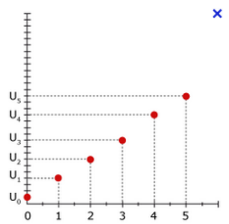
RESUME – SUITES NUMERIQUES

Rappels de 1ère

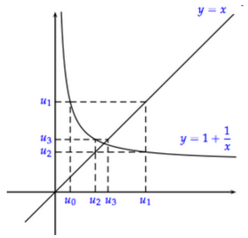
Définition : C'est une fonction de N (ou partie de N) dans R

On peut définir une suite de trois façons différentes :

- Par **une formule explicite** *Ex :* $u_n = 3n + 2$ Calcul du terme de rang n : $u_3 = 3\times 3 + 2 = 9 + 2 = 11$
- Par **une relation de récurrence** *Ex :* $u_{n+1} = 3u_n + 2$ *et* $u_0 = 1$
Calcul du terme de rang n : $u_{0+1} = u_1 = 3u_0 + 2 = 3\times 1 + 2 = 3 + 2 = 5$
- A l'aide d'une ou plusieurs autres suites *Ex :* $u_n = 2w_n + 4$



Représentation graphique suite définie par une formule explicite



Représentation graphique suite définie par une relation de récurrence

Notation : Le terme général se note u_n ou u(n) avec n l'indice (ou le rang) du terme.
La suite u se note (u(n)) ou u ou (u_n)

Sens de variation : étudier le signe de $u_{n+1} - u_n$ (ou comparer u_n et u_{n+1}):

Si $u_{n+1} - u_n > 0$ (ou $u_{n+1} > u_n$) à partir d'un certain rang p, alors (u_n) est croissante à partir de ce rang.
Si $u_{n+1} - u_n < 0$ (ou $u_{n+1} < u_n$) à partir d'un certain rang p, alors (u_n) est décroissante à partir de ce rang.
Si $u_{n+1} - u_n = 0$ (ou $u_{n+1} = u_n$) à partir d'un certain rang p, alors (u_n) est constante à partir de ce rang.

Sens de variation : Comparer $\frac{u_{n+1}}{u_n}$ à 1 (avec $u_n > 0$ pour tout entier naturel) :

Si $\frac{u_{n+1}}{u_n} > 1$ à partir d'un certain rang p, alors (u_n) est croissante à partir de ce rang.
Si $\frac{u_{n+1}}{u_n} < 1$ à partir d'un certain rang p, alors (u_n) est décroissante à partir de ce rang.
Si $\frac{u_{n+1}}{u_n} = 1$ à partir d'un certain rang p, alors (u_n) est constante à partir de ce rang.

Sens de variation : suites du type $u_n = f(n)$ avec f une fonction dérivable sur R^+:

Si f est croissante à partir de $x = x_0$, alors (u_n) est croissante à partir de l'entier suivant x_0
Si f est décroissante à partir de $x = x_0$, alors (u_n) est décroissante à partir de l'entier suivant x_0

Suites arithmétiques :

Définition : $u_{n+1} = u_n + r$.
Pour prouver que u_n est arithmétique prouver que $u_{n+1} - u_n = r$

Formule explicite : $u_n = r(n - p) + u_p$ si je choisis p = 0 alors $u_n = rn + u_0$.

Sens de variations Si r = 0 alors la suite est constante
Si r > 0 alors la suite est croissante
Si r < 0 alors la suite est décroissante

Somme des n premiers entiers: $S = 1 + 2 + 3 + \cdots + n = \frac{(n+1)\times n}{2}$

Somme de termes consécutifs: $S = u_p + u_{p+1} + u_{p+2} + \cdots + u_n = \frac{(u_p + u_n)\times nbre\,de\,termes}{2}$

Nombre de termes = n-p+1

Suites géométriques :

Définition : $u_{n+1} = u_n \times q$. Pour prouver que u_n est géométrique prouver que $\frac{u_{n+1}}{u_n} = q$

Formule explicite : $u_n = q^{n-p} \times u_p$ si je choisis p = 0 alors $u_n = q^n \times u_0$.

	$u_0 > 0$	$u_0 < 0$
$0 < q < 1$	Décroissante	Croissante
$q > 1$	Croissante	Décroissante
q < 0	Non monotone	Non monotone
q=1 ou 0	constante	constante

Somme de termes consécutifs : $S = 1 + q + q^2 + \cdots + q^n = \frac{1-q^{(n+1)}}{1-q}$

Somme de termes consécutifs : $S = u_p + u_{p+1} + u_{p+2} + \cdots + u_n = u_p \times \frac{1-q^{nbre\,de\,termes}}{1-q}$

Pourcentages :

Augmenter une quantité de x % revient à multiplier cette quantité par $(1 + \frac{x}{100})$

Diminuer une quantité de x % revient à multiplier cette quantité par $(1 - \frac{x}{100})$

Suites majorées, minorées, bornées

Soit M et m deux nombres réels, On dit que la suite (u_n) est :
- **majorée** par M si, pour tout n, $u_n \leq M$. On dit aussi que M est un **majorant** de (u_n)
- **minorée** par m si, pour tout n, $m \leq u_n$. On dit aussi que m est un **minorant** de (u_n)
- **bornée** si, pour tout n, $m \leq u_n \leq M$.

Définition de la limite finie d'une suite (convergence)

Une suite u_n tend vers le réel l, quand n tend vers +∞, si tout intervalle ouvert contenant l, contient toutes les valeurs de u_n à partir d'un certain rang. On écrit alors $\lim_{n\to+\infty} u_n = l$.

Définition de la limite infinie d'une suite (divergence)

Une suite u_n tend vers +∞ (resp. -∞) si tout intervalle de la forme]A ; +∞[(resp.]-∞ ; A[) contient toutes les valeurs de u_n à partir d'un certain rang. On écrit alors $\lim_{n\to+\infty} u_n = +\infty$ (resp. $\lim_{n\to+\infty} u_n = +\infty$).

Opérations sur les limites :

Dans ces tableaux, L et L' représentent des réels. ∞ représente +∞ ou -∞.

Produit de limites :

Lim u_n	**L**	**L ≠ 0**	**0**	∞
Lim v_n	**L'**	∞	∞	∞
Lim u_n × v_n	**L × L'**	∞	**IND**	∞

Somme de limites :

Lim u_n	**L**	+∞	**L**	+∞	-∞
Lim v_n	**L'**	-∞	+/-∞	+∞	-∞
Lim u_n + v_n	**L + L'**	**IND**	+/-∞	+∞	-∞

Quotient de limites :

Lim u_n	**L**	**L≠ 0**	**0**	∞	**L**	∞
Lim v_n	**L'≠ 0**	**0**	**0**	**L'**	∞	∞
Lim u_n/v_n	**L / L'**	∞	**IND**	∞	**0**	**IND**

Limites et comparaison

Soient les suites u_n et v_n :

- Si pour tout $n \geq n_0$ on a $v_n \geq u_n$ et $\lim\limits_{n\to+\infty} u_n = +\infty$ alors on a $\lim\limits_{n\to+\infty} v_n = +\infty$

- Si pour tout $n \leq n_0$ on a $v_n \leq u_n$ et $\lim\limits_{n\to+\infty} u_n = -\infty$ alors on a $\lim\limits_{n\to+\infty} v_n = -\infty$

Théorème dit des « gendarmes » :

Soient u_n, v_n et w_n, trois suites telles qu'à partir d'un certain rang n_0, on a :

$u_n \leq v_n \leq w_n$ et si $\lim\limits_{n\to+\infty} u_n = l$ et $\lim\limits_{n\to+\infty} w_n = l$ alors on a $\lim\limits_{n\to+\infty} v_n = l$

Limite des suites majorées ou minorées

- Une suite croissante et majorée est convergente
- Une suite décroissante et minorée est convergente.

$\lim\limits_{n\to+\infty} q^n = ?$

- Si |q |< 1 alors la limite vaut 0
- Si q = 1 alors la limite vaut 1
- Si q > 1 alors la limite vaut +∞
- Si q ≤ -1 alors pas de limite

Raisonnement par récurrence :

Principe : Pour démontrer qu'une propriété dépendant d'un entier naturel n (P_n) est vraie pour tout entier n à partir d'un certain rang n_0 :

1) On vérifie que P_{n0} est vraie

2) On vérifie l'hérédité : on suppose qu'au rang p on a P_p vraie et on cherche à prouver que la propriété reste vraie au rang suivant soit P_{p+1} vraie.

Algorithmes :

Algorithmes de calcul de termes :

Langage naturel (Calculer le terme d'une suite)	**Python**
Avec Boucle « Pour » Saisir n u ← u_0 Pour i allant de 1 à n u← f(u) [la formule de u_{n+1}] Fin Pour Renvoyer (u) *Pour une formule explicite pas besoin de boucle* Saisir n u ← $f(n)$ renvoyer (u)	def suite(n) : u = u_0 for i in range (n): u= f(u) [la formule de u_{n+1}] return(u) *Pour une formule explicite pas besoin de boucle* Saisir n def suite(n) : u = f(n) return (u)
Langage naturel (Calculer les termes d'une suite)	**Python**
L[0] ← u_0 Pour i allant de 1 à n Ajouter à la liste L, f(L(i-1)) [la formule de u_{n+1} ou formule explicite] Fin Pour Renvoyer (L)	def suite(n) : L= [ ] L.append(u_0) for i in range(1,n+1) : L.append(f(L[i-1]) [la formule de u_{n+1} ou formule explicite] return(L)

Algorithmes de calcul de sommes de termes :

Langage naturel	**Python**
Avec Boucle « Pour » Saisir n u ← u_0 S← u_0 Pour i allant de 1 à n u← f(u) [la formule de u_{n+1}] S← S + u Fin Pour Renvoyer (S)	defsomme(n) : u = u_0 S = u for i in range (n): u= f(u) [la formule de u_{n+1}] S = S + u return(u)

Algorithmes de seuil :

Langage naturel	**Python**
u ← u_0 n← 0 Tant que [condition sur u] u← f(u) [la formule de u_{n+1}] n← n +1 FinTant que Renvoyer (n)	defseuil( ) : u = u_0 n = 0 while[condition sur u]: u= f(u) [la formule de u_{n+1}] n = n + 1 return(n)

Démonstrations :

- Suite croissante non majorée tend vers +∞
- Limites de q^n
- Théorème de comparaison

Approfondissements :

Suites adjacentes :

Définition : (u_n) et (v_n) sont adjacentes ssi l'une est croissante et l'autre décroissante et si $\lim_{n\to+\infty} u_n - v_n = 0$

Convergence : deux suites adjacentes convergent vers la même limite.

Suites récurrentes linéaires d'ordre 2 :

Définition : On dit que $(u_n)_{n\in\mathbb{N}}$ est une suite récurrente linéaire d'ordre 2 ssi $\forall n \in \mathbb{N}$ on a :

$$\begin{cases} u_{n+2} = a\, u_{n+1} + b\, u_n \quad \text{avec } (a\,;b) \in \mathbb{R}^2 \\ (u_0\,; u_1) \in \mathbb{R}^2 \end{cases}$$

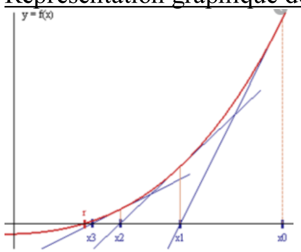
Méthode de Newton (approximation de nombres) :

La méthode de Newton est une méthode qui permet de donner une solution approchée d'une équation de la forme $f(x) = 0$ que l'on ne peut pas résoudre sinon. On peut également résoudre une équation de la forme $f(x) = g(x)$ en posant $h(x) = f(x) - g(x)$ et en résolvant $h(x) = 0$.

Principe de la méthode : Si on cherche la solution de $f(x) = 0$ sur I, f doit être dérivable sur I.

- On choisit un point x_0 que l'on sait assez proche de la solution (estimation grossière)
- On utilise une approximation affine pour dire que $f(x) \approx f(x0) + f'(x0)(x - x0)$
- On résout $f(x) = 0$ avec l'approximation, soit $f(x0) + f'(x0)(x - x0) = 0$
- La solution de cette équation que l'on note x_1 s'approche de la solution recherchée
  $$x1 = x0 - f(x0)/f'(x0)$$
- On réitère le processus avec x1,… Plus on fait d'itération, plus l'estimation de la solution est meilleure.
- De manière générale, pour passer de la kème itération à la k+1 ème, on a :
  $$x_{k+1} = x_k - f(x_k)/f'(x_k)$$
- On s'arrête lorsque l'on a obtenu une estimation suffisamment précise.

Représentation graphique de l'algorithme :



Limites de la méthode :

- Si on a aucune idée de la solution, le critère d'arrêt des itérations peut être compliqué à définir et peut amener à une solution très éloignés.
- En cas de non dérivation ponctuelle ou de problème d'ensemble de définition, on peut se retrouver avec une boucle infinie.

Méthodes (exercices) :

	Hachette	**Hatier**	**Mes exos**	**Sesamaths**
Révisions	1-5	-	1	
A) Démontrer qu'une suite est majorée/bornée/minorée		-	2	
B) Théorème de convergence monotone	**38-39**	58,150-151	3	14
C) Calculs de limites simples	**13-14**	96-101	4	15
D) Limite d'un polynome	**16**	52,54,121	4	16
E) Quotient de polynômes	**17**	53,127	4	17
F) limite q^n	**27-30**	152	4	18
G) indetermination q^n		152	4	18
H) limites des cos/sin		145-149	5	19
I) indet racine carrée	**22**	133	4	20
J) theorème de comparaison	**20-25,**	55-57 ,143-149	5	21
K) Récurrence	6-19	43-47,73-90	6	22

Exercices de synthèse :

	Hachette	**Hatier**	**Mes exos**	**Sesamaths**
Algorithmes	**65**	157-158	8,9	-
Synthèse	**67,77-80**	173-174,159-168	7	23
QCM	**64**		10	-
Vrai/faux			11	-
Prise d'initiative				-
Approfondissement	**75**		12	-