(n+1)+4}$$

$$\forall n \in \mathbb{N} \quad = \frac{2n+1}{n+5}$$

2°) Suites définies par récurrence

• Exemple 1

$\begin{cases} u \\ (u_n) \end{cases}$ est la suite définie par $\begin{cases} u_0 = 5 \\ \forall n \in \mathbb{N} \quad u_{n+1} = 2u_n - 3 \end{cases}$.

Exprimer u_{n+2} en fonction u_{n+1} .

$$u_{n+2} = 2u_{n+1} - 3$$

• Exemple 2

$\begin{cases} u \\ (u_n) \end{cases}$ est la suite définie par $\begin{cases} u_0 = 2 \\ \forall n \in \mathbb{N} \quad u_{n+1} = 6 - u_n \end{cases}$.

$$\begin{aligned} u_{n+2} &= 6 - u_{n+1} \\ &= 6 - (6 - u_n) \\ &= u_n \end{aligned}$$

Donc u est une suite périodique de période 2.

• Exemple 3

$\begin{cases} u \\ (u_n) \end{cases}$ est la suite définie par $\begin{cases} u_0 = 3 \\ \forall n \in \mathbb{N} \quad u_{n+1} = \frac{5}{u_n} \end{cases}$.

$$\begin{aligned} u_{n+2} &= \frac{5}{u_{n+1}} \\ &= \frac{5}{\frac{5}{u_n}} \\ &= 5 \times \frac{u_n}{5} \\ &= u_n \end{aligned}$$

Donc u est une suite périodique de période 2.

3°) Suite définie en fonction d'une autre

Exemple :

u est une suite donnée.

v est la suite définie par $\forall n \in \mathbb{N} \quad v_n = u_n + 4$.

$$v_{n+1} = u_{n+1} + 4$$

VIII. Sens de variation d'une suite

On voit encore une fois, au titre, qu'il y a vraiment une ressemblance entre les suites et les fonctions.

1°) Définitions

Définitions en français :

- On dit qu'une suite est **croissante** lorsque chaque terme est inférieur ou égal au suivant.
- On dit qu'une suite est **décroissante** lorsque chaque terme est supérieur ou égal au suivant.
- On dit qu'une suite est **constante** lorsque chaque terme est égal au suivant.

Variante :

On dit qu'une suite est **croissante** lorsque chaque terme, sauf le premier, est supérieur ou égal au précédent.

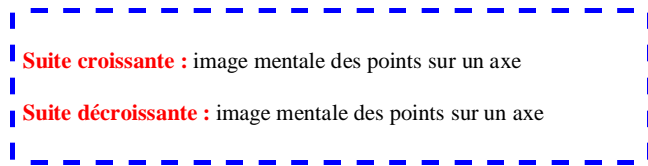
Définitions avec écritures mathématiques :

u est une suite définie sur \mathbb{N} .

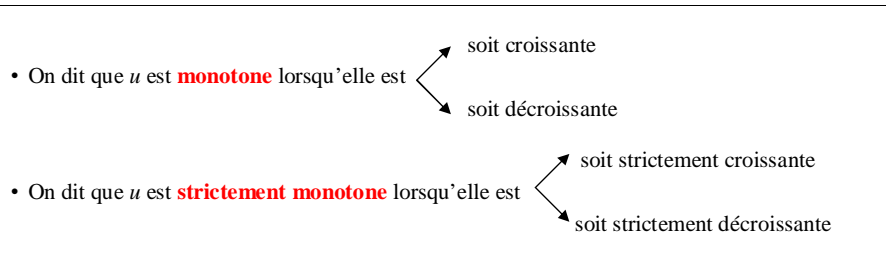
- On dit que u est **croissante** pour exprimer que pour tout entier naturel $n : u_n \leq u_{n+1}$.
- On dit que u est **décroissante** pour exprimer que pour tout entier naturel $n : u_n \geq u_{n+1}$.
- On dit que u est **constante** pour exprimer que pour tout entier naturel $n : u_n = u_{n+1}$.

- Pour une suite croissante, chaque terme est inférieur ou égal au suivant.
- Pour une suite décroissante, chaque terme est supérieur ou égal au suivant.
- Pour une suite constante, chaque terme est égal au suivant.

La définition présente une certaine analogie avec celle de fonction croissante ou décroissante.



2°) Vocabulaire



3°) Remarques

- Une suite constante est monotone : elle est à la fois croissante et décroissante (mais elle ne l'est pas strictement).
- Une suite peut être monotone à partir d'un certain indice. Dans ce cas, on se moque de ce qui se passe pour les premiers termes ; ce qui est intéressant c'est ce qui se passe à partir d'un certain indice jusqu'à $+\infty$.
- Une suite peut être constante à partir d'un certain indice n_0 . On dit qu'elle est **stationnaire à partir de l'indice n_0** .

4°) Méthodes d'étude du sens de variation d'une suite

Voir chapitre suivant.

IX. Le « raisonnement de proche en proche »

1°) Exemple

Une file de voitures constituée de voitures rouges et de voitures vertes. On sait que si une voiture est rouge, alors la suivante est rouge.

Que peut-on dire dans les quatre circonstances suivantes :

- a) la première voiture est rouge ?
- b) la première voiture est verte ?
- c) la dernière voiture est rouge ?
- d) la dernière voiture est verte ?

2°) Mise en œuvre

Voir exercices

3°) Remarques

De nombreux raisonnements de proche en proche interviennent de manière cachée dans le cours de cette année.

Le raisonnement de proche en proche sera rendu rigoureux l'année prochaine avec le raisonnement par récurrence.

X. Calcul d'un terme de rang donné d'une suite définie par récurrence

On donne l'indice du terme dont on veut obtenir la valeur.

On utilise la structure de boucle : comme on connaît le nombre d'itérations au départ, il est préférable d'utiliser une boucle « Pour » plutôt qu'une boucle « Tant que ».

Voir exercices.

XI. Conjectures

- 1°) Ce chapitre se prête particulièrement à la formulation de conjectures pour les propriétés d'une suite :
- signe des termes
 - expression du terme général
 - monotonie

etc.

Nous verrons que la démonstration de ces conjectures n'est pas toujours facile. Elle nécessite souvent le recours à des raisonnements qui ne sont pas au programme de 1^{ère}, notamment lorsqu'il s'agit de suites récurrentes. On se contentera souvent dans ce dernier cas de raisonnements « de proche en proche » qui seront formalisés en Terminale sous la forme de « raisonnements par récurrence ».

2°) Utilisation d'outils logiciels

3°) Mise en garde

Attention, on ne peut pas conjecturer le sens de variation d'une suite à partir des premiers termes.

De manière générale, il faut être très prudent quand on formule des conjectures sur les suites. On veillera à employer les expressions usuelles dans le cadre d'une conjecture : « Il semble que ... », « On peut conjecturer que ... », « On peut penser que ... ».

La plus grande prudence s'impose car il s'agit de conjecturer des propositions quantifiées universellement.