Généralités sur les suites numériques

1.Définition d'une suite numérique. **p2**

2. Propriétés éventuelles d'une suite **p5**

1. Définition d'une suite numérique

1.1. Définition

On nomme suite numérique toute application d'une partie I de $\mathbb{N}$ vers $\mathbb{R}$

1.2. Notation

$$u : \begin{array}{ccc} I & \longrightarrow & \mathbb{R} \\ n & \longrightarrow & u(n) = u_n \end{array}$$

On lit « u indice n ».

On note la suite (u_n).

1.3. exemple

On peut définir une suite numérique en se donnant le terme général, pour tout entier naturel de I :

$u_n = g(n)$, par exemple I= $\mathbb{N}$ et $u_n = n^2 + 3n - 4$

(Il faut que I soit contenu dans l'ensemble de définition de g).

1.4. Remarque

On peut définir une suite numérique en se donnant le premier terme et un procédé (ou un algorithme) permettant de calculer u_{n+1} connaissant u_n pour tout entier naturel n.

C'est à dire connaissant u_0 et $f(u_n) = u_{n+1}$.

(Attention à l'ensemble de définition de f).

Exemple

- $u_0 = 1$ **et** pour tout entier naturel n $u_{n+1} = 3u_n + 4$.
- Si on utilise le tableur de Open office, on peut déterminer facilement les premiers termes de la suite (u_n).

  On écrit les termes de la suite (u_n) dans la première colonne.

  A1 : 1 B1 : =3*A1+4

  A2 : =B1

  On étire jusque A30 et B30.

  En A30 on aura u_{29} et en B30 on aura u_{30}.

	A	B	C
1	1	7	
2	7	25	
3	25	79	
4	79	241	
5	241	727	
6	727	2185	
7	2185	6559	
8	6559	19681	
9	19681	59047	
10	59047	177145	
11	177145	531439	
12	531439	1594321	
13	1594321	4782967	
14	4782967	14348905	
15	14348905	43046719	
16	43046719	129140161	
17	129140161	387420487	
18	387420487	1162261465	
19	1162261465	3486784399	
20	3486784399	10460353201	
21	10460353201	31381059607	
22	31381059607	94143178825	
23	94143178825	282429536479	
24	282429536479	847288609441	
25	847288609441	2541865828327	
26	2541865828327	7625597484985	
27	7625597484985	22876792454959	
28	22876792454959	68630377364881	
29	68630377364881	205891132094647	
30	205891132094647	617673396283945	
31			

. On peut écrire un algorithme pour afficher u_{30} (sans afficher les 30 premiers termes).

Variables : n est un entier naturel
u est un nombre réel

Initialisation : u=1
n=0

Traitement : Tant que $n>30$
$n=n+1$
$u=u\times3+4$
Fin Tant que

Sortie : Afficher u

. <u>Programme en Python</u>

File Edit Format Run Options Window Help

```
print('début de programme')
u,n=1,0
while(n<30):
    n=n+1
    u=u*3+4
print("u_30="+str(u))
print('Fin de progamme')
```

. <u>On exécute le programme</u>


```
Python 3.6.4 Shell
File  Edit  Shell  Debug  Options  Window  Help
Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:54:40) [MSC v.1900 64 bit (AMD64)]
 on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Serge\Documents\Programmes-Python\Gen.suites1.py =====
début de programme
u_30=617673396283945
Fin de progamme
>>>
```

1.5. Principe de récurrence (admis)

Pour démontrer que la propriété de l'entier naturel n : P(n) est vraie pour tout $n \geqslant n_0$

(n_0 **est un entier naturel).**

<u>Il suffit de :</u>

. **vérifier que** $P(n_0)$ **est vraie (initialisation).**

. **Démontrer que la propriété est héréditaire pour tout entier naturel n supérieur ou égal** à n_0 **.**

C'est à dire

On suppose que la propriété P(n) (pour n supérieur ou égal à n_0 **) est vraie (hypothèse de récurrence) et on doit démontrer que P(n+1) est vraie.**

1.6. Remarque

On peut définir une suite numérique en se donnant les deux premiers termes et un procédé (ou un algorithme) permettant de calculer u_{n+2} connaissant u_{n+1} et u_n .

<u>Exemple :</u>

$u_0=0$ et $u_1=1$ et pour tout entier naturel n, on a $u_{n+2}=5u_{n+1}-6u_n$.

. Si on utilise le tableur d'open office.

A1 : 0 B1 : 1 C1 : =5*B1-6*A1

A2 : =B1 B2 : =C1

Puis on étire jusque : A30 ; B30 et C30

En A30 on obtient u_{29}

En B30 on obtient u_{30}

En C30 on obtient u_{31}

	A	B	C	D
1	0	1	5	
2	1	5	19	
3	5	19	65	
4	19	65	211	
5	65	211	665	
6	211	665	2059	
7	665	2059	6305	
8	2059	6305	19171	
9	6305	19171	58025	
10	19171	58025	175099	
11	58025	175099	527345	
12	175099	527345	1586131	
13	527345	1586131	4766585	
14	1586131	4766585	14316139	
15	4766585	14316139	42981185	
16	14316139	42981185	129009091	
17	42981185	129009091	387158345	
18	129009091	387158345	1161737179	
19	387158345	1161737179	3485735825	
20	1161737179	3485735825	10458256051	
21	3485735825	10458256051	31376865305	
22	10458256051	31376865305	94134790219	
23	31376865305	94134790219	282412759265	
24	94134790219	282412759265	847255055011	
25	282412759265	847255055011	2541798719465	
26	847255055011	2541798719465	7625463267259	
27	2541798719465	7625463267259	22876524019505	
28	7625463267259	22876524019505	68629840493971	
29	22876524019505	68629840493971	205890058352825	
30	68629840493971	205890058352825	617671248800299	
31				

. On peut écrire un algorithme pour afficher u_{30} (sans afficher les 30 premiers termes).

Variables : n, u, v et w sont des entiers naturels
Initialisation : u=0
v=1
n=0
Traitement : Tant que n< 30
w=5*v-6*u
u=v
v=w
n=n+1
Fin Tant que
Sortie : Afficher u

. <u>Programme en Python</u>

```
File   Edit   Format   Run   Options   Window   Help
print('Début de programme')
u,v,n=0,1,0
while(n<30):
    w=5*v-6*u
    u=v
    v=w
    n=n+1
print("u_30="+str(u))
print('Fin de programme')
```

. On exécute le progamme

```
>>>
===== RESTART: C:\Users\Serge\Documents\Programmes-Python\Gen-suites2.py
Début de programme
u_30=205890058352825
Fin de programme
>>>
```

2. Propriétés éventuelles d'une suite

2.1. Suites majorées, minorées, bornées

2.1.a. On dit que la suite (u_n) est majorée si et seulement s'il existe un nombre réel M tel que pour tout entier naturel n, on ait : $u_n \leqslant M$.
On dit que M est un majorant de la suite (u_n) .

2.1.b. On dit que la suite (u_n) est minorée si et seulement s'il existe un nombre réel m tel que pour tout entier naturel n, on ait : $m \leqslant u_n$.
On dit que m est un minorant de la suite (u_n) .

2.1.c. Lorsque la suite (u_n) est minorée et majorée, on dit qu'elle est bornée.
On a alors pour tout entier naturel n : $m \leqslant u_n \leqslant M$.

2.1.d. Exemple

On considère la suite (u_n) définie pour tout entier naturel n par $u_n=\frac{2n+3}{n+1}$.

$u_n=g(n)$ avec $g(x)=\frac{2x+3}{x+1}$ (g est définie sur l'intervalle $[0;+\infty[$).

. On veut démontrer que 2 est un minorant de la suite.
Pour cela on détermine le signe de u_n-2 .

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

Pour tout entier naturel n, $\frac{1}{n+1}>0$, donc pour tout entier naturel n, $u_n-2>0 \quad \Leftrightarrow \quad u_n>2$.

2 est un minorant de (u_n) (1 ou -2 sont d'autres minorants de la suite).

. On veut démontrer que 3 est un majorant de la suite.

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

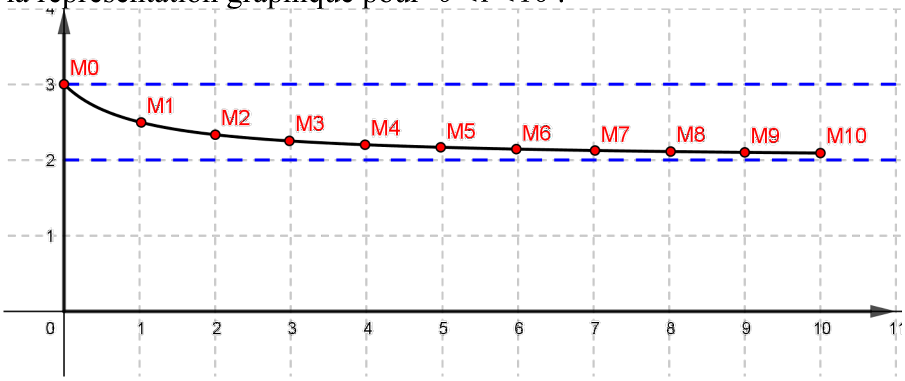
Pour tout entier naturel n, $\frac{n}{n+1}>0$, donc pour tout entier naturel n, $3-u_n>0 \quad \Leftrightarrow \quad 3>u_n$.

3 est un majorant de la suite (u_n) (4 ou 5 sont d'autres majorants de la suite).

. La suite (u_n) est donc bornée.

On pouvait remarquer que pour tout entier naturel n, on avait $u_n=2+\frac{1}{n+1}$ et que : $0<\frac{1}{n+1}\leqslant 1$.

- Si on construit la représentation graphique de la suite (u_n), c'est à dire l'ensemble des points $M_i(i;u_i)$ pour tous les entiers naturels i.
  On a $u_n=g(n)$ donc les points M_i appartiennent à la représentation graphique de g.
  On construit la représentation graphique pour $0\leqslant i\leqslant 10$.



Tous les points M_i appartiennent à la partie du plan comprise entre les droites d'équations y=2 et y=3.

2.2. Limite d'une suite numérique

2.2.a. $u_n=g(n)$

- Si $\lim_{x\to+\infty} g(x)=L$ alors $\lim_{n\to+\infty} g(n)=L$

Exemple

$u_n=\dfrac{1}{2n+1}$ $\quad \lim_{x\to+\infty}\dfrac{1}{2x+1}=0$ donc $\lim_{n\to+\infty} u_n=0$

- si $\lim_{x\to+\infty} g(x)=+\infty$ alors $\lim_{n\to+\infty} u_n=+\infty$

Exemple

$u_n=n^2+n+1$ $\quad \lim_{x\to+\infty}(x^2+x+1)=+\infty$ donc $\lim_{n\to+\infty} u_n=+\infty$.

2.2.b. Définition

On nomme suite convergente, toute suite qui admet une limite finie.

La suite (u_n) converge vers L si et seulement si $\lim_{n\to+\infty} u_n=L$.

Exemple

$u_n=\dfrac{2n+3}{n+1}$ $\quad g(x)=\dfrac{2x+3}{x+1}$ $\quad \lim_{x\to+\infty} g(x)=2$ $\quad \lim_{n\to+\infty} u_n=2$

donc la suite (u_n) converge vers2.

2.2.c. **Une suite non convergente est dite divergente.**

Une suite divergente peut admettre pour limite $+\infty$ ou $-\infty$ ou ne pas admettre de limite (exemple : $(-1)^n$).

2.3. Variations d'une suite

2.3.a. Définitions

- On dit que la suite (u_n) est **croissante** si et seulement si pour tout entier naturel n, on ait : $u_{n+1}\geqslant u_n$.
- On dit que la suite (u_n) est **strictement décroissante** si et seulement si pour tout entier naturel n, on

ait : $u_{n+1} < u_n$

- On dit que la suite (u_n) est **monotone** si elle est soit croissante ou soit décroissante.
- La suite (u_n) est **constante** si et seulement si pour tout entier naturel n, on ait : $u_{+1} = u_n$.

2.3.b. Remarques :

- Si $u_n = g(n)$ alors en général pour déterminer les variations de la suite (u_n) il suffit de déterminer les variations de la fonction g.
- Il existe des suites non monotones.

  Exemple : $((-1)^n)$
- Pour étudier les variations de la suite (u_n) , on peut déterminer le signe de la différence : $u_{n+1} - u_n$ pour tout entier naturel n.

2.3.c. Exemple 1 :

(u_n) est la suite définie par $u_0 = 10$ et pour tout entier naturel n, $u_{n+1} = f(u_n) = \frac{1}{5}u_n + 10$.

$f(x) = \frac{1}{5}x + 10$ (f est strictement croissante sur $\mathbb{R}$).

Attention on ne peut pas affirmer que la suite est croissante.

Pour démontrer que la suite est monotone, on va démontrer que pour tout entier naturel n, le signe de $(u_{n+2} - u_{n+1})$ est égal au signe de $(u_{n+1} - u_n)$.

$$u_{n+2} - u_{n+1} = f(u_{n+1}) - f(u_n) = \frac{1}{5}u_{n+1} + 10 - \left(\frac{1}{5}u_n + 10\right) = \frac{1}{5}(u_{n+1} - u_n)$$

$\frac{1}{5} > 0$ donc $(u_{n+2} - u_{n+1})$ et $(u_{n+1} - u_n)$ sont de même signe.

On calcule $u_1 = \frac{1}{5}u_0 + 10 = \frac{1}{5} \times 10 + 10 = 12$

Conséquence : $u_1 - u_0 = 2 > 0$

Raisonnement par récurrence

On veut démontrer en utilisant le principe de récurrence que la propriété P(n) : $u_{n+1} - u_n > 0$ est vraie pour tout entier naturel n.

- Initialisation

  On a $u_1 - u_0 > 0$ donc $u_1 > u_0$ et P(0) est vraie.
- Hérédité

  Pour démontrer que la propriété est héréditaire, pour tout entier naturel n, on suppose que P(n) est vraie (c'est à dire $u_{n+1} > u_n$) et on doit démontrer que P(n+1) est vraie (c'est à dire $u_{n+2} > u_{n+1}$).

  Si $u_{n+1} > u_n$ alors $u_{n+1} - u_n > 0$, comme $u_{n+2} - u_{n+1}$ a le même signe que $u_{n+1} - u_n$ on obtient $u_{n+2} - u_{n+1} > 0$ et $u_{n+1} > u_n$.
- Conclusion

  Le principe de récurrence nous permet d'affirmer que pour tout entier naturel n, on a $u_{n+1} > u_n$.

Donc **la suite** (u_n) **est strictement croissante.**

2.3.d. Exemple 2 :

(v_n) est la suite définie par $v_0=15$ et pour tout entier naturel n, $v_{n+1}=f(v_n)=\frac{1}{5}v_n+10$.

Attention la suite (v_n) est distincte de la suite (u_n) .

$v_1=\frac{1}{5}\times v_0+10=\frac{1}{5}\times 15+10=13$

$v_1-v_0<0$ soit $v_1<v_0$

En utilisant un raisonnement analogue au précédent on démontre que pour tout entier naturel n, $v_{n+1}<v_n$

Donc **la suite** (v_n) **est strictement décroissante.**

2.3.e. Exemple 3 :

(w_n) est la suite définie par $w_0=12{,}5$ et pour tout entier naturel n, $w_{n+1}=f(w_n)=\frac{1}{5}w_n+10$.

$w_1=\frac{1}{5}\times w_0+10=\frac{1}{5}\times 12{,}5+10=12{,}5$ donc $w_1=w_0$.

On peut aussi démontrer en utilisant le principe de récurrence que pour tout entier naturel n, $w_{n+1}=w_n$.

Donc **la suite** (w_n) **est constante.**