
2OS Mathématiques

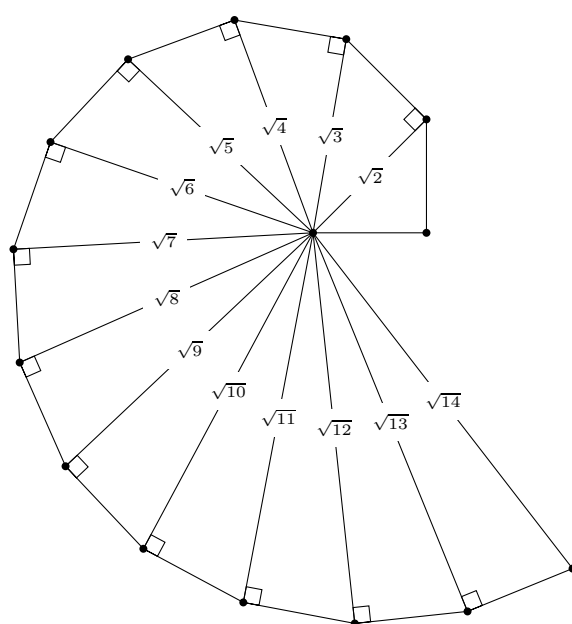


Table des matières

1	Sommes et récurrence	5
1.1	Sommes	5
1.2	Récurrence	8
1.3	Solutions des exercices	12
2	Suites de nombres réels	19
2.1	Exemples d'introduction	19
2.2	Suites dans $\mathbb{R}$	20
2.3	Suites arithmétiques et géométriques	23
2.4	Solutions des exercices	27
3	Cryptographie asymétrique	31
3.1	Plus grand diviseur commun et plus petit multiple commun	31
3.2	Algorithme d'Euclide	32
3.3	Algorithme d'Euclide étendu	33
3.4	Exponentiation modulaire	34
3.5	Théorèmes de Fermat et d'Euler	36
3.6	Factoriser un nombre entier	38
3.7	RSA	39
3.8	Solutions des exercices	43
4	Méthodes numériques	59
4.1	La bibliothèque matplotlib de Python : les fonctions de base	59
4.2	Zéros de fonctions : méthode de la bisection	61
4.3	Zéros de fonctions : méthode de Newton	62
4.4	Zéros de fonctions : méthode de la sécante	63
4.5	Un peu d'intégration numérique	63
4.6	Solutions des exercices	68

Chapitre 1

Sommes et récurrence

1.1 Sommes

1.1.1 On donne

$$x_1 = 3, \quad x_2 = 5, \quad x_3 = 6, \quad x_4 = 2 \quad \text{et} \quad x_5 = 7.$$

Calculer :

$$\text{a) } \sum_{i=1}^5 x_i$$

$$\text{c) } \sum_{k=1}^5 x_k$$

$$\text{e) } \sum_{i=1}^5 (x_i + 8)$$

$$\text{b) } \sum_{i=2}^4 x_i$$

$$\text{d) } \sum_{j=1}^5 x_j^3$$

$$\text{f) } \sum_{k=1}^5 (8 \cdot x_k)$$

1.1.2 On donne

$$x_1 = 3, \quad x_2 = 5, \quad x_3 = 6, \quad x_4 = 2 \quad \text{et} \quad x_5 = 7.$$

On donne également

$$y_1 = 2, \quad y_2 = 8, \quad y_3 = 3, \quad y_4 = 1 \quad \text{et} \quad y_5 = 6.$$

Calculer :

$$\text{a) } \sum_{i=1}^5 (x_i + y_i)$$

$$\text{d) } \sum_{j=1}^5 (2 \cdot x_j)$$

$$\text{g) } \sum_{i=1}^5 (2 \cdot x_i) + \sum_{j=1}^5 (3 \cdot y_j)$$

$$\text{b) } \sum_{i=1}^5 (x_i - y_i)$$

$$\text{e) } \sum_{j=1}^5 (x_j + y_j)^2$$

$$\text{h) } \sum_{j=1}^5 x_j^2 - \left(\sum_{j=1}^5 y_j \right)^2$$

$$\text{c) } \sum_{k=1}^5 (x_k \cdot y_k)$$

$$\text{f) } \sum_{i=1}^5 8$$

$$\text{i) } \sum_{i=1}^4 (x_{i+1} + y_i)$$

1.1.3 Développer les sommes suivantes

$$\text{a) } \sum_{i=0}^n (-1)^i \quad \text{b) } \sum_{j=0}^n 3j \quad \text{c) } \sum_{k=1}^n n \quad \text{d) } \sum_{i=1}^n \frac{n}{i}$$

1.1.4 Développer chacune des sommes écrites à l'aide du symbole Σ , en faisant disparaître ce symbole :

$$\text{a) } \sum_{k=3}^{10} \frac{1}{k^2} \quad \text{b) } \sum_{k=1}^{10} \frac{1}{2k+1} \quad \text{c) } \sum_{k=1}^n \frac{(k+1)!}{k}$$

1.1.5 Développer et calculer les sommes suivantes :

$$\begin{array}{lll} \text{a) } \sum_{i=0}^n 1 & \text{d) } \sum_{m=0}^3 (m^2 - 6m + 9) & \text{g) } \sum_{k=1}^5 \frac{k+2}{k} \\ \text{b) } \sum_{j=1}^{2011} (-1)^j & \text{e) } \sum_{l=1}^5 4l(l^2 - 1) & \text{h) } \sum_{k=1}^6 \frac{1}{3^k} \\ \text{c) } \sum_{k=-n}^n (k+1) & \text{f) } \sum_{i=0}^4 \left(2^i + \left(\frac{1}{2} \right)^i \right) & \text{i) } \sum_{j=-2}^2 \frac{2^{j+3}}{j^2 + 1} \end{array}$$

1.1.6 Soit $x_1, \dots, x_n$ une suite de nombres réels. Calculer $\sum_{i=2}^n (x_i - x_{i-1})$ **1.1.7** Traduire à l'aide du symbole Σ les sommes suivantes :

$$\begin{array}{ll} \text{a) } 1^2 + 2^2 + 3^2 + \dots + 12^2 + 13^2 + 14^2 & \\ \text{b) } 11^2 + 12^2 + 13^2 + \dots + 102^2 + 103^2 + 104^2 & \\ \text{c) } \frac{1}{2} + \frac{2}{3} + \frac{3}{4} + \dots + \frac{13}{14} + \frac{14}{15} + \frac{15}{16} & \\ \text{d) } 1^2 + 3^2 + 5^2 + \dots + 11^2 + 13^2 + 15^2 & \\ \text{e) } (1 \cdot 3) + (2 \cdot 4) + (3 \cdot 5) + (4 \cdot 6) + (5 \cdot 7) & \\ \text{f) } 1 + 8 + 27 + 64 + 125 & \end{array}$$

1.1.8 Ecrire les sommes suivantes à l'aide du symbole Σ .

$$\begin{array}{ll} \text{a) } 2 + 4 + 6 + \dots + 248 = & \text{e) } 2 + 3 + 5 + 9 + 17 + \dots + 1025 = \\ \text{b) } 1000 + 1010 + 1020 + \dots + 1540 = & \text{f) } 4 + 12 + 36 + 108 + 324 = \\ \text{c) } 1^2 + 2^2 + 3^2 + \dots + 15^2 = & \text{g) } 9 - 12 + 15 - 18 + \dots + 303 = \\ \text{d) } 1 + 2 + 4 + 8 + 16 + \dots + 1024 = & \text{h) } 45 - 40 + 35 - 30 + 25 - 20 + 15 = \end{array}$$

$$\text{i)} \quad 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \cdots + \frac{1}{n} =$$

$$\text{k)} \quad \frac{1}{2} + \frac{4}{5} + \frac{9}{10} + \cdots + \frac{10000}{10001} =$$

$$\text{l)} \quad \frac{5}{3} + \frac{10}{7} + \frac{15}{11} + \frac{20}{15} + \cdots + \frac{70}{55} =$$

$$\text{j)} \quad \frac{1}{2} + \frac{2}{3} + \frac{3}{4} + \cdots + \frac{9}{10} + 32 =$$

$$\text{m)} \quad (1 \cdot 3) + (2 \cdot 4) + (3 \cdot 5) + (4 \cdot 6) + (5 \cdot 7) =$$

1.1.9 Traduire à l'aide du symbole Σ la somme suivante :

$$1000 - \frac{3}{1 \cdot (1 + 3 + 5)} + \frac{5}{(1 + 3) \cdot (1 + 3 + 5 + 7)} - \frac{7}{(1 + 3 + 5) \cdot (1 + 3 + 5 + 7 + 9)} + \cdots$$

1.1.10 Est-ce que $A = B = C = D$ si on définit

$$A = \sum_{i=1}^n i \cdot n \quad B = \sum_{k=1}^n k \cdot n \quad C = n \cdot \sum_{k=1}^n k \quad D = k \cdot \sum_{i=0}^n n \quad ?$$

1.1.11 Ecrire les sommes suivantes en faisant en sorte que la première valeur de l'indice soit 0, puis 49 :

$$\text{a)} \quad \sum_{i=2}^{45} 1 = \sum_{i=0}^{\dots} \dots = \sum_{i=49}^{\dots} \dots$$

$$\text{b)} \quad \sum_{i=10}^{20} i$$

$$\text{c)} \quad \sum_{k=-4}^{180} \frac{k}{k+5}$$

Faire calculer le résultat de chaque somme à l'aide d'une boucle `while` en python.

1.1.12 Changer l'indice et ses bornes de sorte que le terme général soit plus simple :

$$\text{a)} \quad \sum_{i=0}^n \frac{(i+1)^2 + 3}{1 + \sqrt{i+1}} \quad \text{b)} \quad \sum_{j=3}^{n+2} \frac{x^{j-3}}{(j-3)^x} \quad \text{c)} \quad \sum_{k=1}^{n+1} (k-1) 2^k + 3^{k-1}$$

1.1.13 On considère une suite de n nombres $x_1, x_2, x_3, \dots, x_n$. Écrire les expressions suivantes de la manière la plus synthétique possible à l'aide d'un symbole de sommation.

- La somme des termes de cette suite de nombres.
- La somme des carrés des termes de cette suite de nombres
- Le carré de la somme des termes de cette suite de nombres
- La différence entre la somme des carrés et le carré de la somme des termes de cette suite de nombre

1.1.14 On considère deux suites de n nombres $x_1, x_2, x_3, \dots, x_n$ et $y_1, y_2, y_3, \dots, y_n$. Écrire les expressions suivantes de la manière la plus synthétique possible.

- Produit de la somme des x_i et de la somme des y_i

- b) Somme des produits $x_i y_i$
- c) Somme des y_i moins k fois la somme des x_i

1.1.15 Ecrire un programme qui fait calculer les sommes suivantes :

- a) $\sum_{i=1}^{100000} (2 + 3i)$
- b) $\sum_{i=-237}^{325} i^2$
- c) $\sum_{j=0}^{100} 2^j$
- d) $\sum_{k=1}^{10^6} (2k - 1)$

1.1.16 Ecrire un programme qui fait calculer les sommes suivantes :

- a) $\sum_{n=1}^{23} \sum_{k=1}^{37} k^n$
- b) $\sum_{i=1}^{100} \sum_{j=1}^{100} 1$
- c) $\sum_{n=1}^{20} \sum_{j=1}^{10} (n^j + 1)$
- d) $\sum_{n=-10}^{10} \sum_{k=-n}^n k$

1.1.17 Soit $x_1, x_2, x_3, \dots, x_n$ des nombres réels et $m = \frac{1}{n} \sum_{i=1}^n x_i$. Montrer que :

$$\frac{1}{n} \sum_{i=1}^n (x_i - m)^2 = \frac{1}{n} \sum_{i=1}^n x_i^2 - m^2$$

1.1.18 Soit $a_1, a_2, a_3, \dots, a_n$ des nombres. Démontrer l'égalité ci-dessous.

$$\left(\sum_{i=1}^n a_i \right)^2 = \sum_{i=1}^n \sum_{j=1}^n a_i \cdot a_j$$

1.1.19 Soit $a_1, a_2, a_3, \dots, a_n$ des nombres. Démontrer l'égalité ci-dessous par récurrence.

$$\left(\sum_{i=1}^n a_i \right)^2 = \sum_{i=1}^n a_i^2 + 2 \cdot \sum_{i=1}^{n-1} \sum_{j=i+1}^n a_i \cdot a_j$$

1.2 Récurrence

La démonstration par récurrence s'applique dans le cas où on doit démontrer une proposition P_n pour tout entier $n \geq 0$ ou $n \geq 1$ ou plus généralement $n \geq n_0$ ($n_0 \in \mathbb{N}$).

Une démonstration par récurrence se fait toujours en deux étapes :

- a) Première phase, dite d'initialisation : on démontre que la proposition est vraie pour la plus petite valeur de l'entier n : 0, 1 ou n_0 .
- b) Deuxième phase : on suppose que la proposition P_n est vraie et on démontre sous cette hypothèse, appelée hypothèse de récurrence, que la proposition P_{n+1} est vraie.

D'un point de vue formel, on peut écrire :

$$\begin{cases} P_0 \text{ est vrai} \\ n \geq 0 \text{ et } P_n \Rightarrow P_{n+1} \end{cases} \Rightarrow \forall n \geq 0 : P_n \text{ est vrai}$$

1.2.1 Démontrer par récurrence que, $\forall n \in \mathbb{N}^*$:

$$\begin{array}{ll} \text{a) } \sum_{k=1}^n k = \frac{n(n+1)}{2} & \text{f) } \sum_{k=1}^n \frac{k^2}{(2k-1)(2k+1)} = \frac{n(n+1)}{2(2n+1)} \\ \text{b) } \sum_{k=1}^n k^2 = \frac{n(n+1)(2n+1)}{6} & \text{g) } \sum_{k=1}^n \frac{k}{2^k} = 2 - \frac{n+2}{2^n} \\ \text{c) } \sum_{k=1}^n k^3 = \frac{n^2(n+1)^2}{4} & \text{h) } \sum_{k=1}^n k \cdot 5^k = \frac{5 + (4n-1)5^{n+1}}{16} \\ \text{d) } \sum_{k=1}^n (-1)^{k+1} k^2 = (-1)^{n+1} \frac{n(n+1)}{2} & \text{i) } \sum_{k=1}^n \frac{1}{k(k+1)(k+2)} = \frac{n(n+3)}{4(n+1)(n+2)} \\ \text{e) } \sum_{k=1}^n \frac{1}{(2k-1)(2k+1)} = \frac{n}{2n+1} \end{array}$$

1.2.2 Démontrer par récurrence que, $\forall n \in \mathbb{N}$:

- a) $8^n - 1$ est divisible par 7
- b) $3^{2n+2} - 2^{n+1}$ est un multiple de 7
- c) $10^{6n+2} + 10^{3n+1} + 1$ est un multiple de 111
- d) $n^3 + 5n$ est divisible par 3
- e) $\frac{2}{3}n^3 + n^2 + \frac{1}{3}n$ est un nombre pair
- f) $11^{n+2} + 12^{2n+1}$ est divisible par 133

1.2.3 Soit $n \in \mathbb{N}$. Prouver par récurrence puis directement que 3 divise $n^3 - n$.

1.2.4 Établir une formule pour la somme

$$\frac{1}{1 \cdot 2} + \frac{1}{2 \cdot 3} + \frac{1}{3 \cdot 4} + \cdots + \frac{1}{n(n+1)}$$

1.2.5 Établir une formule pour la somme

$$1 + \frac{1}{1+2} + \frac{1}{1+2+3} + \cdots + \frac{1}{1+2+3+\cdots+n}$$

1.2.6 Écrire en fonction de n la somme des carrés des n premiers nombres impairs.

1.2.7 Démontrer que « $\forall n \in \mathbb{N}, n^2 - n + 41$ est premier » est une proposition fausse.

1.2.8 Démontrer que « $\forall n \in \mathbb{N}^*, (n \text{ est premier} \Rightarrow 2^n - 1 \text{ est premier})$ » est une proposition fausse.

1.2.9 Trouver l'erreur commise dans la démonstration ci-dessous :

On note P_n l'assertion « $\forall n \in \mathbb{N}^*, (6 \text{ divise } 7^n + 1)$ ».

Prouvons que 6 divise $7^n + 1$. Supposons que P_n est vraie. Alors il existe k tel que $7^n + 1 = 6k$. Ainsi, $7^{n+1} + 1 = 7 \cdot 7^n + 1 = 7(6k - 1) + 1 = 42k - 6 = 6(7k - 1)$. Par conséquent 6 divise $7^{n+1} + 1$, i.e. P_{n+1} est vraie. Par récurrence, P_n est vraie pour tout n .

1.2.10 Trouver l'erreur commise dans la démonstration ci-dessous :

Prouvons que tout le monde a la même taille. On pose $P_n =$ « n personnes ont toujours la même taille ». Cette fois on n'oublie pas d'initialiser : une personne a la même taille qu'elle-même, donc P_1 est vraie.

L'hérédité ensuite. Soit $n > 1$ et supposons que P_n est vraie. On considère alors $n + 1$ personnes. Les personnes 1 jusqu'à n ont la même taille, par hypothèse de récurrence. De même, les personnes 2 jusqu'à $n + 1$ ont la même taille. Ainsi, ces $n + 1$ personnes ont la même taille.

1.2.11 Démontrer par récurrence :

a) Pour tout nombre réel positif x , la propriété suivante est vérifiée :

$$\forall n \in \mathbb{N}, (1 + x)^n \geq 1 + nx$$

b) Pour tout $a \in \mathbb{R}$ et $r \in \mathbb{R} - \{1\}$, on a :

$$\forall n \in \mathbb{N}^*, \quad a + ar + ar^2 + \dots + ar^{n-1} = \frac{a(r^n - 1)}{r - 1}$$

c) Pour tout $\alpha \in \mathbb{R} - \{k\pi \mid k \in \mathbb{Z}\}$, on a :

$$\forall n \in \mathbb{N}, \quad \cos(\alpha) \cdot \cos(2\alpha) \cdot \cos(4\alpha) \cdot \dots \cdot \cos(2^n \alpha) = \frac{\sin(2^{n+1} \alpha)}{2^{n+1} \sin(\alpha)}$$

d) Pour des nombres réels strictement positifs $x_1, x_2, x_3, \dots$, on a :

$$\forall n \in \mathbb{N}, \quad (x_1 + x_2 + \dots + x_n) \cdot \left(\frac{1}{x_1} + \frac{1}{x_2} + \dots + \frac{1}{x_n} \right) \geq n^2$$

1.2.12 Démontrer par récurrence :

a) $\forall n \in \mathbb{N}^*, (n + 1)! > 1! + 2! + 3! + \dots + n!$

b) $\forall n \in \mathbb{N} - \{0, 1, 2, 3\}, n! > 2^n$

1.2.13 Soit (u_n) la suite de Fibonacci définie comme suit :

$$\begin{cases} u_1 = 1 \\ u_2 = 1 \\ u_{n+1} = u_n + u_{n-1}, \quad \text{pour } n \geq 2 \end{cases}$$

- a) Démontrer que $\forall n \in \mathbb{N}^*$, $u_1^2 + u_2^2 + \dots + u_n^2 = u_n \cdot u_{n+1}$
- b) Calculer l'expression $u_n^2 - u_{n-1} \cdot u_{n+1}$ pour quelques valeurs de n .
- c) Démontrer que $\forall n \in \mathbb{N}^*$, $u_n = \frac{(1 + \sqrt{5})^n - (1 - \sqrt{5})^n}{\sqrt{5} \cdot 2^n}$

1.3 Solutions des exercices

1.1.1

$$\text{a) } \sum_{i=1}^5 x_i = 23$$

$$\text{c) } \sum_{k=1}^5 x_k = 23$$

$$\text{e) } \sum_{i=1}^5 (x_i + 8) = 63$$

$$\text{b) } \sum_{i=2}^4 x_i = 13$$

$$\text{d) } \sum_{j=1}^5 x_j^3 = 719$$

$$\text{f) } \sum_{k=1}^5 (8 \cdot x_k) = 184$$

1.1.2

$$\text{a) } \sum_{i=1}^5 (x_i + y_i) = 43$$

$$\text{f) } \sum_{i=1}^5 8 = 40$$

$$\text{b) } \sum_{i=1}^5 (x_i - y_i) = 3$$

$$\text{g) } \sum_{i=1}^5 (2 \cdot x_i) + \sum_{j=1}^5 (3 \cdot y_j) = 106$$

$$\text{c) } \sum_{k=1}^5 (x_k \cdot y_k) = 108$$

$$\text{h) } \sum_{j=1}^5 x_j^2 - \left(\sum_{j=1}^5 y_j \right)^2 = -277$$

$$\text{d) } \sum_{j=1}^5 (2 \cdot x_j) = 46$$

$$\text{i) } \sum_{i=1}^4 (x_{i+1} + y_i) = 34$$

$$\text{e) } \sum_{j=1}^5 (x_i + y_i)^2 = 453$$

1.1.3

a) 1 si n est pair ; 0 si n est impair

b) $3 + 6 + 9 + \dots + 3n$

c) n^2

d) $n + \frac{n}{2} + \frac{n}{3} + \dots + \frac{n}{n-1} + 1$

1.1.4 –

1.1.5

a) $n + 1$

b) -1

c) $2n + 1$

d) $\sum_{m=0}^3 (m-3)^2 = 14$

e) $4 \cdot 0 + 8 \cdot 3 + 12 \cdot 8 + 16 \cdot 15 + 20 \cdot 24 = 840$

f) $1 + 1 + 2 + \frac{1}{2} + 4 + \frac{1}{4} + 8 + \frac{1}{8} + 16 + \frac{1}{16} = \frac{527}{16}$

g) $3 + 2 + \frac{5}{3} + \frac{3}{2} + \frac{7}{5} = \frac{287}{30}$

h) $\frac{1}{3} + \frac{1}{9} + \frac{1}{27} + \frac{1}{81} + \frac{1}{243} + \frac{1}{729} = \frac{364}{729}$

i) $\frac{2}{5} + \frac{4}{2} + \frac{8}{1} + \frac{16}{2} + \frac{32}{5} = \frac{124}{5}$

1.1.6 $\sum_{i=2}^n (x_i - x_{i-1}) = x_n - x_1$

1.1.7

a) $1^2 + 2^2 + 3^2 + \cdots + 12^2 + 13^2 + 14^2 = \sum_{i=1}^{14} i^2 = \sum_{i=0}^{13} (i+1)^2$

b) $11^2 + 12^2 + 13^2 + \cdots + 102^2 + 103^2 + 104^2 = \sum_{i=11}^{104} i^2$

c) $\frac{1}{2} + \frac{2}{3} + \frac{3}{4} + \cdots + \frac{13}{14} + \frac{14}{15} + \frac{15}{16} = \sum_{i=1}^{15} \frac{i}{i+1}$

d) $1^2 + 3^2 + 5^2 + \cdots + 11^2 + 13^2 + 15^2 = \sum_{i=1}^8 (2i-1)^2$

e) $(1 \cdot 3) + (2 \cdot 4) + (3 \cdot 5) + (4 \cdot 6) + (5 \cdot 7) = \sum_{i=1}^5 i(i+2)$

f) $1 + 8 + 27 + 64 + 125 = \sum_{i=1}^5 i^3$

1.1.8

a) $\sum_{i=1}^{124} 2i$

c) $\sum_{k=1}^{15} k^2$

b) $\sum_{i=100}^{154} 10i$

d) $\sum_{i=0}^{10} 2^i$

$$\text{e)} \sum_{i=0}^{10} (2^i + 1)$$

$$\text{j)} \sum_{i=1}^9 \frac{i}{i+1} + 32$$

$$\text{f)} \sum_{i=0}^4 4 \cdot 3^i$$

$$\text{k)} \sum_{i=1}^{100} \frac{i^2}{i^2 + 1}$$

$$\text{g)} \sum_{i=3}^{101} (-1)^{i+1} 3i$$

$$\text{l)} \sum_{i=1}^{14} \frac{5i}{4i-1}$$

$$\text{h)} \sum_{i=0}^6 (-1)^i (45 - 5i)$$

$$\text{m)} \sum_{j=1}^5 j(j+2)$$

$$\text{i)} \sum_{i=1}^n \frac{1}{i}$$

$$\mathbf{1.1.9} \quad 1000 + \sum_{i=1}^{\infty} \frac{(-1)^i (2i+1)}{\sum_{j=1}^i (2j-1) \sum_{k=1}^{i+2} (2k-1)}$$

$$\mathbf{1.1.10} \quad A = B = C = \frac{n^2(n+1)}{2} \neq D = kn(n+1)$$

1.1.11

$$\text{a)} \sum_{i=2}^{45} 1 = \sum_{i=0}^{43} 1 = \sum_{i=49}^{92} 1$$

$$\text{c)} \sum_{k=-4}^{180} \frac{k}{k+5} = \sum_{k=0}^{184} \frac{k-4}{k+1} = \sum_{k=49}^{233} \frac{k-53}{k-48}$$

$$\text{b)} \sum_{i=10}^{20} i = \sum_{i=0}^{10} (i+10) = \sum_{i=49}^{59} (i-39)$$

1.1.12

$$\text{a)} \sum_{i=0}^n \frac{(i+1)^2 + 3}{1 + \sqrt{i+1}} = \sum_{j=1}^{n+1} \frac{j^2 + 3}{1 + \sqrt{j}}$$

$$\text{c)} \sum_{k=1}^{n+1} (k-1) 2^k + 3^{k-1} = \sum_{l=0}^n l \cdot 2^{l+1} + 3^l$$

$$\text{b)} \sum_{j=3}^{n+2} \frac{x^{j-3}}{(j-3)^x} = \sum_{k=0}^{n-1} \frac{x^k}{k^x}$$

1.1.13

$$\text{a)} \sum x_i$$

$$\text{c)} (\sum x_i)^2$$

$$\text{b)} \sum (x_i)^2 = \sum x_i^2$$

$$\text{d)} \sum x_i^2 - (\sum x_i)^2$$

1.1.14

$$\text{a)} \left(\sum_i x_i \right) \left(\sum_i y_i \right) = \sum_i x_i \sum_j y_j = \sum_{i,j} x_i y_j$$

$$\text{b)} \sum_i x_i y_i$$

$$\text{c)} \sum y_i - k \sum x_i$$

1.1.15

```
def somme(f, n, m):
    s = 0
    for i in range(n, m + 1):
        s += f(i)
    return s

def a(i):
    return 2 + 3 * i

def b(i):
    return i**2

def c(j):
    return 2**j

def d(k):
    return 2*k - 1

print("a)", somme(a, 1, 100000))
print("b)", somme(b, -237, 325))
print("c)", somme(c, 0, 100))
print("d)", somme(d, 1, 10**6))
```

1.1.16

```
def somme(f, n_1, m_1, n_2, m_2):
    s = 0
    for i in range(n_1, m_1 + 1):
        for j in range(n_2, m_2 + 1):
            s += f(i, j)
    return s

def a(k, n):
    return k^n

def b(i, j):
    return 1
```

```
def c(n, j):  
    return n**j + 1  
  
print("a)", somme(a, 1, 23, 1, 37))  
print("b)", somme(b, 1, 100, 1, 100))  
print("c)", somme(c, 1, 20, 1, 10))  
  
d = 0  
  
for n in range(-10, 11):  
    for k in range(-n, n + 1):  
        d += k  
  
print("d)", d)
```

1.1.17

1.1.18

1.1.19

1.2.1 –

1.2.2 –

1.2.3 –

1.2.4 –

1.2.5 –

1.2.6 –

1.2.7 –

1.2.8 –

1.2.9 –

1.2.10 –

1.2.11 –

1.2.12 –

1.2.13 –

Chapitre 2

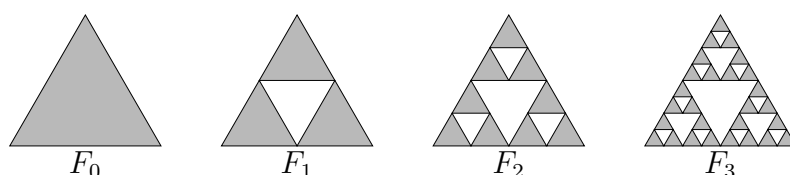
Suites de nombres réels

2.1 Exemples d'introduction

2.1.1 Un patient doit prendre une fois par jour un médicament composé de 12 mg de cermoxène. L'expérience a montré que le corps humain élimine en 24 heures la moitié de la dose de cermoxène qu'il contient.

Quelle sera la quantité de ce produit contenu dans le corps du patient 1 jour ? 2 jours ? 3 jours ? ... n jours après le début du traitement ?

2.1.2 On considère une suite de figures formées de triangles équilatéraux :

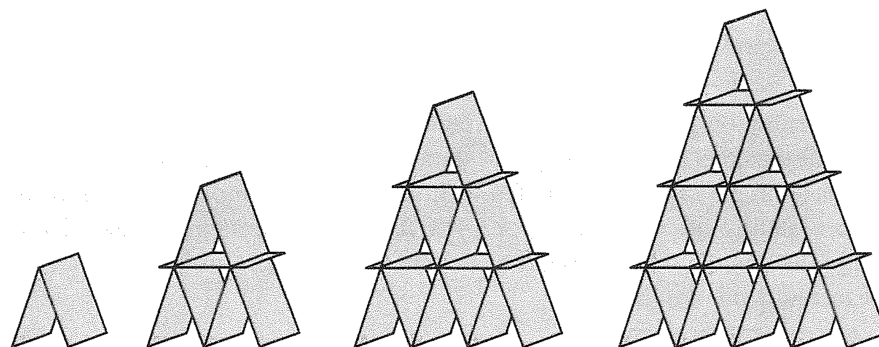


- Combien de triangles sont grisés dans F_n ?
- Sur quelle figure a-t-on pour la première fois plus de 1 000 000 triangles grisés ?
- Si l'aire grisée de F_0 est 1, quelle est l'aire grisée dans F_1 ? dans F_2 ?
Trouver une formule pour l'aire grisée dans F_n .
Vers quelle valeur tend l'aire grisée de F_n lorsque n devient très grand ?

2.1.3 Après avoir mis la machine à calculer en mode radian, introduire un nombre réel quelconque et appliquer à plusieurs reprises la fonction cosinus.

Observer la suite de nombres qui apparaît à l'écran. Commenter ce résultat.

2.1.4 La construction d'un château de cartes à n étages nécessite c_n cartes.



- Déterminer c_1, c_2, c_3, c_4 et c_5 .
- Trouver une relation de récurrence du type $c_{n+1} = c_n + f(n)$
- Trouver une formule explicite pour c_n et démontrer cette formule par récurrence.

2.2 Suites dans $\mathbb{R}$

2.2.1 Calculer les quatre premiers termes de la suite (u_n) définie par :

- $u_n = \frac{n}{n+2}$, avec $n \geq 1$
- $u_n = 1 + (-1)^n$, avec $n \geq 1$
- $u_n = n \sin\left(\frac{n\pi}{2}\right)$, avec $n \geq 1$
- $u_n = \sqrt{n^2 + 3n} - n$, avec $n \geq 1$

2.2.2 Calculer les quatre premiers termes des suites :

- $\begin{cases} a_1 = 1 \\ a_{n+1} = 2a_n + 1, \text{ si } n \geq 1 \end{cases}$
- $\begin{cases} a_1 = 3 \\ a_{n+1} = \frac{a_n - 2}{3}, \text{ si } n \geq 1 \end{cases}$
- $\begin{cases} a_1 = 2 \\ a_{n+1} = \frac{1}{2} \left(a_n + \frac{1}{a_n} \right), \text{ si } n \geq 1 \end{cases}$
- $\begin{cases} a_1 = 1 \\ a_{n+1} = \frac{1}{2} \left(a_n + \frac{1}{a_n} \right), \text{ si } n \geq 1 \end{cases}$

2.2.3 Trouver les termes généraux des suites :

- 4 7 10 13 16 ...
- 2 8 18 32 50 ...
- 1 $-\frac{1}{2}$ $\frac{1}{3}$ $-\frac{1}{4}$ $\frac{1}{5}$...
- $-\frac{1}{2}$ 0 $\frac{1}{4}$ $\frac{2}{5}$ $\frac{3}{6}$...
- 3 9 -27 81 -243 ...
- 1 $\sqrt{2}$ $\sqrt[3]{3}$ $\sqrt{2}$ $\sqrt[5]{5}$...

2.2.4 Les suites ci-dessous sont-elles minorées, majorées, bornées, croissantes, décroissantes ?

a) $a_n = 2 - \frac{1}{n^2}$, avec $n \geq 1$

b) $b_n = -n!$, avec $n \geq 1$

c) $c_n = 2n^2 - 7n$, avec $n \geq 2$

d) $d_n = \frac{n}{n^2 + 10}$, avec $n \geq 3$

e) $e_n = 3^{1+(-1)^n}$, avec $n \geq 1$

f) $\begin{cases} f_1 = 1 \\ f_{n+1} = 1 + 2f_n, \text{ si } n \geq 1 \end{cases}$

2.2.5 Soit la suite $(u_n)_{n \geq 1}$ et $\epsilon > 0$. Si on accepte que la suite converge vers L , trouver le plus petit entier positif N tel que $|u_n - L| < \epsilon$, pour tout $n > N$.

a) $u_n = \frac{3}{n-2}$, avec $L = 0$ et $\epsilon = 0,1$

b) $u_n = \frac{n}{n+1}$, avec $L = 1$ et $\epsilon = 0,25$

En considérant ϵ quelconque, démontrer que les suites ci-dessus sont convergentes.

2.2.6 Calculer :

a) $\lim_{n \rightarrow \infty} \frac{4n}{2n-1}$

b) $\lim_{n \rightarrow \infty} \frac{2n^2}{1-n^2}$

c) $\lim_{n \rightarrow \infty} \frac{5}{n^3}$

d) $\lim_{n \rightarrow \infty} \frac{3n + 2n(-1)^n}{n}$

e) $\lim_{n \rightarrow \infty} \frac{(2n-1)^4}{(1-n^2)^2}$

f) $\lim_{n \rightarrow \infty} \frac{7n^2}{\sqrt{n^4-5}}$

2.2.7 A l'aide du théorème des deux gendarmes, calculer :

a) $\lim_{n \rightarrow \infty} \frac{\cos(n)}{n^2}$

b) $\lim_{n \rightarrow \infty} \frac{(-1)^n}{n}$

c) $\lim_{n \rightarrow \infty} \frac{n \sin(n!)}{n^2 + 1}$

d) $\lim_{n \rightarrow \infty} \frac{n!}{n^n}$

e) $\lim_{n \rightarrow \infty} n \sin\left(\frac{1}{n^2}\right)$

f) $\lim_{n \rightarrow \infty} \frac{1}{n} E(nx)$, avec $x \in \mathbb{R}_+^*$

2.2.8 Calculer :

a) $\lim_{n \rightarrow \infty} \frac{\sqrt{n^2+2}}{2n}$

e) $\lim_{n \rightarrow \infty} \frac{\sqrt{n^2+1} - \sqrt{n^2+4}}{2}$

b) $\lim_{n \rightarrow \infty} n \sin \frac{1}{n^2}$

f) $\lim_{n \rightarrow \infty} \sqrt{n^2+7} - \sqrt{(n+3)(n+6)}$

c) $\lim_{n \rightarrow \infty} \frac{\cos \sqrt{n}}{n}$

g) $\lim_{n \rightarrow \infty} \frac{2^n}{n!}$

d) $\lim_{n \rightarrow \infty} \frac{\sin(n+1) - \sin(n-1)}{\sin n}$

h) $\lim_{n \rightarrow \infty} \frac{n}{a^n}$

2.2.9 Une suite récurrente est définie par $\begin{cases} u_1 = \sqrt{2} \\ u_{n+1} = \sqrt{2u_n}, \text{ si } n \geq 1 \end{cases}$.

- Montrer que la suite $(u_n)_{n \geq 1}$ est croissante et majorée par 2.
- Prouver que la suite $(u_n)_{n \geq 1}$ est convergente et calculer sa limite.

2.2.10 Une suite récurrente est définie par $\begin{cases} u_1 = 2 \\ u_{n+1} = \frac{u_n}{u_n + 1}, \text{ si } n \geq 1 \end{cases}$.

- Montrer que la suite $(u_n)_{n \geq 1}$ est strictement décroissante et minorée par 0.
- Prouver que la suite $(u_n)_{n \geq 1}$ est convergente et calculer sa limite.

2.2.11 Une suite récurrente est définie par $\begin{cases} u_1 = 1 \\ u_{n+1} = \sqrt{12 + u_n}, \text{ si } n \geq 1 \end{cases}$.

- Montrer que $u_n < 4$, pour tout $n \geq 1$.
- Montrer que $u_{n+1}^2 - u_n^2 = (4 - u_n)(3 + u_n)$, puis en déduire que la suite $(u_n)_{n \geq 1}$ est croissante.
- Prouver que la suite $(u_n)_{n \geq 1}$ est convergente et calculer sa limite.

2.2.12 Calculer :

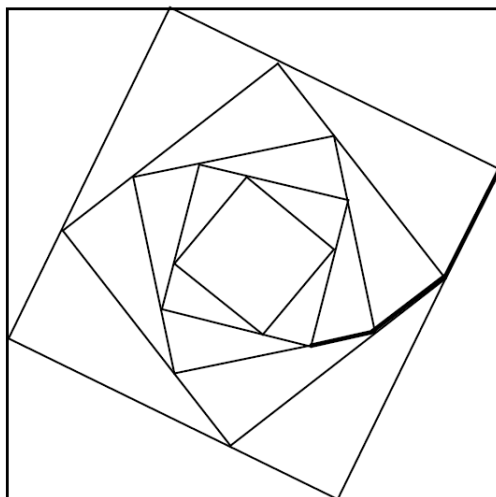
- | | |
|--|---|
| a) $\lim_{n \rightarrow \infty} \frac{2n^2 - 3}{4n}$ | b) $\lim_{n \rightarrow \infty} \left(n \cdot \frac{1}{1 - n} \right)$ |
| c) $\lim_{n \rightarrow \infty} \frac{3^n - 3^{n-1}}{2 + 3^n}$ | d) $\lim_{n \rightarrow \infty} (\sqrt{n^2 + n} - n)$ |
| e) $\lim_{n \rightarrow \infty} \left(n^2 + \frac{2n + n^2 - n^3}{n - 1} \right)$ | f) $\lim_{n \rightarrow \infty} \frac{1 + 2 + \dots + n}{n^2}$ |

2.2.13 Une suite récurrente est définie par $\begin{cases} a_1 = 3 \\ a_{n+1} = \frac{a_n + 12}{4}, \text{ si } n \geq 1 \end{cases}$.

- On pose $v_n = a_n - 4$. Démontrer que $(v_n)_{n \geq 1}$ est une suite géométrique.
- Donner le terme générale de la suite $(a_n)_{n \geq 1}$, puis calculer sa limite.

2.2.14 Dans un carré dont le côté vaut 1, on inscrit un cercle, dans le cercle un carré, et on répète l'opération. Calculer la somme des aires des carrés.

2.2.15 Dans un carré de coté 1, on trace un carré dont les sommets sont situés au tiers des côtés du carré initial, et on répète l'opération. Calculer la longueur de la spirale dont les premiers segments sont tracés en gras ci-après :



2.2.16 On construit une spirale à l'aide d'une série de demi-cercles. Le diamètre du premier demi-cercle vaut 1 alors que le diamètre de chacun des demi-cercles suivants est égal aux trois quarts de celui du cercle précédent. Quelle est la longueur de la spirale et autour de quel point s'enroule-t-elle ?

2.2.17 On considère la suite $(u_n)_{n \in \mathbb{N}^*}$ définie par $u_n = 1 + \frac{1}{n}$.

- Combien y a-t-il de termes de cette suite dans l'intervalle $[1.001 ; 2]$?
- Déterminer la valeur du plus petit entier N tel que, pour tout entier $n \geq N$, on a $u_n \in]0.99 ; 1.01[$.

2.2.18 On considère la suite $(u_n)_{n \in \mathbb{N}^*}$ définie par $u_n = \frac{2n^2}{n+1}$

Calculer le plus petit entier naturel p tel que $(n > p \Rightarrow u_n > 1\,000)$

2.2.19 Déterminer un nombre entier naturel p tel que :

- $n > p \Rightarrow (1.1)^n > 10\,000$
- $n > p \Rightarrow (0.5)^n < 0.005$

2.3 Suites arithmétiques et géométriques

2.3.1 Une suite arithmétique finie a comme premier terme 6 et comme raison 7. La somme de ses termes est 41 094. Quel est le nombre de termes ?

2.3.2 Déterminer une suite arithmétique comportant cinq termes dont la somme est égale à 25 et la somme des carrés est égale à 165.

2.3.3 On considère une suite arithmétique (u_n) de raison r . Démontrer que la suite (v_n) définie par $v_n = -3u_n + 2$ est aussi une suite arithmétique ; quelle est sa raison ?

2.3.4 Soit (u_n) une suite arithmétique de raison r . On définit une nouvelle suite (v_n) par son terme général $v_n = u_{n+1}^2 - u_n^2$. Démontrer que la suite (v_n) est aussi une suite arithmétique ; quelle est sa raison ?

2.3.5 Déterminer une suite arithmétique qui comporte 18 termes, sachant que la somme de ses 17 premiers termes est égale à 663 et que la somme de ses 17 derniers termes est égale à 731.

2.3.6 Trouver la somme de tous les nombres entiers de 1 à 200 qui ne sont multiples ni de 4, ni de 9.

2.3.7 On forme les ensembles $A_1 = \{1\}$, $A_2 = \{2, 3\}$, $A_3 = \{4, 5, 6\}$, $A_4 = \{7, 8, 9, 10\}$, $A_5 = \{11, 12, 13, 14, 15\}$ et ainsi de suite. Calculer la somme des éléments de l'ensemble A_n .

2.3.8 On considère les 10 termes d'une suite géométrique finie. Sachant que le quatrième terme vaut $\frac{3}{2}$ et le neuvième 48, calculer la raison, le premier terme, le dernier terme et la somme des termes de cette suite.

2.3.9 Soit (u_n) une suite géométrique. On définit une nouvelle suite (v_n) par son terme général $v_n = u_{n+1} - u_n$. Démontrer que la suite (v_n) est aussi une suite géométrique.

2.3.10 Calculer la somme des carrés des n termes d'une suite géométrique finie.

2.3.11 a) La population P d'une ville augmente de 5% chaque année. Quelle sera sa population dans 20 ans, dans 50 ans ? Au bout de combien d'années, la population aura-t-elle triplé ?

b) La population mondiale a passé de 2.5 milliards (1950) à 6 milliards (2000). Quel a été le taux d'accroissement annuel ?

2.3.12 Donner une formule pour les sommes

a) $1 + 2 + 3 + \dots + n$

b) $1 + 2 + 4 + 8 + \dots + 2^n$

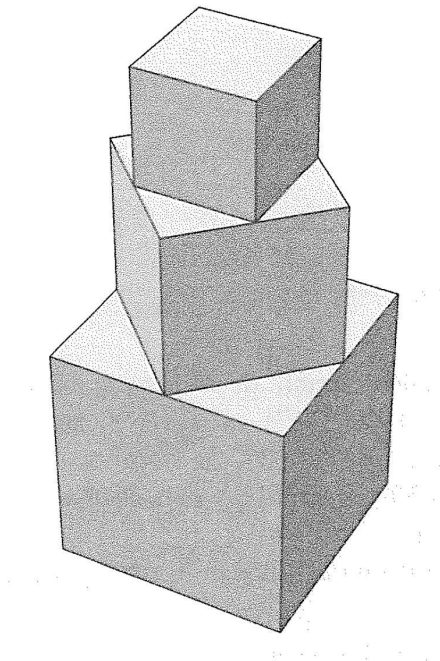
c) $\frac{1}{s} + \frac{1}{s^2} + \frac{1}{s^3} + \dots + \frac{1}{s^n}$

d) $1 - g^2 + g^4 - \dots + (-1)^n g^{2n}$

e) $\frac{1}{1-x} + \left(\frac{1}{1-x}\right)^2 + \left(\frac{1}{1-x}\right)^3 + \dots + \left(\frac{1}{1-x}\right)^n$

2.3.13 Sur un cube de 1 mètre de côté, on pose un deuxième cube de sorte que les sommets de la base du deuxième cube coïncident avec les milieux des côtés de la face supérieure du premier cube. De la même manière, on pose un troisième cube sur le deuxième

et ainsi de suite.



- Calculer la hauteur et le volume de cette pile composée de n cubes.
- Calculer la hauteur et le volume de cette pile de cubes lorsque $n \rightarrow \infty$.

2.3.14 Un nombre réel positif x est donné sous forme d'une *fraction continue* lorsqu'il se présente comme suit :

$$x = x_0 + \frac{1}{x_1 + \frac{1}{x_2 + \frac{1}{x_3 + \frac{1}{x_4 + \dots}}}} \quad \text{où } x_0, x_1, x_2, x_3, x_4, \dots \in \mathbb{N}$$

On le note alors comme une suite (finie ou infinie) de nombres entiers : $x = [x_0, x_1, x_2, x_3, x_4, \dots]$.

Exemple : $x = [2, 3, 4] = 2 + \frac{1}{3 + \frac{1}{4}} = \frac{30}{13}$

- Donner sous forme d'une fraction le nombre $y = [2, 1, 0, 3]$.
- Écrire sous forme de fraction continue le nombre $z = \frac{27}{19}$.
- Démontrer que tout nombre rationnel peut s'écrire sous forme de fraction continue avec un nombre fini de termes, et réciproquement.
- Donner les 10 premiers termes du développement de $\sqrt{2}$ en fraction continue. Démontrer ce que le résultat laisse supposer pour la fraction continue de ce nombre.
- La suite $[x_0, x_1, x_2, x_3, \dots]$ de la fraction continue d'un nombre réel x se calcule avec une calculatrice à l'aide de l'algorithme :
 - Lire x

- b) Écrire $E(x)$, la partie entière de x
- c) Tant que $x \neq E(x)$, remplacer x par $1/(x - E(x))$ et écrire $E(x)$

Déterminer les fractions continues des nombres e , π et $\Phi = \frac{1 + \sqrt{5}}{2}$ (le nombre d'or).

2.3.15 On considère la suite (u_n) définie sur $\mathbb{N}^*$ par

$$u_n = \left(1 + \frac{1}{n}\right)^n.$$

- a) Calculer u_1 , u_2 , u_3 et u_5 .
- b) Donner une valeur approchée de u_{10} et u_{100} .
- c) Que semble-t-on conjecturer sur l'évolution de la suite?
- d) Montrer que pour tout entier $n \geq 1$,

$$u_n = \frac{(n+1)^n}{n^n}.$$

- e) Comparer u_{n+1} et u_n .
- f) Montrer que la suite (u_n) est croissante.
- g) Montrer que la suite (u_n) est majorée.
- h) Montrer que la suite (u_n) est convergente.

2.4 Solutions des exercices

2.1.1 —

2.1.2 —

2.1.3 —

2.1.4 —

2.2.1 a) $\frac{1}{3}, \frac{1}{2}, \frac{3}{5}, \frac{2}{3}$; b) 0, 2, 0, 2; c) 1, 0, -3, 0; d) $1, \sqrt{10} - 2, 3\sqrt{2} - 3, 2\sqrt{7} - 4$.

2.2.2 a) 1, 3, 7, 15; b) $3, \frac{1}{3}, -\frac{5}{9}, -\frac{23}{27}$; c) $2, \frac{5}{4}, \frac{41}{40}, \frac{3'281}{3'280}$; d) 1, 1, 1, 1.

2.2.3 $a_n = 1 + 3n$; $a_n = 2n^2$; $a_n = (-1)^{n+1}\frac{1}{n}$; $a_n = \frac{n-2}{n+1}$; $a_n = (-3)^n$; $a_n = \sqrt[n]{n}$.

2.2.4 a) bornée par 1 et 2, strictement croissante; b) majorée par -1, strictement décroissante;

c) minorée par -6, strictement croissante; d) bornée par 0 et 3/19, strictement décroissante; e) bornée par 1 et 9; f) minorée par 2, strictement croissante.

2.2.5 a) $N = 33, N = \frac{3(1+\epsilon)}{\epsilon}$; b) $N = 3, N = \frac{1-\epsilon}{\epsilon}$.

2.2.6 a) 2; b) -2; c) 0; d) -; e) 16; f) 7.

2.2.7 a) 0; b) 0; c) 0; d) 0; e) 0; f) x .

2.2.8 —

2.2.9 a) -; b) 2.

2.2.10 a) -; b) 0.

2.2.11 a) - b) - c) 4.

2.2.12 a) ∞ ; b) -1; c) 2/3; d) 1/2; e) 2; f) 1/2.

2.2.13 a) suite géométrie de premier terme -1 et de raison 1/4; b) $a_n = -(1/4)^{n-1} + 4$, avec $n \geq 1$, convergeant vers 4.

2.2.14 2.

2.2.15 $\frac{1}{3 - \sqrt{5}}.$

2.2.16 Longueur de 2π et enroulement autour de $4/7$.

2.2.17 a) 1000 b) $N = 101$

2.2.18 $p = 500$

2.2.19 a) $p = 96$ b) $p = 7$

2.3.1 108

2.3.2 1, 3, 5, 7, 9 ou 9, 7, 5, 3, 1

2.3.3 Il suffit de démontrer que $v_{n+1} - v_n$ ne dépend pas de n ; la raison de (v_n) est $-3r$.

2.3.4 La différence $v_{n+1} - v_n$ ne dépend pas de n ; la raison de (v_n) est $2r^2$.

2.3.5 Le premier terme de la suite arithmétique cherchée est 7, sa raison est 4.

2.3.6 13263 (attention aux multiples de 36)

2.3.7 $s_n = \frac{n(n^2 + 1)}{2}$

2.3.8 $r = 2, u_1 = \frac{3}{16}, u_{10} = 96, s_{10} = \frac{3069}{16}$

2.3.9 $\frac{v_{n+1}}{v_n} = r$ où r est la raison de la suite géométrique (u_n)

2.3.10 Les carrés des termes d'une suite géométrique de premier terme u_1 et de raison r forment une nouvelle suite géométrique de première terme u_1^2 et de raison r^2 .

La somme cherchée est donc $u_1^2 \cdot \frac{1 - r^{2n}}{1 - r^2}$

2.3.11 a) Population dans 20 ans $\approx 2.7 P$, population dans 50 ans $\approx 11,5 P$

La population aura triplé au bout de 22.5 années.

b) Taux cherché : 1.766%

2.3.12

a) $\frac{n(n+1)}{2}$

c) $\frac{s^n - 1}{s^n(s-1)}$

e) $\frac{1 - (1-x)^n}{x(1-x)^n}$

b) $2^{n+1} - 1$

d) $\frac{1 + (-1)^n \cdot g^{2n+2}}{1 + g^2}$

2.3.13 Les hauteurs des cubes forment une suite géométrique de premier terme 1 (mètre) et de raison $r = \frac{\sqrt{2}}{2}$. La hauteur de la pile est

$$1 + \frac{\sqrt{2}}{2} + \frac{1}{2} + \frac{\sqrt{2}}{4} + \frac{1}{8} + \dots = \frac{1}{1 - \frac{\sqrt{2}}{2}} = \frac{2}{2 - \sqrt{2}} = 2 + \sqrt{2}$$

Les volumes des cubes forment une suite géométrique de premier terme 1 (m³) et de raison $r = \frac{\sqrt{2}}{4}$. Le volume de la pile est

$$1 + \frac{\sqrt{2}}{4} + \frac{1}{8} + \frac{\sqrt{2}}{32} + \frac{1}{64} + \dots = \frac{1}{1 - \frac{\sqrt{2}}{4}} = \frac{4}{4 - \sqrt{2}} = \frac{2(4 + \sqrt{2})}{7}$$

2.3.14

a) $y = \frac{9}{4}$

b) $z = [1, 2, 2, 1, 2]$

d) $\sqrt{2} = [1, 2, 2, 2, 2, 2, 2, \dots]$

e)

$$e = [2, 1, 2, 1, 1, 4, 1, 1, 6, 1, 1, 8, 1, 1, 10, \dots]$$

$$\pi = [3, 7, 15, 1, 292, 1, 1, 1, 2, 1, 3, \dots]$$

$$\Phi = [1, 1, 1, 1, 1, 1, 1, \dots]$$

2.3.15

a) Calcul des premiers termes :

$$u_1 = \left(1 + \frac{1}{1}\right)^1 = 2$$

$$u_2 = \left(1 + \frac{1}{2}\right)^2 = \frac{9}{4} = 2,25$$

$$u_3 = \left(1 + \frac{1}{3}\right)^3 = \frac{64}{27} \approx 2,370$$

$$u_5 = \left(1 + \frac{1}{5}\right)^5 = \left(\frac{6}{5}\right)^5 = \frac{7776}{3125} \approx 2,488$$

b) Valeurs approchées :

$$u_{10} = \left(1 + \frac{1}{10}\right)^{10} \approx 2,594$$

$$u_{100} = \left(1 + \frac{1}{100}\right)^{100} \approx 2,705$$

c) On constate que la suite semble croissante et semble se rapprocher d'une limite voisine de 2,718.

d) —

e) —

f) —

Chapitre 3

Cryptographie asymétrique

3.1 Plus grand diviseur commun et plus petit multiple commun

3.1.1 Soit a et b deux entiers positifs. On note $\gcd(a, b)$, de l'anglais *greatest common divisor*, le plus grand diviseur commun de a et b . On peut aussi le noter $\text{pgcd}(a, b)$.

Montrer que $\gcd(a, b)$ existe.

On dira de plus que a et b sont *premiers entre eux* si $\gcd(a, b) = 1$

3.1.2 Trouver tous les diviseurs communs positifs de

- a) 16 et 48,
- b) 30 et 45,
- c) 18 et 65.

3.1.3 Trouver le plus grand diviseur commun de

- a) 35 et 65,
- b) 135 et 156,
- c) 49 et 99.

3.1.4 Trouver le plus grand diviseur commun de 17017 et 19210.

3.1.5 Trouver le plus grand diviseur commun de 21331 et de 43947. (L'utilisation d'une calculatrice peut être judicieuse.)

3.1.6 Trouver le plus grand diviseur commun de 210632 et de 423137. (L'utilisation d'un ordinateur peut être judicieuse.)

3.1.7 Montrer que pour tout $n \in \mathbb{N}$, n et $n + 1$ sont premiers entre eux.

3.1.8 Montrer que si $a \mid b$, alors $\gcd(a, b) = a$.

3.1.9 Soit a et b deux nombres entiers. Supposons qu'il existe r et s deux nombres, également entiers, tels que $a \cdot r + b \cdot s = 1$. Montrer que a et b sont premiers entre eux.

3.2 Algorithme d'Euclide

3.2.1 Appliquer l'algorithme d'Euclide (soustractions successives) à

- a) 135 et 156 ;
- b) 17017 et 19210 ;
- c) 21331 et 43947.

3.2.2 Appliquer l'algorithme d'Euclide (divisions avec reste) à

- a) 121 et 365 ;
- b) 89 et 144 ;
- c) 295 et 595 ;
- d) 1001 et 1309.

3.2.3 En utilisant l'algorithme d'Euclide (divisions avec reste), trouver le plus grand diviseur commun de

- a) 17017 et 18900 ;
- b) 21063 et 43137 ;
- c) 92263 et 159037 ;
- d) 112345 et 112354.

3.2.4 Montrer que si e est un diviseur de a et que e est un diviseur de b , alors e est un diviseur de $ar + bs$ pour des entiers r et s quelconques.

3.2.5 Calculer $\text{pgcd}(987, 610)$ à l'aide de l'énumération des diviseurs, puis à l'aide de l'algorithme d'Euclide.

3.2.6 Calculer à l'aide de l'algorithme d'Euclide le pgcd des nombres a et b suivants :

- a) $a = 1233, b = 9999$;
- b) $a = 12345, b = 54321$;

3.2.7 Calculer les nombres s et t tels que $s \cdot a + t \cdot b = \text{pgcd}(a, b)$, avec les nombres a et b suivants :

- a) $a = 72, b = 39$;
- b) $a = 1008, b = 25$;
- c) $a = 1993, b = 210$;
- d) $a = 1995, b = 323$.

3.2.8 Montrer que $\text{pgcd}(n, 0) = n$, quel que soit l'entier naturel n .

3.2.9 Écrire un programme qui calcule le pgcd de deux nombres entiers a et b à l'aide de l'algorithme d'Euclide.

3.3 Algorithme d'Euclide étendu

3.3.1 En utilisant l'algorithme d'Euclide étendu, trouver d , le plus grand diviseur commun et trouver également r, s tels que $a r + b s = d$ pour a et b donnés ci-dessous :

- a) 270 et 114 ;
- b) 242 et 1870 ;
- c) 600 et 11312 ;
- d) 11213 et 1001 ;
- e) 500 et 3000.

3.3.2 Écrire un programme qui permet de calculer à l'aide de l'algorithme d'Euclide étendu les nombres s et t tels que $a \cdot s + b \cdot t = \text{pgcd}(a, b)$, pour a et b deux entiers positifs quelconques.

Utiliser ce programme pour calculer s, t et $\text{pgcd}(a, b)$ si a et b valent :

- a) 3405 et 3367 ;
- b) 13594140393 et 542988845 ;
- c) 17192503173298665204451 et 3719762684395467374891 ;
- d) 83575760507228709062535623 et 125120843342028658261747445 ;
- e) 1652687235802237428240170266331616396548041575642963 et 1134864281085530578683431668611349549126999384309721.

3.3.3 Montrer que, pour tout nombre naturel n , les nombres n et $n^2 + 1$ sont premiers entre eux.

3.4 Exponentiation modulaire

3.4.1 Si n est un nombre naturel, la n -ième puissance d'un nombre a est, par définition, le produit de n facteurs égaux à a . Ainsi, d'après cette définition, le calcul de a^n nécessite $n - 1$ multiplications. On peut cependant obtenir le même résultat en effectuant moins d'opérations.

Voici à titre d'exemple l'évaluation de a^{35} .

- On écrit l'exposant n comme une somme de puissance de 2. Ici, $35 = 32 + 2 + 1$;
- on calcule ensuite les puissances paires de a : $a^2 = a \cdot a$, $a^4 = a^2 \cdot a^2$, $a^8 = a^4 \cdot a^4$, $a^{16} = a^8 \cdot a^8$, $a^{32} = a^{16} \cdot a^{16}$.

- on multiplie pour terminer les « bons » carrés : $a^{35} = a^{32} \cdot a^2 \cdot a^1$;

Le nombre de multiplications nécessaires est dans ce cas de 7, au lieu de 34.

- a) Combien de multiplications nécessite cet algorithme pour calculer chacune des puissances suivantes : a^{10} , a^{61} , a^{1000} ?
- b) Calculer $835^{25} \pmod{1073}$, en 6 multiplications.

3.4.2 Dans le shell de python, calculer

$$17^{13} \pmod{23}$$

en suivant la procédure ci-dessous :

- a) Stocker la valeur de l'exposant dans une variable : `n = 13`.
- b) Déclarer deux variables `r` et `x`, dont la valeur initiale est 1 et 17, respectivement.
- c) À l'aide de la fonction prédéfinie `bin`, obtenir le code binaire de l'exposant : `bin(n)`.
- d) En utilisant judicieusement et le nombre de fois que c'est nécessaire les instructions ci-dessous, faire en sorte que la variable `r` contienne le résultat cherché.

```
r = r*x % 23
x = x*x % 23
```

3.4.3 En utilisant une seule boucle `while`, des divisions entières par 2 et des calculs de restes modulo 2, écrire en python le code qui fait afficher à la console le code binaire d'un nombre entier, sur une colonne.

3.4.4 Effectuer les calculs ci-dessous :

- | | |
|--------------------------|--------------------------|
| a) $25^5 \pmod{133}$ | e) $234^{65} \pmod{667}$ |
| b) $100^{65} \pmod{133}$ | f) $447^{65} \pmod{667}$ |
| c) $107^5 \pmod{133}$ | g) $99^{417} \pmod{667}$ |
| d) $46^{65} \pmod{133}$ | h) $664^{65} \pmod{667}$ |

3.4.5 Écrire un programme en python qui permet de faire calculer

$$a^b \bmod n$$

pour a , b et n des nombres entiers positifs qui sont formés de centaines de chiffres.

3.4.6 Utiliser le programme de l'exercice précédent pour faire calculer $a^b \bmod n$, si

$a = 120394871029348710928374019238472034923495602345762938476565823$

$b = 202932093845709283475092834750928347509283475092384750923847507$

$n = 323490850293452304958775209384776108273649876450247856029384751$

3.5 Théorèmes de Fermat et d'Euler

Soit $m \geq 2$ et a un entier tel que $\text{pgcd}(m, a) = 1$. L'ordre de a modulo m est le plus petit entier positif e tel que $a^e \equiv 1 \pmod{m}$.

3.5.1 Déterminer l'ordre de chaque élément non nul de $\mathbb{Z}_5$.

3.5.2 Déterminer l'ordre de chaque inversible de $\mathbb{Z}_9$.

3.5.3 Trouver l'ordre de 2 dans $\mathbb{Z}_m$ si

- a) $m = 11$;
- b) $m = 17$;
- c) $m = 31$;
- d) $m = 9$;
- e) $m = 14$.

3.5.4 Ecrire un programme en python qui permet de trouver l'ordre de a dans $\mathbb{Z}_m$, sachant que $\text{pgcd}(m, a) = 1$.

3.5.5 À l'aide du programme écrit à l'exercice précédent, trouver l'ordre de tous les éléments non nuls de $\mathbb{Z}_m$ si

- a) $m = 11$;
- b) $m = 13$;
- c) $m = 17$.

3.5.6 À l'aide du même programme, trouver l'ordre de tous les éléments inversibles de $\mathbb{Z}_{24}$

3.5.7 Effectuer les calculs ci-dessous et réduire le résultat modulo 7.

- a) 5^6 b) 2^6 c) 4^6 d) 6^6

3.5.8 Effectuer les calculs ci-dessous et réduire le résultat modulo 17.

- a) 8^{16} b) 2^{16} c) 11^{16} d) 5^{16}

3.5.9 Effectuer les calculs ci-dessous :

- a) $3^5 \pmod{5}$ d) $5^7 \pmod{7}$ g) $11^{13} \pmod{13}$ j) $8^{19} \pmod{19}$
- b) $2^5 \pmod{5}$ e) $6^7 \pmod{7}$ h) $10^{19} \pmod{19}$ k) $12^{17} \pmod{17}$
- c) $3^7 \pmod{7}$ f) $5^{13} \pmod{13}$ i) $2^{19} \pmod{19}$ l) $10^{17} \pmod{17}$

Que peut-on en déduire ?

3.5.10 À l'aide de calculs élémentaires et sans utiliser directement le théorème de Fermat, expliquer pourquoi si a est inversible dans $\mathbb{Z}_{15}$, alors

$$a^8 \equiv 1 \pmod{15}$$

3.5.11 De même, expliquer pourquoi si a est inversible dans $\mathbb{Z}_{21}$, alors

$$a^{12} \equiv 1 \pmod{21}$$

3.5.12 À l'aide du théorème de Fermat, calculer $2^9 \pmod{11}$ et vérifier que 2^9 est l'inverse de 2 modulo 11.

3.5.13 On note $\varphi(n)$ le nombre d'éléments inversibles de $\mathbb{Z}_n$. Calculer $\varphi(n)$ pour les valeurs de n données ci-dessous :

a) $n = 4, n = 8, n = 16, n = 32$;

b) $n = 3, n = 9, n = 27, n = 81$;

c) $n = 7^5$;

d) $n = 4027$;

e) $n = 4087$;

3.5.14 Écrire, en python, un programme qui fait calculer $\varphi(m)$, pour m un entier positif donné.

3.5.15 Vérifier le théorème d'Euler pour 1, 3, 7 et 9 dans $\mathbb{Z}_{10}$.

3.5.16 Calculer $a^5 \pmod{7}$ pour $a = 1, 2, 3, 4, 5, 6$ et vérifier que

$$a^5 \cdot a \equiv 1 \pmod{7}$$

3.5.17 Vérifier l'équivalence ci-dessous

$$2^{\varphi(21)} \equiv 1 \pmod{21}$$

3.5.18 Calculer $48^{322} \pmod{25}$.

3.5.19 Calculer $40^{322} \pmod{21}$.

3.6 Factoriser un nombre entier

3.6.1 Un nombre supérieur à 1 est dit *premier* s'il a exactement deux diviseurs : 1 et lui-même. Les nombres qui ne sont pas premiers sont dits *composés*.

Soit $n \in \mathbb{N}$ un nombre composé. Montrer que n admet un facteur inférieur ou égal à $\sqrt{n}$.

3.6.2 A l'aide du crible d'Eratosthène, déterminer « à la main » les nombres premiers inférieurs à 500.

2	3	5	7	9	11	13	15	17	19
21	23	25	27	29	31	33	35	37	39
41	43	45	47	49	51	53	55	57	59
61	63	65	67	69	71	73	75	77	79
81	83	85	87	89	91	93	95	97	99
101	103	105	107	109	111	113	115	117	119
121	123	125	127	129	131	133	135	137	139
141	143	145	147	149	151	153	155	157	159
161	163	165	167	169	171	173	175	177	179
181	183	185	187	189	191	193	195	197	199
201	203	205	207	209	211	213	215	217	219
221	223	225	227	229	231	233	235	237	239
241	243	245	247	249	251	253	255	257	259
261	263	265	267	269	271	273	275	277	279
281	283	285	287	289	291	293	295	297	299
301	303	305	307	309	311	313	315	317	319
321	323	325	327	329	331	333	335	337	339
341	343	345	347	349	351	353	355	357	359
361	363	365	367	369	371	373	375	377	379
381	383	385	387	389	391	393	395	397	399
401	403	405	407	409	411	413	415	417	419
421	423	425	427	429	431	433	435	437	439
441	443	445	447	449	451	453	455	457	459
461	463	465	467	469	471	473	475	477	479
481	483	485	487	489	491	493	495	497	499

3.6.3 Ecrire un programme en python qui renvoie la liste de tous les nombres premiers inférieurs à 10^6 . Utiliser le crible d'Eratosthène.

3.6.4 Trouver un facteur de 4841.

3.6.5 Montrer que 521 est un nombre premier.

3.6.6 Factoriser les nombres ci-dessous.

- | | | | |
|--------|--------|---------|---------|
| a) 200 | e) 35 | i) 713 | m) 4897 |
| b) 150 | f) 77 | j) 1147 | n) 6319 |
| c) 128 | g) 143 | k) 473 | o) 1591 |
| d) 164 | h) 323 | l) 493 | p) 901 |

3.7 RSA

3.7.1 Chiffrer et déchiffrer le message $m = 8$ à l'aide du système RSA, avec $p = 7$, $q = 11$ et $e = 17$.

3.7.2 Connaissant la clef publique (n, e) d'un individu, déterminer la clé privée d dans les cas suivants :

- | | |
|--------------------------|---------------------------|
| a) $n = 1073, e = 117$; | d) $n = 117, e = 17$; |
| b) $n = 1073, e = 115$; | e) $n = 105, e = 13$; |
| c) $n = 111, e = 31$; | f) $n = 10001, e = 187$. |

3.7.3 Un ennemi intercepte le message chiffré $c = 10$, dont le destinataire possède la clef publique $e = 5$, $n = 35$. Quel est le texte clair m ?

3.7.4 Soit $p = 23$ et $q = 31$. On donne encore $e = 17$.

- Calculer d , l'inverse de e modulo $(p - 1) \cdot (q - 1) = 22 \cdot 30 = 660$.
- Ecrire la clef privée correspondante.
- Ecrire la clef publique correspondante.
- Chiffrer le « message » $m = 333$ en utilisant la formule

$$c = m^e \pmod{n}$$

si $n = p \cdot q$.

- Retrouver le message à partir du chiffre c en utilisant la formule

$$m = c^d \pmod{n}$$

- Chiffrer $m = 555$.
- Déchiffrer $c = 100$.

3.7.5 Soit $p = 13$ et $q = 19$. On donne encore $e = 11$.

- Calculer d , l'inverse de e modulo $(p - 1) \cdot (q - 1)$.
- Ecrire la clef privée correspondante.

- c) Ecrire la clef publique correspondante.
- d) Chiffrer le « message » $m = 123$ en utilisant la formule

$$c = m^e \pmod n$$

si $n = p \cdot q$.

- e) Retrouver le message à partir du chiffre c en utilisant la formule

$$m = c^d \pmod n$$

- f) Chiffrer $m = 222$.
- g) Déchiffrer $c = 55$.

3.7.6 Soit $p = 7$ et $q = 23$. On donne encore $e = 5$.

- a) Calculer d , l'inverse de e modulo $(p - 1) \cdot (q - 1)$.
- b) Ecrire la clef privée correspondante.
- c) Ecrire la clef publique correspondante.
- d) Chiffrer le « message » $m = 111$ en utilisant la formule

$$c = m^e \pmod n$$

si $n = p \cdot q$.

- e) Retrouver le message à partir du chiffre c en utilisant la formule

$$m = c^d \pmod n$$

- f) Chiffrer $m = 22$. Commenter le résultat.
- g) Déchiffrer $c = 100$.

3.7.7 Un professeur envoie ses notes au secrétariat de l'école par mail. La clef publique du professeur est $(3; 55)$ et celle du secrétariat est $(3; 33)$.

- a) Vérifier que la clef du professeur (supposée connue de lui seul) est 27 et que celle du secrétariat est 7.
- b) Pour assurer la confidentialité de ses messages, le professeur chiffre les notes avec la clef RSA du secrétariat. Quel est le message chiffré qui correspond à la note 12 ?
- c) Pour assurer l'authenticité des messages contenant les notes, le professeur signe ses messages pour le secrétariat après les avoir chiffrés. Le secrétariat reçoit le message 23. Quelle est la note correspondante ?

3.7.8 Définir les termes cryptographie, cryptanalyse et cryptologie.

3.7.9 Bob utilise le protocole RSA et publie sa clef publique $N = 187$ et $e = 3$.

- a) Encoder le message $m = 15$ avec la clef publique de Bob.

- b) En utilisant le fait que $\varphi(N) = 160$, retrouver la factorisation de N , puis la clef privée de Bob.

3.7.10 Bob₁ et Bob₂ ont pour clef publique RSA respectivement (N, e_1) et (N, e_2) avec e_1 et e_2 premiers entre eux.

Alice envoie le même message m crypté avec les clefs publiques RSA de Bob₁ et Bob₂ en c_1 et c_2 .

Expliquer comment Eve, qui intercepte les deux messages cryptés et qui connaît les clefs publiques de Bob₁ et Bob₂, peut retrouver le message clair m .

3.7.11 Supposons que l'entier n soit le produit de deux nombres premiers p et q proches (on peut toujours supposer que $p > q$).

On pose $t = \frac{p+q}{2}$ et $s = \frac{p-q}{2}$.

Montrer que :

- l'entier s est petit ;
- $n = t^2 - s^2$;
- t est légèrement supérieur à la racine carrée de n ;
- à l'aide de ces informations, écrire un programme pour factoriser n ;
- appliquer cet algorithme pour factoriser 899, 110417 et 364957402 ;
- Trouver la clef secrète d correspondante à $(n = 51983, e = 17)$.

3.7.12 Le fichier `nombres_p_q_1.txt` contient deux nombres premiers p et q .

On choisit $e = 65537$ qui satisfait la condition $\gcd(e, \varphi(n)) = 1$, où $n = p \cdot q$. On dispose ainsi d'une clé publique (n, e) .

- Dans un fichier `clef.py`, construire la clé privée (n, d) . Afficher à la console la valeur de d .
- Coder le message suivant (présent dans `message_1.txt`) à l'aide de la clé publique (n, e) et l'afficher à la console :

12

- Décoder le message chiffré suivant (présent dans `chiffre_1.txt`) à l'aide de la clé privée (n, d) et l'afficher à la console.

25567439145662233958692485794698610817081042607044
 76375740884794753357993630922231929342522549252625
 83666181770268506210137463778407056604376942495443
 54344442167885488175571510178529674397470719290002
 50564288716583429743123017673776013168016434939838
 37662442086433370446824338338022627966782937829223
 36951763006508399321713627145581271533380725975341
 78087248990732332457837418659413481406361983817252
 46437822132392448183737304695659768807011137853820

23696953041028388595531536455164593424945746561789
34884366199670482448616653509555427947511254862696
41398200342295055677990818953963680224811410674869
657356236000829

3.8 Solutions des exercices

3.1.1 L'ensemble D des diviseurs communs de a et b n'est pas vide, car $1 \mid a$ et $1 \mid b$. Vu que l'ensemble des diviseurs de a est fini, D est également fini. Vu que D est fini, cet ensemble admet un maximum, noté d . Le nombre d n'est autre que $\gcd(a, b)$.

3.1.2

a) $\{1, 2, 4, 8, 16\}$,

b) $\{1, 3, 5, 15\}$,

c) $\{1\}$.

3.1.3

a) 5

b) 3

c) 1

3.1.4 17

3.1.5 $21331 = 83 \cdot 257$ et $43947 = 3 \cdot 3 \cdot 19 \cdot 257$.

On peut donc écrire $\gcd(21331, 43947) = 257$.

3.1.6 $210632 = 2 \cdot 2 \cdot 2 \cdot 113 \cdot 233$ et $423137 = 11 \cdot 11 \cdot 13 \cdot 269$.

On peut donc écrire $\gcd(21331, 43947) = 1$.

3.1.7 Soit $m \in \mathbb{N}$ tel que $m \mid n$ et $m \mid (n+1)$. Le nombre m doit alors diviser la différence $(n+1) - n$. Ce qui fait que $m \mid 1$. Cela implique finalement que $m = 1$. On a donc $\gcd(n, n+1) = 1$.

3.1.8 Supposons que $a \mid b$. Dans ce cas, a est un diviseur commun de a et b . C'est forcément le plus grand, car si $d \mid a$, alors $d \leq a$.

3.1.9 Notons $d = \gcd(a, b)$. On sait que $d \mid a$ et $d \mid b$, ce qui fait que $d \mid a \cdot r$ et $d \mid b \cdot r$. Par conséquent, $d \mid (a \cdot r + b \cdot s)$. Finalement, $d \mid 1$ et donc $d = 1$.

3.2.1 –

3.2.2 –

3.2.3 –

3.2.4 –

3.2.5 —

3.2.6 —

3.2.7 —

3.2.8 —

3.2.9 —

3.3.1 —

3.3.2 —

3.3.3 —

3.4.1 —

3.4.2 –

3.4.3

```
def beo_bin(n):  
    while n:  
        print(n%2)  
        n = n//2  
beo_bin(127)
```

3.4.4

a) $25^5 \bmod 133$

$$25^1 \bmod 133 = 25$$

$$25^2 \bmod 133 = 93$$

$$25^4 \bmod 133 = 4$$

$$25 * 4 \bmod 133 = 100$$

b) $100^{65} \bmod 133$

$$100^1 \bmod 133 = 100$$

$$100^2 \bmod 133 = 25$$

$$100^4 \bmod 133 = 93$$

$$100^8 \bmod 133 = 4$$

$$100^{16} \bmod 133 = 16$$

$$100^{32} \bmod 133 = 123$$

$$100^{64} \bmod 133 = 100$$

$$100 * 100 \bmod 133 = 25$$

c) $107^5 \bmod 133$

$$107^1 \bmod 133 = 107$$

$$107^2 \bmod 133 = 11$$

$$107^4 \bmod 133 = 121$$

$$107 * 121 \bmod 133 = 46$$

d) $46^{65} \bmod 133$

$$46^1 \bmod 133 = 46$$

$$46^2 \bmod 133 = 121$$

$$46^4 \bmod 133 = 11$$

$$46^8 \bmod 133 = 121$$

$$46^{16} \bmod 133 = 11$$

$$46^{32} \bmod 133 = 121$$

$$46^{64} \bmod 133 = 11$$

$$46 * 11 \bmod 133 = 107$$

e) $234^{65} \bmod 667$

$$234^1 \bmod 667 = 234$$

$$234^2 \bmod 667 = 62$$

$$234^4 \bmod 667 = 509$$

$$234^8 \bmod 667 = 285$$

$$234^{16} \bmod 667 = 518$$

$$234^{32} \bmod 667 = 190$$

$$234^{64} \bmod 667 = 82$$

$$234 * 82 \bmod 667 = 512$$

f) $447^{65} \bmod 667$

$$447^1 \bmod 667 = 447$$

$$447^2 \bmod 667 = 376$$

$$447^4 \bmod 667 = 639$$

$$447^8 \bmod 667 = 117$$

$$447^{16} \bmod 667 = 349$$

$$447^{32} \bmod 667 = 407$$

$$447^{64} \bmod 667 = 233$$

$$447 * 233 \bmod 667 = 99$$

g) $99^{417} \bmod 667$

$$99^1 \bmod 667 = 99$$

$$99^2 \bmod 667 = 463$$

$$99^4 \bmod 667 = 262$$

$$99^8 \bmod 667 = 610$$

$$99^{16} \bmod 667 = 581$$

$$99^{32} \bmod 667 = 59$$

$$99^{64} \bmod 667 = 146$$

$$99^{128} \bmod 667 = 639$$

$$99^{256} \bmod 667 = 117$$

$$99 * 59 * 639 * 117 \bmod 667 = 447$$

h) $664^{65} \bmod 667$

$$664^1 \bmod 667 = 664$$

$$664^2 \bmod 667 = 9$$

$$664^4 \bmod 667 = 81$$

$$664^8 \bmod 667 = 558$$

$$664^{16} \bmod 667 = 542$$

$$664^{32} \bmod 667 = 284$$

$$664^{64} \bmod 667 = 616$$

$$664 * 616 \bmod 667 = 153$$

3.4.5 —

3.4.6 —

3.5.1 —

3.5.2 —

3.5.3 —

3.5.4 —

3.5.5 —

3.5.6 —

3.5.7 Pour tout $1 \leq a \leq 6$, on a $a^6 \equiv 1 \pmod{7}$.

3.5.8 Pour tout $1 \leq a \leq 16$, on a $a^{16} \equiv 1 \pmod{17}$.

3.5.9 On peut émettre l'hypothèse que si p est premier, alors $a^p \equiv a \pmod{p}$. C'est un théorème, appelé le « petit théorème de Fermat ».

3.5.10 Soit $(1, 2, 4, 7, 8, 11, 13, 14)$ la liste des nombres inversibles de $\mathbb{Z}_{15}$ et a un nombre de cette liste. Il a 8 nombres inversibles. On construit une nouvelle liste comme suit :

$$\ell = (a \cdot 1, a \cdot 2, a \cdot 4, a \cdot 7, a \cdot 8, a \cdot 11, a \cdot 13, a \cdot 14)$$

Voyons pourquoi cette liste ne compte que des éléments distincts : Soit $a \cdot z_1$ et $a \cdot z_2$ deux éléments de ℓ et supposons que

$$a \cdot z_1 \pmod{15} = a \cdot z_2 \pmod{15}$$

Notons encore a' l'inverse de a modulo 15. On a alors

$$a' \cdot a \cdot z_1 \pmod{15} = a' \cdot a \cdot z_2 \pmod{15}$$

ce qui veut dire que

$$z_1 \pmod{15} = z_2 \pmod{15}$$

vu que $a' \cdot a \pmod{15} = 1$. Cela signifie que si $z_1 \neq z_2$, alors $a \cdot z_1 \neq a \cdot z_2$, modulo 15 bien entendu.

La liste ℓ compte donc 8 éléments, tous inversibles, vu que l'inverse de $a \cdot z$ est donné par $z' \cdot a'$ si z' désigne l'inverse de z modulo 15. On peut donc dire que ℓ est la liste des inversibles de $\mathbb{Z}_{15}$, éventuellement placés dans un ordre différent.

Cela signifie que

$$1 \cdot 2 \cdot 4 \cdot 7 \cdot 8 \cdot 11 \cdot 13 \cdot 14 \equiv a \cdot 1 \cdot a \cdot 2 \cdot a \cdot 4 \cdot a \cdot 7 \cdot a \cdot 8 \cdot a \cdot 11 \cdot a \cdot 13 \cdot a \cdot 14 \pmod{15}$$

et donc que

$$1 \cdot 2 \cdot 4 \cdot 7 \cdot 8 \cdot 11 \cdot 13 \cdot 14 \equiv 1 \cdot 2 \cdot 4 \cdot 7 \cdot 8 \cdot 11 \cdot 13 \cdot 14 \cdot a \cdot a \cdot a \cdot a \cdot a \cdot a \cdot a \cdot a \pmod{15}$$

Après simplification, on obtient $1 \equiv a^8 \pmod{15}$. La simplification est possible vu que tous les nombres impliqués dans l'écriture ci-dessus sont des inversibles.

3.5.11 Il y a 12 inversibles dans $\mathbb{Z}_{21}$. Un argument analogue à celui donné à l'exercice [3.5.10](#) donne la congruence $a^{12} \equiv 1 \pmod{21}$ si a est inversible.

3.5.12 —

3.5.13 —

3.5.14 —

3.5.15 —

3.5.16 —

3.5.17 —

3.5.18 —

3.5.19 —

3.6.1 Soit n un nombre entier composé. Cela signifie que $n = a \cdot b$ avec a et b des entiers différents de 1. Supposons que $a > \sqrt{n}$ et que $b > \sqrt{n}$. On peut, dans ce cas, écrire que $a \cdot b > \sqrt{n} \cdot \sqrt{n} = n$, ce qui est contradictoire. L'un des deux facteurs est donc plus petit ou égal à $\sqrt{n}$.

3.6.2

2	3	5	7	11	13	17	19	23	29
31	37	41	43	47	53	59	61	67	71
73	79	83	89	97	101	103	107	109	113
127	131	137	139	149	151	157	163	167	173
179	181	191	193	197	199	211	223	227	229
233	239	241	251	257	263	269	271	277	281
283	293	307	311	313	317	331	337	347	349
353	359	367	373	379	383	389	397	401	409
419	421	431	433	439	443	449	457	461	463
467	479	487	491	499					

3.6.3

```
def listePremiers(borne):
    liste_premiers = list(range(3, borne, 2))
    liste_premiers = [2] + liste_premiers
    r = int(borne**(1/2))

    n = 3
    j = 1
```

```

t = len(liste_premiers)

while n <= r:
    for i in range(j + n, t, n):
        if liste_premiers[i]:
            liste_premiers[i] = 0
    j += 1
    while not liste_premiers[j]:
        j += 1
    n = liste_premiers[j]

premiers = []
for p in liste_premiers:
    if p:
        premiers.append(p)
return premiers

```

3.6.4 $4841 = 103 \cdot 47$

3.6.5 Vu que $\sqrt{521} \simeq 22.83$, si 521 n'admet pas de facteur premier inférieur à 22, c'est un nombre premier. Aucun des nombres de la liste

$$(2, 3, 5, 7, 11, 13, 17, 19)$$

n'étant un diviseur de 521, c'est un nombre premier.

3.6.6 On donne ci-dessous la liste des facteurs premiers de chaque nombre :

- | | | | |
|--------------------------|-------------|-------------|-------------|
| a) (2, 2, 2, 5, 5) | e) (5, 7) | i) (23, 31) | m) (59, 83) |
| b) (2, 3, 5, 5) | f) (7, 11) | j) (31, 37) | n) (71, 89) |
| c) (2, 2, 2, 2, 2, 2, 2) | g) (11, 13) | k) (11, 43) | o) (37, 43) |
| d) (2, 2, 41) | h) (17, 19) | l) (17, 29) | p) (17, 53) |

3.7.1 —**3.7.2** —**3.7.3** $m = 5$ **3.7.4** Soit $p = 23$ et $q = 31$. On donne encore $e = 17$.

a) Calculer d , l'inverse de e modulo $(p - 1) \cdot (q - 1) = 22 \cdot 30 = 660$.

$$660 = 1 * 660 + (0) * 17$$

$$17 = 0 * 660 + (1) * 17$$

$$14 = 1 * 660 + (-38) * 17$$

$$3 = -1 * 660 + (39) * 17$$

$$2 = 5 * 660 + (-194) * 17$$

$$1 = -6 * 660 + (233) * 17$$

b) Ecrire la clef privée correspondante :

$$(23; 31; 233)$$

c) Ecrire la clef publique correspondante :

$$(713; 17)$$

d) Chiffrer le « message » $m = 333$ en utilisant la formule

$$c = m^e \mod n$$

$$\text{si } n = p \cdot q.$$

$$333^1 = 333$$

$$333^2 = 374$$

$$333^4 = 128$$

$$333^8 = 698$$

$$333^{16} = 225$$

$$333 * 225 = 60$$

e) Retrouver le message à partir du chiffre c en utilisant la formule

$$m = c^d \mod n$$

$$60^1 = 60$$

$$60^2 = 35$$

$$60^4 = 512$$

$$60^8 = 473$$

$$60^{16} = 560$$

$$60^{32} = 593$$

$$60^{64} = 140$$

$$60^{128} = 349$$

$$60 * 473 * 593 * 140 * 349 = 333$$

f) Chiffrer $m = 555$.

$$555^1 = 555$$

$$555^2 = 9$$

$$555^4 = 81$$

$$555^8 = 144$$

$$555^{16} = 59$$

$$555 * 59 = 660$$

g) Déchiffrer $c = 100$.

$$100^1 = 100$$

$$100^2 = 18$$

$$100^4 = 324$$

$$100^8 = 165$$

$$100^{16} = 131$$

$$100^{32} = 49$$

$$100^{64} = 262$$

$$100^{128} = 196$$

$$100 * 165 * 49 * 262 * 196 = 41$$

3.7.5 Soit $p = 13$ et $q = 19$. On donne encore $e = 11$.

a) Calculer d , l'inverse de e modulo $(p - 1) \cdot (q - 1) = 12 \cdot 18 = 216$.

$$216 = 1 * 216 + (0) * 11$$

$$11 = 0 * 216 + (1) * 11$$

$$7 = 1 * 216 + (-19) * 11$$

$$4 = -1 * 216 + (20) * 11$$

$$3 = 2 * 216 + (-39) * 11$$

$$1 = -3 * 216 + (59) * 11$$

b) Ecrire la clef privée correspondante :

$$(13; 19; 59)$$

c) Ecrire la clef publique correspondante :

$$(247; 11)$$

d) Chiffrer le « message » $m = 123$ en utilisant la formule

$$c = m^e \mod n$$

$$\text{si } n = p \cdot q.$$

$$123^1 = 123$$

$$123^2 = 62$$

$$123^4 = 139$$

$$123^8 = 55$$

$$123 * 62 * 55 = 24$$

e) Retrouver le message à partir du chiffre c en utilisant la formule

$$m = c^d \mod n$$

$$24^1 = 24$$

$$24^2 = 82$$

$$24^4 = 55$$

$$24^8 = 61$$

$$24^{16} = 16$$

$$24^{32} = 9$$

$$24 * 82 * 61 * 16 * 9 = 123$$

f) Chiffrer $m = 222$.

$$222^1 = 222$$

$$222^2 = 131$$

$$222^4 = 118$$

$$222^8 = 92$$

$$222 * 131 * 92 = 40$$

g) Déchiffrer $c = 55$.

$$55^1 = 55$$

$$55^2 = 61$$

$$55^4 = 16$$

$$55^8 = 9$$

$$55^{16} = 81$$

$$55^{32} = 139$$

$$55 * 61 * 9 * 81 * 139 = 139$$

3.7.6 Soit $p = 7$ et $q = 23$. On donne encore $e = 5$.

a) Calculer d , l'inverse de e modulo $(p - 1) \cdot (q - 1) = 6 \cdot 22 = 132$.

$$132 = 1 * 132 + (0) * 5$$

$$5 = 0 * 132 + (1) * 5$$

$$2 = 1 * 132 + (-26) * 5$$

$$1 = -2 * 132 + (53) * 5$$

b) Ecrire la clef privée correspondante :

$$(7; 23; 53)$$

c) Ecrire la clef publique correspondante :

$$(161; 5)$$

d) Chiffrer le « message » $m = 111$ en utilisant la formule

$$c = m^e \mod n$$

$$\text{si } n = p \cdot q.$$

$$111^1 = 111$$

$$111^2 = 85$$

$$111^4 = 141$$

$$111 * 141 = 34$$

e) Retrouver le message à partir du chiffre c en utilisant la formule

$$m = c^d \mod n$$

$$34^1 = 34$$

$$34^2 = 29$$

$$34^4 = 36$$

$$34^8 = 8$$

$$34^{16} = 64$$

$$34^{32} = 71$$

$$34 * 36 * 64 * 71 = 111$$

f) Chiffrer $m = 22$.

$$22^1 = 22$$

$$22^2 = 1$$

$$22^4 = 1$$

$$22 * 1 = 22$$

Le chiffre est identique au message. C'est une situation critique, à éviter.

g) Déchiffrer $c = 100$.

$$100^1 = 100$$

$$100^2 = 18$$

$$100^4 = 2$$

$$100^8 = 4$$

$$100^{16} = 16$$

$$100^{32} = 95$$

$$100 * 2 * 16 * 95 = 32$$

3.7.7

a) Pour le professeur $\varphi(55) = 40$ et $27 \cdot 3 = 81 = 1 \pmod{40}$.

Pour le secrétariat $\varphi(33) = 20$ et $7 \cdot 3 = 21 = 1 \pmod{20}$.

b) Le professeur envoie $m = 12 \pmod{33}$. Or $12^2 = 12 \pmod{33}$; donc $m = 12 \pmod{33}$.

c) La note $(23^3 \pmod{55})^7 \pmod{33} \equiv 12^7 \pmod{33} \equiv 12$.

3.7.8 La cryptographie est l'art de rendre inintelligible, de crypter, de coder, un message pour ceux qui ne sont pas habilités à en prendre connaissance. Le chiffre, le code est le procédé, l'algorithme, la fonction, qui permet de crypter un message.

La cryptanalyse est l'art pour une personne non habilitée, de décrypter, de décoder, de déchiffrer, un message. C'est donc l'ensemble des procédés d'attaque d'un système cryptographique.

La cryptologie est l'ensemble formé de la cryptographie et de la cryptanalyse.

3.7.9

a) Le message codé est $c \equiv 15^3 \pmod{187} \equiv 9$.

b) Ecrivons $N = pq$. On a donc $\varphi(N) = (p-1)(q-1) = pq - p - q + 1 = N - (p+q) + 1$, et ainsi $p + q = N - \varphi(N) + 1 = 187 - 160 + 1 = 28$.

Les nombres p et q sont racines du polynôme

$$X^2 - (p+q)X + pq = X^2 - 28X + 187$$

Le discriminant est $282 - 4 \cdot 187 = 36$ et ainsi $p = (28 - 6)/2 = 11$ et $q = (28 + 6)/2 = 17$.

3.7.10 Puisque e_1 et e_2 sont premiers entre eux, il existe deux entiers u et v tels que $u \cdot e_1 + v \cdot e_2 = 1$. Eve peut calculer u et v , et finalement retrouve le message en faisant

$$c_1^u \cdot c_2^v \equiv m^{ue_1} \cdot m^{ve_2} \equiv m^{ue_1+ve_2} \equiv m \pmod{N}$$

3.7.11

```
# exemple
```

```
p=1093
```

```
q=1091
```

```
n=p*q
```

```
# algorithme
```

```
from math import sqrt
```

```
t=int(sqrt(n))
```

```
z=2
```

```
while int(sqrt(z)) != z:
```

```
    t += 1
```

```
    z = t**2-n
```

```
print('p=',t+sqrt(z))
```


Chapitre 4

Méthodes numériques

4.1 La bibliothèque matplotlib de Python : les fonctions de base

4.1.1 Copier ces lignes dans une fenêtre de Python et observer le résultat.

```
import matplotlib.pyplot as plt
import numpy as np

plt.plot([1,3,4],[2,1,6])

plt.show()
```

4.1.2 Copier ces lignes dans une fenêtre de Python et observer le résultat.

```
import matplotlib.pyplot as plt
import numpy as np

x = np.linspace(-2, 2, 100)
y = 2*x**2+3*x-4

plt.plot(x,y)
plt.show()
```

4.1.3 Copier ces lignes dans une fenêtre de Python et observer le résultat.

```
import matplotlib.pyplot as plt
import numpy as np
from math import *

abscisses = np.linspace(-4,4,100)
ordonnées = [cos(x)+3*sin(2*x) for x in abscisses]

plt.plot(abscisses,ordonnées)
plt.show()
```

4.1.4 Copier ces lignes dans une fenêtre de Python et observer le résultat.

```
import matplotlib.pyplot as plt
import numpy as np

x = np.linspace(-4,4,100)
y = np.cos(x)+3*np.sin(2*x)

plt.plot(x,y)
plt.show()
```

4.1.5 Copier ces lignes dans une fenêtre de Python et observer le résultat.

```
import matplotlib.pyplot as plt
import numpy as np

x = np.linspace(-1,3,100)
y = -2*x+3
plt.plot(x,y)
y = x**2 - 4*x + 4

plt.plot(x,y)
plt.show()
```

4.1.6 Copier ces lignes dans une fenêtre de Python et observer le résultat.

```
import matplotlib.pyplot as plt
import numpy as np

x = np.linspace(-np.pi,np.pi,100)
for n in range(1,5):
    y = np.cos(n*x)
    plt.plot(x,y, label="n="+str(n))

plt.legend(loc="lower right")
plt.show()
```

4.1.7 Afficher la représentation graphique de la fonction $f(x) = x^3 + 3x^2 - 9x + 1$ pour x entre -2 et 4 .

4.1.8 Afficher la représentation graphique de la fonction $f(x) = 3 \cos(2x) - 2 \sin(3x)$ pour x entre $-\pi$ et π .

4.1.9 Afficher les représentations graphiques de la famille de fonctions $f(x) = \frac{1 - nx}{x - 1}$ pour n allant de -3 à 3 . On se placera dans un repère allant de -2 à 4 pour les abscisses et de -8 à 8 pour les ordonnées. On fera apparaitre aussi une légende en haut à gauche.

4.1.10

- a) Tracer en vert la fonction qui à x associe $\log(x)$ pour x variant de 0.1 à 10 avec un pas de 0.1. Utiliser la fonction `np.log()`.
Ajouter les légendes et le titre appropriés.
- b) Tracer en rouge la fonction qui à x associe $\sqrt{x}$ pour x variant de 0.1 à 10 avec un pas de 0.1. Utiliser la fonction `np.sqrt()`.
Ajouter les légendes et le titre appropriés.
- c) Selon vous quelle est la fonction qui croît le plus vite en $+\infty$? qui décroît le plus vite en 0?
Vérifier en superposant les 2 courbes.
- d) Tracer `plt.plot(x, np.exp(np.log(x)))`. Quelle est la forme de la courbe obtenue? Pourquoi obtient-on cette forme?

4.1.11 Tracer la spirale d'Archimède donné par le système d'équations paramétriques.

$$\begin{cases} x(t) &= t \cos(t) \\ y(t) &= t \sin(t) \end{cases}$$

4.1.12 Tracer la courbe de la fonction f dans l'intervalle I ainsi que sa tangente au point d'abscisse a .

- a) $f(x) = x^2$, $I = [-1; 3]$, $a = 1$.
- b) $f(x) = e^x - 3x^2$, $I = [-1; 5]$, $a = 4$.
- c) $f(x) = \sin(x) - \frac{x}{4}$, $I = [-1; \pi]$, $a = 2$.

4.2 Zéros de fonctions : méthode de la bisection

4.2.1 Soit la fonction $f(x) = (x + 2)(x + 1)(x - 1)$.

- a) Représenter la fonction dans l'intervalle $I = [0.1; 2.4]$.
- b) Écrire les six premiers intervalles successifs obtenus par la méthode de la bisection pour chercher les zéros de la fonction $f(x)$ dans l'intervalle I .

4.2.2 Pour chacune des fonctions suivantes, faire trois itérations de la méthode de la bisection à partir de l'intervalle indiqué pour trouver sa racine dans cet intervalle. Puis déterminer le nombre d'itérations nécessaires pour obtenir une solution dont le chiffre des millièmes est significatif.

- a) $f(x) = 0.3x^2 - 0.6x - 0.8$ dans l'intervalle $[2.8; 3]$.
- b) $f(x) = x^5 + x^3 - 1$ dans l'intervalle $[0; 1]$.
- c) $f(x) = (x \cos(x))^2 - 2.3$ dans l'intervalle $[1.3; 5.3]$.

4.2.3 On considère l'équation :

$$e^x - (x + 5) = 0$$

- a) Déterminer le nombre et la position approximative des solutions positives de cette équation.

- b) Utiliser l'algorithme de la bisection pour déterminer chacune de ces racines avec une erreur absolue inférieure à 10^{-7} .

4.2.4 Montrer que chacune des fonctions f suivantes possède un zéro unique. Calculer ce zéro par l'algorithme de la bisection.

- | | |
|---|-----------------------------|
| a) $f(x) = x^3 + 10$ | d) $f(x) = 3x - \cos(x)$ |
| b) $f(x) = x^5 - 2x^4 + 100x^3 - 2$ | e) $f(x) = \ln x - \cos(x)$ |
| c) $f(x) = x - 1 + \frac{1}{2} \sin(x)$ | f) $f(x) = x + e^x$ |

4.2.5 Montrer que la fonction $f = x^3 + 2x - 1$ possède un zéro dans l'intervalle $I = [0 ; 1]$. Calculer ce zéro par l'algorithme de la bisection.

4.2.6 Déterminer une valeur approchée à 10^{-6} près des éventuels zéros de la fonction

$$f(x) = x^7 + 23x^5 + 2x^4 - 2$$

Calculer ce zéro par l'algorithme de la bisection.

4.2.7 Utiliser la méthode de la dichotomie pour obtenir une valeur approchée du nombre $\sqrt[3]{2}$ à 10^{-4} près.

4.2.8 Utiliser la méthode de la dichotomie pour obtenir une valeur approchée du nombre π .

4.3 Zéros de fonctions : méthode de Newton

4.3.1 Soit la fonction

$$f(x) = -5x^3 + 7x^2 + 3x - 3$$

- Calculer les coordonnées des points à tangente horizontale et du point d'inflexion. Tracer le graphe de f dans l'intervalle $[-1 ; 2]$.
- Calculer la valeur des premiers termes de la suite de Newton si l'on choisit l'estimation initial $x_0 = 1$.
- Calculer la valeur des premiers termes de la suite de Newton si l'on choisit l'estimation initial $x_0 = 2$.
- Déterminer l'unique zéro de f compris entre 1 et 2.

4.3.2 L'équation

$$e^x - 3x^2 = 0$$

possède les solutions $s_1 = -0.4589623$ et $s_2 = 0.91$ ainsi qu'une troisième solution s_3 située près de 4.

Utiliser la méthode de Newton pour déterminer s_3 avec six décimales.

4.3.3 Trouver une estimation d'un zéro strictement positif des fonctions suivantes en appliquant la méthode de Newton.

a) $f(x) = x^3 - 2$

b) $f(x) = \sin(x) - \frac{x}{4}$

4.3.4 Soit la fonction

$$f(x) = x^3 - 2x - 5$$

a) Montrer que l'équation $f(x) = 0$ admet une unique solution α sur $\mathbb{R}$. Montrer que $2 < \alpha < 3$.

b) A partir de $x_0 = 1$, déterminer le zéro de la fonction.

4.3.5 Donner une approximation de $\sqrt{2}$ avec la méthode de Newton en partant du point $x_0 = 2$.

4.3.6 Une des difficultés de la méthode de Newton est de bien choisir x_0 de sorte que l'on ne tombe pas sur un point où la dérivée s'annule.

Utiliser l'algorithme de Newton avec la fonction $f(x) = \cos(x)$ en partant des points d'abscisse 1, 0.5, 0.1 puis 0.01.

Commenter et expliquer la situation à l'aide d'un graphique.

4.4 Zéros de fonctions : méthode de la sécante

4.4.1 A l'aide de la méthode de la sécante, calculer la racine douzième de 1.1 avec une précision de 8 décimales.

4.4.2 Rechercher, en utilisant la méthode de la sécante, les solutions de l'équation

$$x^3 + 1 = 3x$$

4.4.3 Calculer dans l'intervalle $I = [0; \pi]$ une approximation de la solution de l'équation

$$\cos(x) = 2 \sin(x)$$

4.4.4 Rechercher, en utilisant la méthode de la sécante, les solutions de l'équation

$$e^{-x} = -\ln(x)$$

4.5 Un peu d'intégration numérique

4.5.1 Sans utiliser de bibliothèque autre que la bibliothèque mathématique standard de python, écrire une fonction qui prend deux nombres réels $a < b$ et un nombre entier n en paramètres. Cette fonction doit renvoyer une liste de $n + 1$ valeurs réelles, notées $x_0, x_1, \dots, x_n$ telles que $x_0 = a$, $x_n = b$, tous les autres nombres étant uniformément répartis dans l'intervalle $[a, b]$.

L'instruction

```
print(listePoints(0, 2, 10))
```

fait afficher la liste

[0, 0.2, 0.4, 0.6, 0.8, 1.0, 1.2, 1.4, 1.6, 1.8, 2]

à quelques erreurs d'arrondi près.

4.5.2 Compléter le code de la fonction de l'exercice précédent en lui ajoutant un paramètre optionnel `aleatoire=False`. Par défaut, le comportement de la fonction est le même que celle de l'exercice précédent. Si le paramètre `aleatoire` prend la valeur `True`, la fonction doit renvoyer une liste de $n + 1$ valeurs réelles, notées $x_0, x_1, \dots, x_n$ telles que $x_0 = a$, $x_n = b$, tous les autres nombres étant aléatoirement choisis dans l'intervalle $[a, b]$.

L'instruction

```
print(listePoints(0, 2, 10)) # Valeur par défaut de aleatoire
```

fait afficher la liste

```
[0, 0.2, 0.4,
 0.6000000000000001, 0.8, 1.0, 1.2, 1.4,
 1.5999999999999999, 1.7999999999999998,
 1.9999999999999998]
```

L'instruction

```
print(listePoints(0, 2, 10, True))
```

fait afficher la liste

```
[0,
 0.06479694651588885, 0.28981342176727964, 0.3179007067671271,
 0.4070274425545328, 1.015739468424949, 1.1182123677493638,
 1.2618287932253878, 1.473438685747842, 1.519412756197574,
 2]
```

Il faut noter que le résultat change à chaque fois que le programme tourne.

4.5.3 Créer une classe `Fonction` qui stocke dans un champ une fonction mathématique f . Dans cette classe, créer une méthode dont le nom est `listeValeurs` qui prend une liste de nombres en paramètre et qui renvoie la liste des valeurs prises par la fonction f en chacun des points.

Importer le module `math` de python et vérifier que l'exécution des lignes

```
lp = listePoints(0, math.pi / 2, 10)
f = Fonction(math.sin)
for val in f.listeValeurs(lp):
    print(val)
```

donne l'affichage suivant :


```

0.0
0.15643446504023087
0.3090169943749474
0.45399049973954675
0.5877852522924731
0.7071067811865475
0.8090169943749473
0.8910065241883678
0.9510565162951535
0.9876883405951378
1.0

```

4.5.4 Compléter la classe `Fonction` avec une méthode dont l'en-tête est

```
def sommeRect(self, a, b, n, aleatoire=False)
```

qui renvoie

- la liste des n points choisis dans l'intervalle $[a, b]$;
- une liste de valeurs définies par l'expression

$$\min(f(x_i); f(x_{i+1}))$$

pour i compris entre 0 et $n - 1$.

- Une approximation de

$$\int_a^b f(x) dx$$

par une somme d'aires de rectangles, en sélectionnant à chaque itération le rectangle « sous la courbe ».

- une liste de valeurs définies par l'expression

$$\max(f(x_i); f(x_{i+1}))$$

pour i compris entre 0 et $n - 1$.

- Une approximation de

$$\int_a^b f(x) dx$$

par une somme d'aires de rectangles, en sélectionnant à chaque itération le rectangle « en dessus de la courbe ».

La valeur par défaut du paramètre `aleatoire` donne un découpage régulier de l'intervalle $[a, b]$. Si la valeur `True` est passée en paramètre, le découpage est aléatoire.

Faire calculer

$$\int_0^2 (2x^2 - x^3) dx$$

en découpant l'intervalle en 10 morceaux, uniformément ou aléatoirement.

Écrire les instructions qui donnent l'affichage suivant lorsque le programme tourne :

Intervalle découpé en 10 morceaux identiques :

1.0848000000000002
1.5552

Intervalle découpé aléatoirement en 10 morceaux (5 essais)

0.8943893487419508
1.7009476901969376
*
0.8878935584553052
1.770292543499321
*
0.8352384010982953
1.5033073760799847
*
0.9881561937421527
1.5762026218159841
*
0.8617505588265767
1.7164006543400632

4.5.5 À l'aide de la méthode `sommeRect()` de la classe `Fonction`, calculer une approximation de

$$\int_{-0.5}^1 e^{-x^2} dx$$

dont les 6 premières décimales sont correctes, en découpant l'intervalle $[-0.5, 1]$ en n morceaux. Quelle est la plus petite valeur de n qui garantit ce résultat ?

4.5.6 Pour $n \in \{10, 100, 1000, 10000, 100000\}$, faire calculer une approximation de

$$\int_1^2 \frac{1}{x} dx$$

uniforme et aléatoire.

- a) Comparer les valeurs obtenues par les deux méthodes.
- b) Comparer avec la valeur exacte obtenue par le calcul de la primitive.

Quelle conclusion en tirer ?

4.5.7 Compléter la classe `Fonction` avec une méthode dont l'en-tête est

```
def sommeTrap(self, a, b, n, aleatoire=False)
```

qui renvoie une approximation de

$$\int_a^b f(x) dx$$

à l'aide d'une somme d'aires de trapèzes. La somme peut être calculée de la façon suivante :

Pour tout i compris entre 0 et $n - 1$, l'expression

$$(x_{i+1} - x_i) \cdot \frac{(f(x_i) + f(x_{i+1})))}{2}$$

contribue à la somme totale.

4.5.8 À l'aide de la méthode `sommeTrap()` de la classe `Fonction`, calculer une approximation de

$$\int_{-0.5}^1 e^{-x^2} dx$$

dont les 6 premières décimales sont correctes, en découpant l'intervalle $[-0.5, 1]$ en n morceaux. Quelle est la plus petite valeur de n qui garantit ce résultat ?

4.5.9 Compléter la classe `Fonction` avec une méthode dont l'en-tête est

```
def sommeSim(self, a, b, n, aleatoire=False)
```

qui renvoie une approximation de

$$\int_a^b f(x) dx$$

à l'aide d'une somme d'aires, calculées de la façon suivante :

- L'intervalle $[a, b]$ est découpé en n sous-intervalles, régulièrement ou non.
- On détermine pour chaque sous-intervalle $I_i = [x_i, x_{i+1}]$ l'expression mathématique de la parabole qui passe par les trois points $(x_i, f(x_i))$, $(m, f(m))$ et $(x_{i+1}, f(x_{i+1}))$ où m désigne le milieu de l'intervalle I_i .
- On calcule ensuite la valeur exacte de l'intégrale de cette parabole sur l'intervalle I_i , qui contribue à la somme totale.
- Ce calcul ayant été fait pour tout i , on renvoie la somme qui donne l'approximation de l'intégrale.

4.6 Solutions des exercices

4.1.1 —

4.1.2 La fonction `np.linspace(debut, fin, nombre)` permet de créer une liste de `N` nombres qui commencent à la valeur `debut` et s'arrête à la valeur `fin` et uniformément répartis.

4.1.3 Tracé de $y = \cos(x) + 3 \sin(2x)$

4.1.4 Tracé de $y = \cos(x) + 3 \sin(2x)$

4.1.5 —

4.1.6 —

4.1.7

```
import matplotlib.pyplot as plt
import numpy as np
def f(x):
    return x**3+3*(x**2)-9*x+1
x = np.linspace(-2,4,100)
plt.plot(x,f(x))
plt.grid(True)
plt.title("Exercice 5.1.7")
plt.xlabel('x')
plt.ylabel('f(x)')
plt.show()
```

4.1.8

```
import matplotlib.pyplot as plt
import numpy as np
import math
pi = math.pi
```

```
def f(x):  
    return 3*np.cos(2*x)-2*np.sin(3*x)  
x = np.linspace(-pi,pi,100)  
plt.plot(x,f(x))  
plt.grid(True)  
plt.title("Exercice 5.1.8")  
plt.xlabel('x')  
plt.ylabel('f(x)')  
plt.show()
```

4.1.9

Il faut faire attention à ne pas représenter la droite $x = 1$.

```
import matplotlib.pyplot as plt  
import numpy as np  
def f(x, n):  
    return (1-n*x)/(x-1)  
x = np.linspace(-2,4,201)  
for n in range(-3, 4):  
    l = 'n = ' + str(n)  
    plt.plot(x,f(x, n), label=l)  
plt.grid(True)  
plt.title("Exercice 5.1.9")  
plt.xlabel('x')  
plt.ylabel('f(x)')  
plt.xlim(-2, 4)  
plt.ylim(-8, 8)  
plt.legend()  
plt.show()
```

4.1.10

```
import matplotlib.pyplot as plt
```

```
import numpy as np
x = np.linspace(0.1,10,100)
plt.plot(x,np.log(x),color="green",label="log(x)")
plt.plot(x,np.sqrt(x),color="red",label="sqrt(x)")
plt.plot(x,np.exp(np.log(x)),color="blue")
plt.grid(True)
plt.title("Exercice 5.1.10")
plt.xlabel('x')
plt.ylabel('f(x)')
plt.legend()
plt.show()
```

On a $e^{\ln(x)} = x$.

4.1.11

```
import matplotlib.pyplot as plt
import numpy as np
t = np.linspace(0, 10 * np.pi, 300) # Spirale d'Archimede
x = t * np.cos(t)
y = t * np.sin(t)
plt.plot(x, y)
plt.show()
```

4.1.12

Par exemple :

```
import matplotlib.pyplot as plt
import numpy as np
def f(x):
    return x*x
def df(x):
    return 2*x
x = np.linspace(-1,3,100)
```

```
y = df(1)*(x-1)+f(1)
plt.plot(x,f(x),color="green",label="x^2")
plt.plot(x,y,color="red",label="tangente")
plt.grid(True)
plt.title("Exercice 5.1.12")
plt.xlabel('x')
plt.ylabel('f(x)')
plt.legend()
plt.show()
```

4.2.1

$I_0 = [0.1; 2.4]$; $I_1 = [0.1; 1.25]$; $I_2 = [0.675; 1.25]$; $I_3 = [0.9625; 1.25]$; $I_4 = [0.9625; 1.10625]$;
 $I_5 = [0.9625; 1.034375]$

4.2.2

- a) 2.9; 2.95; 2.925. Il faut au moins neuf itérations.
- b) 0.5; 0.75; 0.875. Il faut au moins onze itérations.
- c) 3.3; 2.3; 1.8. Il faut au moins treize itérations.

4.2.3

- a) Si $f(x) = e^x - (x + 5)$, alors $f'(x) = e^x - 1$ est positive pour $x > 0$ et donc f est strictement croissante sur $\mathbb{R}^+$. Par conséquent, f ne croise l'axe des x qu'une seule fois. Puisque $f(0) = -4$ et $f(2) = 0.38$, il y a une seule racine entre $x = 0$ et $x = 2$.
- b) On obtient $x = 1.9368473291$ en 23 itérations à partir de l'intervalle $[0; 2]$.

4.2.4

- | | |
|---------------|---------------|
| a) -2.1544347 | d) 0.3167508 |
| b) 0.2718682 | e) 1.302964 |
| c) 0.6840367 | f) -0.5671433 |

4.2.5

$x = 0.4533976515$

4.2.6

$$x = 0.595471$$

4.2.7 —**4.2.8** —**4.3.1**

a) minimum : $(-0.18; -3.28)$, maximum : $(1.11; 2.12)$, point d'inflexion : $(0.47; -0.58)$

b) 0, 1, 0, 1, 0, 1, 0, ...

c) 1.6896552, 1.5618930, 1.5372456, 1.5363390, 1.5363378

d) 1.5363378

4.3.2

$$s_3 = 3.733079$$

4.3.3

a) 1.259921050

b) 2.474576794

4.3.4

a) A partir du tableau des variations de f

b) 2.0945514815

4.3.5 —**4.3.6** —**4.4.1**

$$r = 1,00796779$$

4.4.2

$$x_1 = -1.8793852416, x_2 = 0.3472963553 \text{ et } x_3 = 1.5320888862$$

4.4.3

$x = 0.4636476092$

4.4.4

$x = 0.5671432873$

4.5.1

```
def listePoints(a, b, n):  
    if a > b:  
        a, b = b, a  
    lp = [a]  
    delta = (b - a) / n  
    for i in range(n):  
        a = a + delta  
        lp.append(a)  
    return lp
```

4.5.2

```
import random  
  
def listePoints(a, b, n, aleatoire=False):  
    if a > b:  
        a, b = b, a  
    lp = [a]  
    if not aleatoire:  
        delta = (b - a) / n  
        for i in range(n):  
            a = a + delta  
            lp.append(a)  
    else:  
        for i in range(n - 1):  
            x = a + (b - a) * random.random()
```

```
        lp.append(x)
    lp.append(b)
    lp.sort()
    return lp
```

4.5.3

```
import math
import random

def listePoints(a, b, n, aleatoire=False):
    if a > b:
        a, b = b, a
    lp = [a]
    if not aleatoire:
        delta = (b - a) / n
        for i in range(n):
            a = a + delta
            lp.append(a)
    else:
        for i in range(n - 1):
            x = a + (b - a) * random.random()
            lp.append(x)
        lp.append(b)
        lp.sort()
    return lp

class Fonction(object):
    def __init__(self, f):
        self.f = f
```

```
def listeValeurs(self, lp):
    lv = []
    for x in lp:
        lv.append(self.f(x))
    return lv

if __name__ == "__main__":
    lp = listePoints(0, math.pi / 2, 10)
    f = Fonction(math.sin)
    for val in f.listeValeurs(lp):
        print(val)
```

4.5.4

```
import math
import random

def listePoints(a, b, n, aleatoire=False):
    if a > b:
        a, b = b, a
    lp = [a]
    if not aleatoire:
        delta = (b - a) / n
        for i in range(n):
            a = a + delta
            lp.append(a)
    else:
        for i in range(n - 1):
            x = a + (b - a) * random.random()
            lp.append(x)
```

```
        lp.append(b)
        lp.sort()
    return lp
```

```
class Fonction(object):
    def __init__(self, f):
        self.f = f

    def listeValeurs(self, lp):
        lv = []
        for x in lp:
            lv.append(self.f(x))
        return lv

    def sommeRect(self, a, b, n, aleatoire=False):
        lp = listePoints(a, b, n, aleatoire)
        sSup, sInf = 0.0, 0.0
        lvSup = []
        lvInf = []
        for i in range(n):
            delta = lp[i + 1] - lp[i]
            valeurs = self.f(lp[i]), self.f(lp[i + 1])

            v_inf = min(valeurs)
            lvInf.append(v_inf)
            sInf += v_inf * delta

            v_sup = max(valeurs)
            lvSup.append(v_sup)
            sSup += v_sup * delta
```

```
        return lp, lvSup, sSup, lvInf, sInf

if __name__ == "__main__":
    g = Fonction(lambda x : 2 * x**2 - x**3)
    lp, lvSup, sSup, lvInf, sInf = g.sommeRect(0.0, 2.0, 10)
    print("Intervalle découpé en 10 morceaux identiques : ")
    print()
    print(sInf)
    print(sSup)
    print()
    print("Intervalle découpé aléatoirement en 10 morceaux (5 essais)")
    print()
    for i in range(5):
        lp, lvSup, sSup, lvInf, sInf = g.sommeRect(0.0, 2.0, 10, True)
        # print(lv)
        print(sInf)
        print(sSup)
        print("*")
```

4.5.5

```
if __name__ == "__main__":
    h = Fonction(lambda x : math.e**(-x**2))
    n = 100
    lp, lvSup, sSup, lvInf, sInf = h.sommeRect(-0.5, 1.0, n)
    print("Intervalle découpé en {} morceaux identiques : ".format(n))
    print()
    print(sInf)
    print(sSup)
    print()
    while sSup - sInf > 0.000001:
```

```

        n *= 10

        lp, lvSup, sSup, lvInf, sInf = h.sommeRect(-0.5, 1.0, n)
    print(sSup)
    print(sInf)
    print(n)
    print("Valeur donnée par le site Wolfram Alpha")
    print("1.2081051392252194741551703728723061084382587757768974")

if __name__ == "__main__":
    aire = 0.693147180559836
    g = Fonction(lambda x : 1 / x)
    nombres = [10, 100, 1000, 10000, 100000]
    for n in nombres:
        lp, lvSup, sSup, lvInf, sInf = g.sommeRect(1.0, 2.0, n)
        lpa, lvSupa, sSupa, lvInfra, sInfra = g.sommeRect(1.0, 2.0, n, True)
        print(sInf, sInfra, sSup, sSupa)
        print(5 * "*")
        lpa, lvSupa, sSupa, lvInfra, sInfra = g.sommeRect(1.0, 2.0, n, True)
    print(5 * "*")
    print(aire)

def sommeTrap(self, a, b, n, aleatoire=False):
    lp = listePoints(a, b, n, aleatoire)
    s = 0.0
    lv = []
    for i in range(n):
        delta = lp[i + 1] - lp[i]
        valeur = (self.f(lp[i]) + self.f(lp[i + 1])) * 0.5

        lv.append(valeur)
        s += delta * valeur

```

```
    return lp, lv, s
```

4.5.6 4.5.7 4.5.8

```
if __name__ == "__main__":
    h = Fonction(lambda x : math.e**(-x**2))
    n = 100
    lp_1, lv_1, s_1 = h.sommeTrap(-0.5, 1.0, n)
    n *= 10
    lp_2, lv_2, s_2 = h.sommeTrap(-0.5, 1.0, n)
    while abs(s_2 - s_1) > 0.000001:
        lp_1, lv_1, s_1 = h.sommeTrap(-0.5, 1.0, n)
        n *= 10
        lp_2, lv_2, s_2 = h.sommeTrap(-0.5, 1.0, n)
    print(s_2)
    print(n)
    print("Valeur donnée par le site Wolfram Alpha")
    print("1.2081051392252194741551703728723061084382587757768974")

# Deux fonctions auxiliaires
def coeffParaboleParTroisPoints(x_1, y_1, x_2, y_2, x_3, y_3):
    a = ((y_3 - y_1)/(x_3 - x_1) - (y_2 - y_1)/(x_2 - x_1)) / (x_3 - x_2)
    b = -a * (x_2 + x_1) + (y_2 - y_1)/(x_2 - x_1)
    c = y_1 - a * x_1**2 - b * x_1
    return a, b, c

def primPara(a, b, c):
    return lambda x: 1/3 * a * x**3 + 1/2 * b * x**2 + c * x

# Méthode de la classe Fonction
def sommeSim(self, a, b, n, aleatoire=False):
    lp = listePoints(a, b, n, aleatoire)
    s = 0.0
```

```
for i in range(n):
    x_3 = lp[i + 1]
    y_3 = self.f(x_3)
    x_2 = (lp[i + 1] + lp[i]) / 2
    y_2 = self.f(x_2)
    x_1 = lp[i]
    y_1 = self.f(x_1)
    a, b, c = coeffParaboleParTroisPoints(x_1, y_1, x_2, y_2, x_3, y_3)
    F = primPara(a, b, c)
    s += F(x_3) - F(x_1)

return lp, s
```