

Définition et notations

Ex 1 : Notations, termes, indices

On considère la suite u définie sur \mathbb{N} , par : 0 ; 1 ; 8 ; 27 ; 64 ; 125 ...

- Déterminer u_0 , u_1 et u_2 .
- Avec quel indice note-t-on le cinquième terme de la suite u ?
- Déterminer le terme de rang 6.
- Conjecturer le terme général de la suite u en fonction de n .
- La suite u peut aussi se noter u_n : vrai ou faux ?
- La suite u est-elle une fonction ? Si, oui, donner son ensemble de définition.

Ex 2 : QCM : restituer les notions du cours

On considère une suite u définie sur \mathbb{N} de premier premier terme d'indice 0.

Dans chaque cas, indiquer la ou les bonnes réponses.

- Quel est le 5^{ème} terme de la suite ?
 a) u_4 b) u_5 c) u_7 d) 4
- Quel est le rang du 10^{ème} terme ?
 a) u_{10} b) u_9 c) 9 d) 10 e) 11
- Quel est le terme de rang 7 ?
 a) u_6 b) u_7 c) 6 d) 8
- Les égalités suivantes sont-elles valides ?
 a) $u_3 = 2,7$ b) $u_{-5} = 4$ c) $u_{2,4} = 8$ d) $u_5 = -50$

Ex 3 : Vrai ou faux : restituer les notions du cours

Indiquer si la situation peut-être traduite par une suite définie sur une partie de \mathbb{N} .

- À chaque saut du perchiste, on associe la performance.
- À chaque intensité du courant, on associe la tension relevée aux bornes du générateur.
- À chaque instant de la journée, on associe la température extérieure.
- À chaque heure de la journée, on associe le nombre d'automobiles passées sur l'avenue des Champs-Élysées.
- À chaque mois, on associe la durée des communications téléphoniques.

Ex 4 : Combien de termes

Combien dénombre-t-on de termes :

- a) de u_1 à u_{37} b) de u_0 à u_{18} c) de u_{17} à u_{85}

Suite définie par une formule explicite

Ex 5 : Calculs des premiers termes

Dans chaque cas, déterminer les valeurs des quatre premiers termes de la suite (u_n) :

- a) $u_n = n^2 - 4$ b) $u_n = \frac{1}{n+1}$ c) $u_n = (-1)^n$ d) $u_n = n^2 - 5n + 4$

Ex 6 : Trouver une formule explicite

Déterminer une formule explicite de la suite telle que :

- Chaque terme est égal à la moitié de son rang.
- Chaque terme est égal au cube de son rang.
- $u_n = f(n)$ où $f(x) = \sqrt{2x+7}$
- $u_0 = \frac{1}{2}$, $u_1 = \frac{1}{4}$, $u_2 = \frac{1}{8}$, $u_3 = \frac{1}{16}$...

5) $u_0 = 1$, $u_1 = -1$, $u_2 = 1$, $u_3 = -1$...

6) $u_0 = 5$, $u_1 = -\frac{1}{5}$, $u_2 = 5$, $u_3 = -\frac{1}{5}$...

Ex 7 : Un peu de logique

Soit (u_n) la suite définie, pour tout entier naturel, par $u_n = n^3 - 6n^2 + 5n + 4$

- Déterminer u_0 et u_1 .
- La proposition P : « $\forall n \in \mathbb{N}, u_n = 4$ » est-elle vraie ?
- Écrire la négation de la proposition P.
- Combien de termes u_n sont-ils égaux à 4 ? Justifier.

Ex 8 : Ensemble de définition

Indiquer, en justifiant, si chaque proposition est vraie ou fausse.

La suite de terme général u_n est définie pour tout entier naturel n .

1) $u_n = \frac{5}{3^n - 1}$ 2) $u_n = \frac{3}{3n - 2}$

3) $u_n = \sqrt{-n^2 + 9n - 8}$ 4) $u_n = \sqrt{3 - 3(-1)^n}$

Suite définie par une formule de récurrence

Ex 9 : Déterminer u_n en fonction de u_{n-1}

Soit (u_n) la suite définie par $u_0 = 2$ et pour tout entier naturel n ,
 $u_{n+1} = u_n^2 - 3u_n + 2$.

- Calculer u_1 , u_2 et u_3 .
- Exprimer u_n en fonction de u_{n-1} .
- Faire de même avec la suite (v_n) définie par $v_0 = 1$ et $v_{n+1} = \frac{2}{v_n + 5}$.

Ex 10 : Déterminer u_{n+1} en fonction de u_n

Dans chaque cas, déterminer une formule de récurrence de la suite.

- Chaque terme est égal au triple du terme précédent.
- La somme de deux termes consécutifs est toujours égale à 5.
- Chaque terme est une augmentation de 20 % du terme précédent.
- $u_{n+1} = f(u_n)$ où $f(x) = \frac{3x+5}{4x+1}$
- $u_n = 7^{n-3}$ 6) $u_n = 2n - 5$
- $u_n = 1 \times 2 \times 3 \times \dots \times n$ 8) $u_n = \frac{1}{n+1}$
- $u_0 = 8$, $u_1 = 10$, $u_2 = 13$, $u_3 = 17$, $u_4 = 22$...
- $u_0 = 1$, $u_1 = 5$, $u_2 = 21$, $u_3 = 85$...

Ex 11 : Calculs des premiers termes

Dans chaque cas, calculer les trois premiers termes de la suite.

- $u_0 = 3$ et, pour tout entier naturel n , $u_{n+1} = 3u_n - 2n + 2$
- $u_1 = 4$ et, pour tout entier naturel n , $u_n = 2u_{n-1} + n$
- $u_0 = 1$, $u_1 = -1$ et, pour tout entier naturel n , $u_{n+2} = 2u_{n+1} - 3u_n + 2$

Ex 12 : Ensemble de définition

Soit (u_n) la suite définie par $u_0 = -5$, et pour tout entier naturel n , par $u_{n+1} = \sqrt{u_n + 6}$.

- 1) Déterminer la fonction f telle que, pour tout entier naturel, $u_{n+1} = f(u_n)$. Quel est l'ensemble de définition de f ?
- 2) Soit $I = [-5; 3]$. Montrer que si $x \in I$, alors $f(x) \in I$.
- 3) Vérifier que $u_0 \in I$ et démontrer que, si $u_n \in I$, alors $u_{n+1} \in I$. En déduire que la suite (u_n) est bien définie sur \mathbb{N} .
- 4) Indiquer si les suites ci-dessous sont bien définies sur \mathbb{N} .
 - a) $v_0 = 10$, et pour tout entier naturel n , $v_{n+1} = \frac{1}{v_n + 5}$
 - b) $w_0 = 7,5$, et pour tout entier naturel n , $w_{n+1} = \sqrt{2w_n - 6}$

Algorithme, calculatrice, tableur

Ex 13 : Algorithme avec des listes

algorithme a	Traduction en python
Lire n Pour i allant de 0 à n $u[i] \leftarrow 2 * i - 3 * i + 1$ Afficher $u[i]$ FinPour	<pre>n=int(input("n=")) u=[] for i in range(0,n-1): u.append(2*i**2-3*i+1) print (u[i])</pre> <p>(Faire tourner le programme)</p>
algorithme b Lire n Lire $u[0]$ Pour i allant de 1 à n $u[i] \leftarrow 2 * u[i-1] * u[i-1] - 3 * u[i-1] + 1$ Afficher $u[i]$ FinPour	<pre>n=int(input("n=")) u=[] u.append(0) for i in range(0,n-1): u.append(2*u[i-1]**2-3*u[i-1]+1) print (u[i])</pre> <p>(Faire tourner le programme)</p>

Déterminer le rôle de ces deux algorithmes.

Ex 14 : Calculatrice

À l'aide de la calculatrice, déterminer les dix premiers termes de :

- 1) La suite u définie pour tout entier naturel n , par $u_n = \sqrt{n^2 + \frac{1}{2}n}$.
- 2) La suite v définie pour tout entier naturel n , par $\begin{cases} v_0 = 3 \\ v_{n+1} = \sqrt{v_n^2 + \frac{1}{2}v_n} \end{cases}$

Ex 15 : Tableur

Dans chacun des cas ci-dessous, une suite a été définie sur un tableur : indiquer laquelle.

1)		2)	
B3	$f(x) = \text{RACINE}(A3^2+3)$	B5	$f(x) = \text{RACINE}(B4^2+3)$
n	$u(n)$	n	$u(n)$
1	0,173205081	1	0
2	1,414213562	2	1
3	2,236067977	3	2
4	3,16227766	4	3,16227766
5	4,123105626	5	4,123105626

Comportement d'une suite

Ex 16 : Vrai ou faux : restituer les notions du cours

- Soit (u_n) une suite telle que pour tout entier naturel n , $0 < u_n < 5$.
- 1) (u_n) est minorée par 4.
 - 2) (u_n) est minorée par 0.
 - 3) (u_n) est minorée par 1.

- 4) 6 est un majorant de (u_n) .
- 5) (u_n) est bornée.
- 6) (u_n) est nécessairement croissante.
- 7) (u_n) peut admettre une limite infinie.
- 8) (u_n) admet nécessairement une limite finie.
- 9) (u_n) admet une infinité de majorants et de minorants.

Ex 17 : Vrai ou faux : restituer les notions du cours

- 1) Une suite peut être à la fois croissante et décroissante.
- 2) Si (u_n) est de signe constant, alors (u_n) est monotone.
- 3) Une suite croissante est toujours majorée.
- 4) Une suite peut être à la fois croissante et majorée.
- 5) Une suite décroissante est toujours majorée.
- 6) Soit une suite (u_n) et la fonction f telle que pour tout $n \in \mathbb{N}$, $u_n = f(n)$.
 - a) Si (u_n) est croissante, alors f est croissante sur \mathbb{R}^+ .
 - b) Si f est croissante sur \mathbb{R}^+ , alors (u_n) est croissante.
- 7) Une suite décroissante peut avoir une limite égale à 100.
- 8) On peut déterminer le signe de la dérivée d'une suite (u_n) pour déterminer les variations de (u_n) .

Ex 18 : Conjecturer avec un tableur ou une calculatrice

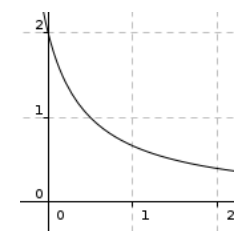
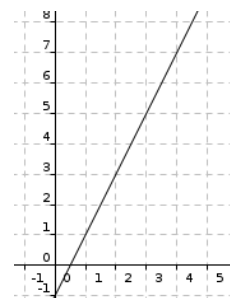
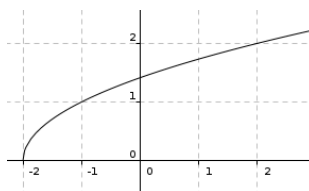
Dans chaque cas, utiliser un tableur ou une calculatrice pour conjecturer le comportement de la suite (variations et limites éventuelles)

- 1) $u_n = 2\sqrt{n+4}$
- Déterminer aussi le premier indice n tel que $u_n > 100$
- 2) $\begin{cases} u_0 = 5 \\ u_{n+1} = 2\sqrt{u_n + 4} \end{cases}$ 3) $v_n = \frac{10n}{n+1}$ 4) $\begin{cases} v_0 = 7 \\ v_{n+1} = \frac{10v_n}{v_n + 1} \end{cases}$

Ex 19 : Représenter graphiquement une suite définie par récurrence

Dans chaque cas, on considère la fonction f telle que, pour tout entier naturel n , $u_{n+1} = f(u_n)$. À l'aide de la droite $d: y = x$, représenter les premiers termes de la suite sur les axes, puis conjecturer le comportement de la suite (variations et limites éventuelles).

- 1) $u_0 = -1,5$ et $u_{n+1} = \sqrt{u_n + 2}$
- 2) $u_0 = 1,5$ et $u_{n+1} = 2u_n - 1$



- 3) $u_0 = 2$ et $u_{n+1} = \frac{1}{u_n + 0,5}$

Ex 20 : Conjecturer

Dans chaque cas, utiliser la méthode votre choix pour conjecturer le comportement de la suite (variations et limites éventuelles)

$$1) u_n = \frac{n^2 - 2n - 5}{n+3} \quad 2) \begin{cases} u_0 = 5 \\ u_{n+1} = \frac{10}{u_n + 2} \end{cases} \quad 3) \begin{cases} u_0 = 5 \\ u_{n+1} = \frac{u_n + 3}{4u_n} \end{cases}$$

Ex 21 : Étudier la monotonie

Dans chaque cas, étudier la monotonie de la suite (u_n) .

- 1) $u_0 = 1$ et $u_{n+1} = u_n + n^2 - 3n + 5$
- 2) $u_n = n \times \left(\frac{1}{2}\right)^n$
- 3) $u_n = 1^2 + 2^2 + \dots + n^2$
- 4) $u_0 = 5$ et $u_{n+1} = u_n - 2n$
- 5) $u_n = 1 \times 2 \times 3 \times \dots \times n$
- 6) $u_n = \frac{n^2}{3^n}$
- 7) $u_n = n^3 - 12n^2 + 45n$ (Aide : étudier une fonction)
- 8) $u_n = \frac{n^2 - 4}{n^2 + 1}$ (Aide : étudier une fonction)
- 9) (u_n) est la suite des aires d'un rectangle dont les côtés mesurent n et $3,5 - n$.
- 10) (u_n) est la suite des coefficients directeurs des droites d_n d'équation $3x - ny + 5n = 0$

Ex 22 : Suites bornées

Dans chacun des cas, indiquer si la suite est minorée, majorée ou bornée.

- 1) $u_n = 4\left(\frac{2}{3}\right)^n - 1$ 2) $u_n = \frac{(-1)^n}{5} + 4$ 3) $u_n = \frac{n^2}{n^2 + 2}$
- 4) $u_n = 3 - 4\sin(5n)$ 5) $u_n = \frac{3^n}{4} - 1$ 6) $u_n = 1 - \sqrt{4 - \frac{1}{n^2}}$ (pour $n > 1$)
- 7) $u_n = \left(1 - \frac{2}{n}\right)\left(3 - \frac{4}{n}\right)$ (pour $n > 1$) 8) $u_n = \frac{3 + \sin n}{2 - \sin n}$

Ex 23 : Une peu de logique

Soit P la proposition : « la suite (u_n) , définie pour tout entier naturel n , est minorée par 5 »

Laquelle des propositions suivantes est la négation de P ?

- Q₁ : « (u_n) est minorée par 4 »
- Q₂ : « $\forall n \in \mathbb{N}, u_n < 5$ »
- Q₃ : « (u_n) est majorée par 5 »
- Q₄ : « $\exists n \in \mathbb{N}$, tel que $u_n < 5$ »

Ex 24 : Utilisation d'une autre suite

Soit $(u_n)_{n \geq 0}$ la suite définie sur \mathbb{N}^* par $u_1 = 8$ dont on donne les premiers termes : 8, 10, 13, 17, 22, 28 ... et dont la formule de récurrence est l'une des quatre suivantes :

$$u_{n+1} = u_n + 2n - 1, \quad u_{n+1} = u_n + 3n - 1, \quad u_{n+1} = u_n^2 - 7u_n + 8, \quad u_{n+1} = u_n + n + 1$$

- 1) a) Sélectionner la bonne formule de récurrence que l'on notera (R).
- b) Montrer que la suite (v_n) définie, pour tout entier naturel $n \geq 1$, par $v_n = \frac{1}{2}n^2 + \frac{1}{2}n$, vérifie la relation (R) mais que la suite (v_n) n'est pas la suite (u_n) .
- 2) Pour tout entier naturel $n \geq 1$, on pose $w_n = u_n - v_n$.
 - a) Démontrer que la suite (w_n) est constante.
 - b) En déduire une expression de u_n en fonction de n .
 - c) Démontrer que (u_n) est croissante.
 - d) Déterminer le plus petit entier naturel n tel que $u_n > 1000$.

Ex 25 : Représentation graphique et limite

Donner un exemple de représentation possible d'une suite :

- 1) Croissante qui a pour limite 5.
- 2) Croissante qui a pour limite $+\infty$.
- 3) Décroissante qui a pour limite 5.
- 4) Décroissante qui a pour limite $-\infty$.
- 5) Non monotone qui a pour limite 5.
- 6) Non monotone qui n'a pas de limite.

Ex 26 : Algorithme

algorithme a	Traduction en python
Fonction f(x) retourner -x ⁴ +2*x ³ -x+5 Lire n u[0]=f(0) Pour i allant de 0 à n-1 u[i+1] ← f(i+1) Si (u[i]>u[i+1]) alors afficher ("non") Sinon afficher ("oui") finSi FinPour	def f(x): return -x**4+2*x**3-x+5 n=int(input("n=")) u=[] u.append(f(0)) for i in range(0,n): u.append(f(i+1)) if (u[i]>u[i+1]): print("non") else: print("oui") (Faire tourner le programme)
algorithme b Fonction f(x) retourner -x ⁴ +2*x ³ -x+5 Lire n Lire p u[0]← p Pour i allant de 0 à n-1 u[i+1] ← f(u[i]) Si (u[i]>u[i+1]) alors afficher ("non") Sinon afficher ("oui") finSi FinPour	def f(x): return -x**4+2*x**3-x+5 n=int(input("n=")) u0=float(input("u0=")) u=[] u.append(u0) for i in range(0,n): u.append(f(u[i])) if (u[i]>u[i+1]): print("non") else: print("oui") (Faire tourner le programme)

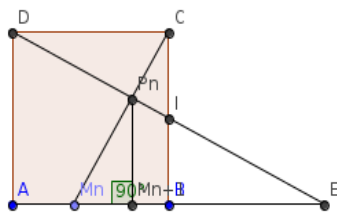
- 1) a) Quel est le rôle de chacun des algorithmes a et b ? Préciser la signification des messages « oui » et « non ».
- b) Que doit afficher l'algorithme pour que la suite soit croissante jusqu'au rang n ?
- 2) Tester l'un des deux algorithmes pour répondre aux questions suivantes :
 - a) Que peut-on dire de la suite de terme général $u_n = -n^4 + 2n^3 - n + 5$ jusqu'au rang 10 ? 20 ? 30 ?
 - b) Que peut-on dire de la suite définie par $\begin{cases} u_0 = 0 \\ u_{n+1} = \frac{1}{2 - u_n} \end{cases}$ jusqu'au rang 8 ? 16 ? 32 ? (On admet que $u_n \in [0; 1[$) (Faire tourner le programme)
 - c) Que peut-on dire de la suite de terme général $u_n = \frac{n^8}{2^n}$; jusqu'au rang 8 ? 12 ? 16 ? (Faire tourner le programme)

Ex 27 : Une conjecture avec GeoGebra (consulter suites_geo27.html)

ABCD est un carré de côté 1 et I est le milieu de [BC].

On considère une suite de points M_n définie de la manière suivante :

$M_0 = A$ et pour tout entier naturel n , M_{n+1} est le projeté orthogonal sur [AB] du point d'intersection P_n des droites (CM_n) et (DI) .



1) À l'aide de Geogebra, faire une figure et visualiser la positions des points M_0 , M_1 et M_2 .

2) Pour tout entier naturel n , on pose $u_n = AM_n$.

a) En déplaçant le point M_n ; conjecturer géométriquement la limite de (u_n) .

b) Démontrer que $u_0 = 0$ et que, pour tout entier naturel n , $u_{n+1} = \frac{2}{3 - u_n}$

c) À l'aide d'un tableur ou d'une calculatrice, vérifier (sans démontrer) la conjecture effectuée en 2) a).

Ex 28 : Une conjecture avec un tableur

Soit (u_n) la suite définie sur \mathbb{N} par $\begin{cases} u_0 = -1 \\ u_{n+1} = \frac{-u_n - 4}{u_n + 3} \end{cases}$.

1) a) À l'aide d'un tableur, donner les 40 premiers termes de la suite.

b) Représenter graphiquement le nuage de points $(n; u_n)$.

c) En observant le nuage de points, quelles conjectures peut-on faire sur le comportement de cette suite ?

2) a) Compléter le tableau de valeurs en faisant figurer le calcul de

$$v_n = \frac{3}{u_n + 2} \text{ . Que remarque-t-on ?}$$

b) Conjecturer une expression explicite de u_n en fonction de n .

3) Vérifier qu'avec la formule conjecturée en 2) b) :

a) on a bien $\begin{cases} u_0 = -1 \\ u_{n+1} = \frac{-u_n - 4}{u_n + 3} \end{cases}$

b) on peut justifier le sens de variation de (u_n) conjecturé dans 1) c)

c) on voit plus facilement la limite de (u_n) conjecturée dans 1) c)

Ex 29 : Valeurs approchées de \sqrt{x}

On va comparer deux méthodes d'obtention de valeurs approchées de $\sqrt{17}$, puis de manière plus générale de \sqrt{x} où x est un entier naturel.

Question A) Méthode de dichotomie

On construit une suite croissante (a_n) et une suite décroissante (b_n) telles que, pour tout entier naturel n , $a_n < \sqrt{17} < b_n$, en partageant, après chaque étape n , l'intervalle contenant $\sqrt{17}$ en deux intervalles de même amplitude, et en choisissant celui qui contient $\sqrt{17}$.

Étape 0 : On sait que $4 < \sqrt{17} < 5$, donc on prend $a_0 = 4$ et $b_0 = 5$.

Étape 1 : On partage l'intervalle [4;5] en deux intervalles [4;4,5] et [4,5;5].

Comme $4,5^2 > 17$, on choisit l'intervalle [4;4,5] qui contient $\sqrt{17}$.

Ainsi on prend $a_1 = 4$ et $b_1 = 4,5$.

On réitère un certain nombre n de fois l'opération jusqu'à obtenir une précision p telle que $b_n - a_n \leq p$.

a) Déterminer les premiers termes des suites (a_n) et (b_n) jusqu'au rang 3.

b) Compléter l'algorithme ci-dessous

algorithme a	Traduction en python
afficher ("Valeur approchée de racine de ") Lire x lire a[0] lire b[0] n ← lire p Tant que (b[n]-a[n]) faire m ← (a[n]+b[n])/2 Si (m*m<) alors a[n+1] ← b[n+1] ← FinSi Sinon a[n+1] ← b[n+1] ← FinSinon n ← n+1 Fin Tant que afficher ("a[",n,"]=",a[n]) afficher ("b[",n,"]=",b[n])	<pre> x=float(input("Valeur approchée de racine de=")) a0=float(input("a0=")) b0=float(input("b0=")) p=float(input("p=")) n=0 a=[] b=[] a.append(a0) b.append(b0) while (b[n]-a[n]>p): m=(a[n]+b[n])/2 if (m**2<x): a.append(m) b.append(b[n]) else: a.append(a[n]) b.append(m) n=n+1 print("a[",n,"]= ",a[n]) print("b[",n,"]= ",b[n]) </pre> <p>(Faire tourner le programme)</p>

c) Utiliser le programme pour obtenir une valeur approchée de $\sqrt{17}$, $\sqrt{13}$, et $\sqrt{23}$ avec une précision de 10^{-5} puis de 10^{-8} .

Question B) Méthode de Héron (consulter suites_geo29.ggb)

On considère un rectangle d'aire 17 et dont les côtés mesurent u_0 et $\frac{17}{u_0}$.

On prend ici pour u_0 la partie entière de $\sqrt{17}$. Pour rendre ce rectangle « un peu plus carré », on construit le rectangle de même aire ayant pour longueur d'un côté u_1 , égale à la moyenne des deux mesures du rectangle

$$\text{précédent : } u_1 = \frac{1}{2} \left(u_0 + \frac{17}{u_0} \right)$$

En réitérant indéfiniment l'opération, on construit la suite (u_n) de longueurs de rectangles définie par $u_0 = 4$ et

$$u_{n+1} = \frac{1}{2} \left(u_n + \frac{17}{u_n} \right) \text{ qui va tendre vers } \sqrt{17} \text{ c'est à dire}$$

vers la longueur du côté du carré d'aire 17.

1) Compléter l'algorithme ci-dessous qui donne une valeur approchée de \sqrt{x} avec une précision p telle que $|u_n - \sqrt{x}| \leq p$.

algorithme a	Traduction en python
afficher ("Valeur approchée de racine de ") Lire x u[0] ← n ← lire p Tant que (.....) faire u[n+1] ← n ← afficher ("u[",n,"]= ",u[n])	<pre> from math import sqrt, floor x=float(input("Valeur approchée de racine de=")) p=float(input("p=")) u0=float(input("u0=")) n=0 u=[] u.append(floor(sqrt(x))) while (abs(u[n]-sqrt(x))>p): u.append(0.5*(u[n]+x/u[n])) n=n+1 print("u[",n,"]= ",u[n]) </pre> <p>(Faire tourner le programme)</p>

c) Utiliser le programme pour obtenir une valeur approchée de $\sqrt{17}$, $\sqrt{13}$, et $\sqrt{23}$ avec une précision de 10^{-5} puis de 10^{-8} .

d) Quelle critique peut-on faire au sujet de cet algorithme ? (Une histoire de poule et d'oeuf !)

Question C) Quelle est la méthode la plus performante ?

