RAPPELS DE COLLÈGE

Calcul numérique

Les priorités opératoires

• **Pour calculer une expression sans parenthèses**, on effectue d'abord les puissances, puis les multiplications et les divisions, enfin les additions et les soustractions.

Exemple

$5-2\times3^2=5-2\times9=5-18=-13$

• **Pour calculer une expression avec des parenthèses**, on effectue d'abord les calculs entre parenthèses.

Exemple

$(6-2^3)\times5=(6-8)\times5=-2\times5=-10$

Les fractions

a, b, c, d désignent des nombres ($c\neq0$, $d\neq0$).

• $\dfrac{a}{c}+\dfrac{b}{c}=\dfrac{a+b}{c}$ • $\dfrac{a}{c}-\dfrac{b}{c}=\dfrac{a-b}{c}$ • $\dfrac{a}{c}\times\dfrac{b}{d}=\dfrac{a\times b}{c\times d}$ • $a\times\dfrac{b}{c}=\dfrac{a\times b}{c}$ • $\dfrac{a}{b}:\dfrac{c}{d}=\dfrac{a}{b}\times\dfrac{d}{c}$ $(b\neq0)$

Les puissances

a désigne un nombre relatif et n désigne un nombre entier positif.

• Pour $n\geqslant2$, $10^n=1\underbrace{0\ldots0}_{n\text{ zéros}}$; $10^1=10$; $10^0=1$

• $10^{-n}=\dfrac{1}{10^n}=\dfrac{1}{1\underbrace{0\ldots0}_{n\text{ zéros}}}=\underbrace{0{,}0\ldots0}_{n\text{ zéros}}1$

• Pour $n\geqslant2$, $a^n=\underbrace{a\times a\times\ldots\times a}_{n\text{ facteurs}}$;

$a^1=a$; $a^0=1$ $(a\neq0)$.

• Si $a\neq0$, $a^{-n}=\dfrac{1}{a^n}$.

• La **notation scientifique** d'un nombre décimal non nul est l'écriture de la forme $a\times10^n$, a étant écrit avec un seul chiffre, autre que 0, avant la virgule.

Exemples

$178\,500=1{,}785\times10^5$ $0{,}006\,82=6{,}82\times10^{-3}$

• **Puissances de 10 et préfixes**

Préfixe	giga	méga	kilo	unité	milli	micro	nano
Symbole	G	M	k		m	µ	n
10^n	10^9	10^6	10^3	$10^0=1$	10^{-3}	10^{-6}	10^{-9}

Calcul littéral

Développement – Factorisation

a, b, c, d, k désignent des nombres.

Développer

$k(a+b)=ka+kb$

$(a+b)(c+d)=ac+ad+bc+bd$

$(a+b)(a-b)=a^2-b^2$

Factoriser

Équation produit nul

a, b, c, d désignent des nombres ($a\neq c$).

• Les solutions d'une équation produit nul :

$(ax+b)(cx+d)=0$

sont les nombres x tels que :

$ax+b=0$ ou $cx+d=0$

Équation du premier degré

• Résoudre l'équation $x+1=3x+5$

$x+1-3x=3x+5-3x$

$-2x+1-1=5-1$

$\dfrac{-2x}{-2}=\dfrac{4}{-2}$

$x=-2$

On regroupe tous les termes en x dans un membre.

L'équation a pour seule solution -2.

Équation $x^2=a$

a désigne un nombre.

• Si $a>0$, l'équation admet exactement deux solutions, $\sqrt{a}$ et $-\sqrt{a}$.

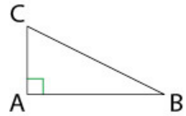
• Si $a=0$, l'équation admet une solution, 0.

• Si $a<0$, l'équation n'a pas de solution.

Triangles rectangles

Théorème de Pythagore

- Si ABC est un triangle rectangle en A, alors $BC^2 = AB^2 + AC^2$.
- **Réciproque :** Si ABC est un triangle tel que $BC^2 = AB^2 + AC^2$, alors ce triangle est rectangle en A.



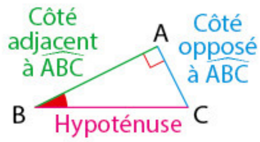
Trigonométrie

ABC est un triangle rectangle en A.

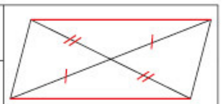
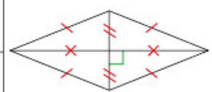
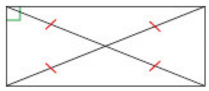
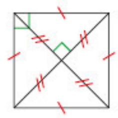
$$\cos(\widehat{ABC}) = \frac{\text{longueur du côté adjacent à } \widehat{ABC}}{\text{longueur de l'hypoténuse}} = \frac{AB}{BC}$$

$$\sin(\widehat{ABC}) = \frac{\text{longueur du côté opposé à } \widehat{ABC}}{\text{longueur de l'hypoténuse}} = \frac{AC}{BC}$$

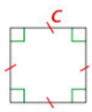
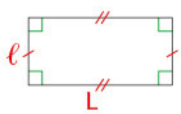
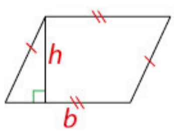
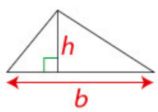
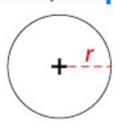
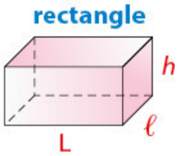
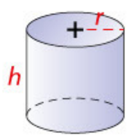
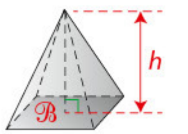
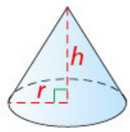
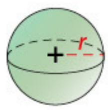
$$\tan(\widehat{ABC}) = \frac{\text{longueur du côté opposé à } \widehat{ABC}}{\text{longueur du côté adjacent à } \widehat{ABC}} = \frac{AC}{AB}$$



Quadrilatères particuliers

Parallélogramme	• Quadrilatère qui a ses côtés opposés parallèles. • Quadrilatère dont les diagonales se coupent en leur milieu.	
Losange	• Parallélogramme qui a deux côtés consécutifs de même longueur. • Parallélogramme dont les diagonales sont perpendiculaires.	
Rectangle	• Parallélogramme qui a un angle droit. • Parallélogramme dont les diagonales ont même longueur.	
Carré	• Quadrilatère qui est à la fois un rectangle et un losange. • Quadrilatère dont les diagonales se coupent en leur milieu, ont même longueur et sont perpendiculaires.	

Longueurs, aires et volumes

Carré	**Rectangle**	**Parallélogramme**	**Triangle**	**Cercle, disque**
 Aire : $\mathcal{A} = c^2$	 Aire : $\mathcal{A} = L \times \ell$	 Aire : $\mathcal{A} = b \times h$	 Aire : $\mathcal{A} = \frac{b \times h}{2}$	 Longueur : $L = 2\pi r$ Aire : $\mathcal{A} = \pi r^2$
Parallélépipède rectangle	**Cylindre**	**Pyramide**	**Cône**	**Boule**
 Volume : $\mathcal{V} = L \times \ell \times h$	 Volume : $\mathcal{V} = \pi r^2 h$	 Volume : $\mathcal{V} = \frac{1}{3}\mathcal{B}h$	 Volume : $\mathcal{V} = \frac{1}{3}\pi r^2 h$	 Volume : $\mathcal{V} = \frac{4}{3}\pi r^3$

FORMULAIRE DE SECONDE

Les ensembles de nombres

Ensemble des nombres : • entiers naturels : $\mathbb{N}$ • entiers relatifs : $\mathbb{Z}$ • décimaux : $\mathbb{D}$ • rationnels : $\mathbb{Q}$
• réels : $\mathbb{R}$ (un nombre réel non rationnel est dit **irrationnel** : $\sqrt{2}$, π, …)

Ensemble des nombres réels x	Représentation	Intervalle
$a \leqslant x \leqslant b$	a [——] b	$[a\,;b]$
$a \leqslant x < b$	a [——[b	$[a\,;b[$
$x < b$	——[b	$]-\infty\,;b[$
$x \geqslant a$	a [——→	$[a\,;+\infty[$

• $\mathbb{R} =]-\infty\,;+\infty[$ • $\mathbb{R}^* =]-\infty\,;0[\,\cup\,]0\,;+\infty[$ • $\mathbb{R}-\{1\} =]-\infty\,;1[\,\cup\,]1\,;+\infty[$

Calcul numérique

Les puissances

a désigne un nombre réel non nul, n et p deux nombres entiers relatifs.

• $a^n \times a^p = a^{n+p}$ • $\dfrac{a^n}{a^p} = a^{n-p}$ • $(a^n)^p = a^{n\times p}$

a et b désignent deux nombres réels non nuls, n un nombre entier relatif.

• $a^n \times b^n = (ab)^n$ • $\dfrac{a^n}{b^n} = \left(\dfrac{a}{b}\right)^n$

Les racines carrées

• a désigne un nombre réel positif.
$\sqrt{a}$ est le nombre réel positif dont le carré est a.
Ainsi, $\sqrt{a} \geqslant 0$ et $\left(\sqrt{a}\right)^2 = a$.

• Pour tout nombre réel a, $\sqrt{a^2} = |a|$.

• a et b désignent deux nombres réels positifs.

$\sqrt{ab} = \sqrt{a} \times \sqrt{b}$ $\quad \sqrt{\dfrac{a}{b}} = \dfrac{\sqrt{a}}{\sqrt{b}}$ $(b \neq 0)$

Calcul littéral

Identités remarquables

a et b désignent des nombres réels.

Développer

• $(a+b)^2 = a^2 + 2ab + b^2$
• $(a-b)^2 = a^2 - 2ab + b^2$
• $(a+b)(a-b) = a^2 - b^2$

Factoriser

Inéquation

• Résoudre l'inéquation $2x - 3 < 5x + 9$

$2x - 3 - 5x < 5x + 9 - 5x$

$-3x - 3 + 3 < 9 + 3$

$\dfrac{-3x}{-3} > \dfrac{12}{-3}$

$x > -4$

On regroupe tous les termes en x dans un membre.

L'ensemble des solutions est $\mathcal{S} =]-4\,;+\infty[$.

Signe de $ax+b$

• **Cas où $a > 0$**

x	$-\infty$		$-\dfrac{b}{a}$		$+\infty$
Signe de $ax+b$		$-$	0	$+$	

• **Cas où $a < 0$**

x	$-\infty$		$-\dfrac{b}{a}$		$+\infty$
Signe de $ax+b$		$+$	0	$-$	

Signe du produit $(2x+3)(-x+2)$

x	$-\infty$		$-\dfrac{3}{2}$		2		$+\infty$
$2x+3$		$-$	0	$+$		$+$	
$-x+2$		$+$		$+$	0	$-$	
Produit		$-$	0	$+$	0	$-$	

L'ensemble des solutions de l'inéquation $(2x+3)(-x+2) \geqslant 0$ est l'intervalle $\left[-\dfrac{3}{2}\,;2\right]$.

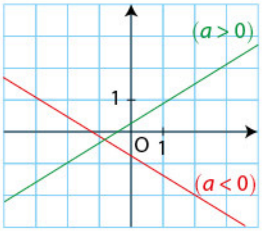
Signe du quotient $\dfrac{2x+3}{-x+2}$

x	$-\infty$		$-\dfrac{3}{2}$		2		$+\infty$
$2x+3$		$-$	0	$+$		$+$	
$-x+2$		$+$		$+$	0	$-$	
Quotient		$-$	0	$+$	‖	$-$	

L'ensemble des solutions de l'inéquation $\dfrac{2x+3}{-x+2} \geqslant 0$ est l'intervalle $\left[-\dfrac{3}{2}\,;2\right[$.

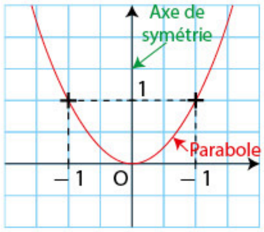
Fonctions

Fonction affine $f : x \mapsto ax + b$



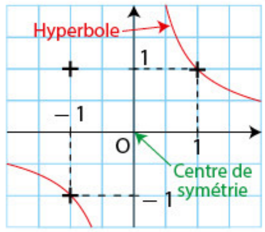
Si $a > 0$, f est croissante sur $\mathbb{R}$.
Si $a < 0$, f est décroissante sur $\mathbb{R}$.

Fonction carré $f : x \mapsto x^2$



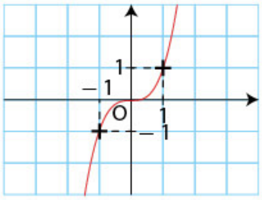
f est décroissante sur $]-\infty\,;0]$ et croissante sur $[0\,;+\infty[$.

Fonction inverse $f : x \mapsto \dfrac{1}{x}$



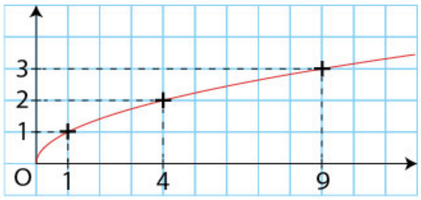
f est décroissante sur $]-\infty\,;0[$ et décroissante sur $]0\,;+\infty[$.

Fonction cube $f : x \mapsto x^3$



f est croissante sur $\mathbb{R}$.

Fonction racine carrée $f : x \mapsto \sqrt{x}$



f est croissante sur $[0\,;+\infty[$.

Statistiques et probabilités

Exemple de série statistique

Mesures de l'envergure d'un lot de papillons

Envergure (mm)	51	52	53	54	55
Effectif	5	8	11	7	2

```
1 variable
x̄     =52.7878787
Σx    =1742
Σx²   =91998
σx    =1.12162154
sx    =1.13901206
n     =33
```

```
1 variable
minX =51
Q1   =52
Med  =53
Q3   =54
maxX =55
Mod  =53
```

Effectif total : $N = 33$

Médiane : c'est la 17^{e} valeur : $Me = 53$

1er quartile : $\dfrac{N}{4} = 8{,}25$ donc Q_1 est la 9^{e} valeur : $Q_1 = 52$

3^{e} quartile : $\dfrac{3N}{4} = 24{,}75$ donc Q_3 est la 25^{e} valeur : $Q_3 = 54$

Moyenne : $m = \dfrac{5 \times 51 + 8 \times 52 + 11 \times 53 + 7 \times 54 + 2 \times 55}{33}$

et $m \approx 52{,}8$

Variance : $V \approx 1{,}26$

Écart-type : $s = \sqrt{V}$ et $s \approx 1{,}12$

Probabilités

E : univers d'une expérience aléatoire

Dans une situation d'équiprobabilité, la probabilité d'un événement A est donnée par :

$$P(A) = \frac{\text{nombre d'issues de A}}{\text{nombre d'issues de E}}$$

Pour tous événements A et B :

- $0 \leqslant P(A) \leqslant 1$
- $P(A \cup B) + P(A \cap B) = P(A) + P(B)$
- $P(A) + P(\overline{A}) = 1$

Points et vecteurs

Le plan est muni d'un repère orthonormé.

On considère les points $A(x_A\,;y_A)$ et $B(x_B\,;y_B)$.

- Les coordonnées du milieu I de $[AB]$ sont :

$$x_I = \frac{x_A + x_B}{2} \qquad y_I = \frac{y_A + y_B}{2}$$

- La distance entre A et B est :

$$AB = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}$$

- Les coordonnées de $\overrightarrow{AB}$ sont $(x_B - x_A\,;y_B - y_A)$.

On considère les vecteurs $\vec{u}(x\,;y)$ et $\vec{v}(x'\,;y')$.

- Coordonnées de $\vec{u} + \vec{v}$: $(x + x'\,;y + y')$
- Coordonnées de $\lambda\vec{u}$ $(\lambda \in \mathbb{R})$: $(\lambda x\,;\lambda y)$
- On suppose de plus $\vec{u} \neq \vec{0}$ et $\vec{v} \neq \vec{0}$.

Dire que $\vec{u}$ et $\vec{v}$ sont colinéaires signifie qu'il existe un nombre réel k tel que $\vec{v} = k\vec{u}$.

- $\vec{u}$ et $\vec{v}$ colinéaires équivaut à $xy' - x'y = 0$.

Mathématiques

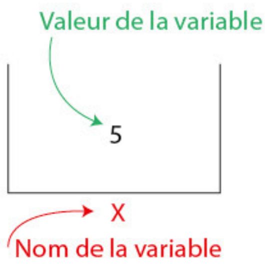
Hyperbole 2de

Des outils pour l'Algorithmique et la Programmation, pour l'utilisation des TICE et la Logique

1 Les instructions d'entrée-sortie, l'affectation, les variables

A Notion de variable

Dans un programme, une variable est repérée par son nom et possède une valeur qui évolue au cours de l'exécution du programme.
On peut la schématiser par une boîte qui porte une étiquette et son contenu.



B Type d'une variable

Les langages de programmation distinguent différents types de variables.
Par exemple, une variable peut être de type :
- **nombre entier** ;
- **nombre flottant**, c'est-à-dire nombre à virgule ;
- **chaîne de caractères**, sa valeur est alors une suite ordonnée de caractères ;
- **liste**, c'est-à-dire une suite ordonnée d'objets du langage.

Par exemple : L = [1, 3, 5, 7, 9] ou M = ["a", "e", "i", "o", "u", "y"].
- **booléen**, elle n'a alors que deux valeurs possibles : True (Vrai) et False (Faux).

Par exemple $a < 5$ est un booléen qui a la valeur True si a est strictement inférieur à 5 ou False sinon.
Des opérateurs et des fonctions du langage permettent de travailler avec chaque type de variables.

C L'affectation

L'instruction d'affectation permet d'attribuer une valeur à une variable.

Algorithme	Programme Python
$X \leftarrow 2$	X=2

La variable X est affectée de la valeur 2.

D Les instructions d'entrée-sortie

Les instructions d'entrée-sortie permettent de saisir en entrée et d'afficher en sortie les valeurs des variables.

Algorithme	Programme Python
Saisir X	X=float(input())
Afficher X	print(X)

Ces instructions permettent aussi d'afficher un message, par exemple :
- n=int(input("Nombre d'essais:"));
- print("Surface=",S).

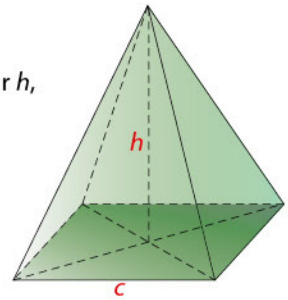
INFO

Dans le langage Python, l'instruction d'entrée précise le type de la variable : int (nombre entier), float (nombre à virgule) ou chaîne de caractères lorsque rien n'est précisé.

Algorithmique et programmation

Exemple 1

Calculer le volume d'une pyramide à base carrée



• Une pyramide a pour base un carré de côté c, en cm, et pour hauteur h, en cm.
On note S l'aire, en cm^2, de sa base et V son volume, en cm^3.

• Voici :
– un algorithme qui permet de saisir en entrée les valeurs de c et h puis qui calcule S, V et affiche le volume V ;
– son codage en langage Python.

Algorithme
Saisir c, h
$S \leftarrow c^2$
$V \leftarrow \frac{1}{3} Sh$
Afficher V

Programme Python

```python
c=float(input("c ="))
h=float(input("h ="))
S=c**2
V=1/3*S*h
print("V =",V)
```

• On exécute le programme avec $c = 3$ et $h = 4{,}5$.
On obtient dans la console `V = 13.5`.
Ainsi, le volume de la pyramide est 13,5 cm^3.

Exemple 2

Écrire une séquence d'instructions

• Un opticien propose à ses clients une réduction sur le prix des montures d'un pourcentage égal à leur âge.
Sophie a 16 ans, elle aura donc droit à une réduction de 16 % sur le prix de sa monture.

• Voici :
– un algorithme qui permet de saisir le prix initial, en euro, de la monture, l'âge du client puis qui calcule et affiche le prix réduit de la monture ;
– son codage en langage Python.

Algorithme
Saisir prix, age
reduction←prix×age/100
prix_reduit←prix−reduction
Afficher prix_reduit

Programme Python

```python
prix=float(input("Prix ="))
age=int(input("Age ="))
reduction=prix*age/100
prix_reduit=prix-reduction
print("Prix réduit =",prix_reduit)
```

Les variables et leurs valeurs sont :
• prix : le prix initial, en euro ;
• age : l'âge du client, en année ;
• reduction : la réduction appliquée, en euro ;
• prix_reduit : le prix après réduction, en euro.
Les valeurs des variables prix et age sont saisies en entrée. On calcule ensuite et on affecte aux variables reduction et prix_reduit les résultats des calculs.

• On exécute le programme avec prix = 65 et age = 25.
On obtient dans la console `Prix réduit = 48.75`.
Ainsi, après réduction, le client paie sa monture 48,75 €.

2 Notion de fonction

A Fonction en programmation

Les langages de programmation donnent la possibilité de définir des fonctions.
Une fonction réalise un traitement et renvoie un résultat, elle peut être appelée à plusieurs reprises par un programme.

Programme Python

```
def f(x,y):
    …
    return z
```

paramètres : x et y

Une fonction peut admettre aucun, un ou plusieurs paramètres.

B Paramètres

Lors d'un appel à la fonction, des valeurs particulières (arguments) sont données aux paramètres de la fonction.

C Intérêt des fonctions

Programmer des fonctions facilite l'écriture des programmes, permet de les structurer et les rend plus lisibles.

Exemple 3

Écrire une fonction qui renvoie l'aire d'un triangle

- On connaît la longueur c d'un côté d'un triangle, la hauteur h relative à ce côté et on calcule son aire.
- La fonction Triangle a pour paramètres la longueur c et la hauteur h. Elle renvoie pour résultat l'aire S du triangle.
- Voici le codage de cette fonction en langage Python.

Programme Python

```
def Triangle(c,h):
    S=c*h/2
    return S
```

- Par exemple, dans la console, on effectue un appel à la fonction Triangle lorsque c a pour valeur 7 et b a pour valeur 9. Ainsi, un triangle dont la longueur d'un côté est 7 et la hauteur relative à ce côté est 9 a une aire égale à 31,5.

```
>>> Triangle(7,9)
31.5
```

Remarque : on peut également écrire un programme qui fait appel à cette fonction.

Programme Python

```
4
5 côté=float(input("Côté ="))
6 hauteur=float(input("Hauteur ="))
7 aire=Triangle(côté,hauteur)
8 print("Aire =",aire)
```

Sur la ligne 7, le programme fait appel à la fonction Triangle avec les arguments côté et hauteur.

3 Les instructions conditionnelles

A Condition

Une **condition** est un énoncé qui peut être vrai ou faux.
Par exemple, $a = b$, $x \geqslant 0$, n est pair sont des conditions vraies ou fausses suivant les valeurs des variables a, b, x et n.

B Traitements conditionnels

Dans un programme, on peut tester une condition et, selon que celle-ci est vraie ou fausse, effectuer un traitement ou un autre.
On parle alors de **traitements conditionnels**.

Algorithme

```
Si condition alors
  |Traitement 1
  |sinon
  |Traitement 2
Fin Si
```

Programme Python

```
if X==2:
    ...
else:
    ...
```

Ici, la condition X==2 est vraie lorsque X est égal à 2 et fausse sinon.

C Une situation particulière

Dans un test, on peut ne pas effectuer de traitement dans le cas où la condition est fausse. Dans cette situation, on omet « sinon ».

Algorithme

```
Si condition alors
  |Traitement
Fin Si
```

Programme Python

```
if X==2:
  ...
```

Exemple 4

- L'algorithme suivant permet de saisir un nombre entier naturel n puis détermine et affiche si ce nombre est divisible par 11.
- L'algorithme est codé en langage Python.

n%11 renvoie pour résultat le reste de la division de n par 11

Algorithme

```
Saisir n
Si n est divisible par 11 alors
  |C ← "n est divisible par 11"
  |sinon
  |C ← "n n'est pas divisible par 11"
Fin Si
Afficher C
```

Programme Python

```python
n=int(input("n ="))
if n%11==0:
    C="est divisible par 11"
else:
    C="n'est pas divisible par 11"
print(n,C)
```

- On exécute le programme avec $n = 86\,834$.
On obtient dans la console :

```
86834 est divisible par 11
```

INFO
La condition n%11==0 peut aussi être considérée comme une variable B de type booléen. B prend la valeur True lorsque n est divisible par 11 et la valeur False sinon.

Exemple 5

Programmer une fonction définie par intervalles

- *f* est la fonction qui, à tout nombre réel x, associe le nombre réel $f(x)$ défini par :

$$\begin{cases} f(x) = 1 + \sqrt{x} & \text{si } x \geqslant 0 \\ f(x) = (x-1)^2 & \text{si } x < 0 \end{cases}$$

- Voici un algorithme qui détermine le nombre réel y image de x par la fonction *f*.
- L'algorithme est codé par une fonction F écrite en langage Python.

Algorithme

Si $x \geqslant 0$ alors
  $y \leftarrow 1 + \sqrt{x}$
sinon
  $y \leftarrow (x-1)^2$
Fin Si

Programme Python

```
from math import *

def F(x):
    if x>=0:
        y=1+sqrt(x)
    else:
        y=(1-x)**2
    return y
```

On importe le module math car on utilise la fonction sqrt (racine carrée).

- On exécute la fonction F.

Avec $x = 2$, on obtient :

```
>>> F(2)
2.414213562373095
```

Ainsi $f(2) \approx 2{,}4$.

Avec $x = -3$, on obtient :

```
>>> F(-3)
16
```

Ainsi $f(-3) = 16$.

Exemple 6

Programmer des instructions conditionnelles

- Un magasin de reprographie applique le tarif suivant :

Nombre de photocopies	Tarif à l'unité
De 1 à 50	0,15 €
À partir de 51	0,10 €

- Voici un algorithme qui détermine le prix dû pour *n* photocopies réalisées.
- L'algorithme est codé par une fonction Prix écrite en langage Python.

Algorithme

Si $n \leqslant 50$ alors
  $x \leftarrow 0{,}15x$
sinon
  $n \leftarrow 7{,}5 + 0{,}1(n-50)$
Fin Si

Programme Python

```
def Prix(n):
    if n<=50:
        x=0.15*n
    else:
        x=7.5+0.1*(n-50)
    return x
```

- On exécute la fonction Prix.

Avec $n = 45$, on obtient :

```
>>> Prix(45)
6.75
```

Ainsi le prix dû pour 45 photocopies est 6,75 €.

Avec $n = 72$, on obtient :

```
>>> Prix(72)
9.7
```

Ainsi le prix dû pour 72 photocopies est de 9,70 €.

INFO

Dans le langage Python, l'indentation, c'est-à-dire le décalage vers la droite, permet de repérer les instructions qui font partie d'un traitement conditionnel.

4 Boucle bornée

A Répéter *n* fois

Une boucle permet de répéter plusieurs fois de suite un même traitement.
Lorsque le nombre *n* d'itérations est connu à l'avance, on parle de **boucle bornée**.

B Compteur

Afin de compter le nombre d'itérations, on utilise un compteur initialisé par exemple à 1 et qui s'incrémente automatiquement de 1 à chaque itération jusqu'à la valeur n.

Algorithme	Programme Python
Pour *i* allant de 1 à *n* │Traitement Fin Pour	`for i in range (1, n+1):` `    …`

Par exemple, dans la boucle
`for i in range (1,6):`
…
le compteur i prend successivement les valeurs 1, 2, 3, 4, 5.

Exemple 7

Étudier l'évolution d'une population

• Deux chercheurs étudient l'évolution d'une population de truites dans un lac de montagne.
Ils constatent une diminution annuelle moyenne de 20 % de cette population.
Afin de sauvegarder l'espèce, ils décident d'introduire chaque année 50 nouvelles truites.
On estime la population initiale du lac à 500 truites.

• Voici :
– un algorithme qui permet de saisir un nombre d'années n, $n \geqslant 1$, puis qui calcule et affiche le nombre de truites au bout de n années ;
– son codage en langage Python.

Algorithme	Programme Python

Saisir n
$X \leftarrow 500$
Pour i allant de 1 à n
│ $X \leftarrow 0{,}8X + 50$
Fin Pour
Afficher X
	`n=int(input("n ="))`
	`X=500`
	`for i in range(1,n+1):`
	`    X=0.8*X+50`
	`print("X =",X)`

• On exécute le programme.

Avec $n = 12$,
on obtient : `X = 267.17986918400004`
Ainsi, après 12 années,
le lac compte environ 267 truites.

Avec $n = 25$,
on obtient : `X = 250.94447329657396`
Après 25 années,
le lac compte environ 251 truites.

Lorsqu'on donne à n des valeurs de plus en plus grandes, on remarque que le nombre de truites de ce lac tend à se stabiliser à 250.

INFO
Dans le langage Python, la fin de l'indentation marque la fin du traitement effectué dans la boucle.

Algorithmique et programmation

Exemple 8

Calculer une somme à l'aide d'une boucle

• L'algorithme suivant calcule la somme S des carrés des nombres entiers de 1 à n, avec n nombre entier naturel, $n \geqslant 1$.

Pour tout nombre entier naturel $n \geqslant 1$, on pose $S = 1^2 + 2^2 + \ldots + n^2$.

• L'algorithme est codé par une fonction Somme écrite en langage Python.

Algorithme

$S \leftarrow 0$
Pour i allant de 1 à n
  $S \leftarrow S + i^2$
Fin Pour

Programme Python

```python
def Somme(n):
    S=0
    for i in range(1,n+1):
        S=S+i**2
    return S
```

• On exécute le programme avec $n = 5$, voici le tableau de suivi des variables :

i		1	2	3	4	5
S	0	1	5	14	30	55

On obtient dans la console :

```
>>> Somme(5)
55
```

Ainsi $1^2 + 2^2 + 3^2 + 4^2 + 5^2 = 55$.

Exemple 9

Compter le nombre d'occurrences d'un caractère

• Il s'agit de compter dans une phrase le nombre de fois où l'on rencontre un caractère donné.

• Voici :

– un algorithme qui détermine le nombre d'occurrences du caractère dans la phrase ;

– son codage par une fonction Nbr écrite en langage Python.

Algorithme

$l \leftarrow$ longueur(phrase)
$n \leftarrow 0$
Pour k allant de 0 à $l - 1$
  Si phrase[k] = car alors
    $n \leftarrow n + 1$
  Fin Si
Fin Pour

Programme Python

```python
def Nbr(phrase,car):
    l=len(phrase)
    n=0
    for k in range(0,l):
        if phrase[k]==car:
            n=n+1
    return n
```

La variable n compte le nombre d'apparitions du caractère car dans phrase.

• On exécute la fonction avec phrase="Je programme en Python" et car="e".

Voici le tableau de suivi des variables :

k		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
phrase[*k*]		J	e		p	r	o	g	r	a	m	m	e		e	n		P	y	t	h	o	n
n	0	0	1	1	1	1	1	1	1	1	1	1	2	2	3	3	3	3	3	3	3	3	3

On obtient dans la console :

```
>>> Nbr("Je programme en Python","e")
3
```

Ainsi, la lettre e est présente 3 fois dans la phrase saisie.

INFO

La variable phrase est du type chaîne de caractères.

La fonction len renvoie le nombre de caractères l de phrase.

Les caractères sont repérés par phrase[0], phrase[1], …, phrase [l–1].

5 Boucle non bornée

A Répéter tant que …

Dans une boucle, le nombre d'itérations peut dépendre d'une condition.
Le traitement est alors répété tant que la condition est vraie.
Lorsque la condition est fausse, on sort de la boucle et la suite des instructions du programme s'applique.

B Nombre d'itérations

Le nombre d'itérations n'est donc en général pas prévu à l'avance et on parle de **boucle non bornée**.

Algorithme

```
Tant que condition
  | Traitement
Fin Tant que
```

Programme Python

```
while X<2:
    ...
```

Ici, tant que la condition X<2 est vraie, on exécute les instructions de la boucle.

Exemple 10

Simuler une expérience aléatoire

• Au jeu des petits chevaux, pour sortir de l'écurie, on effectue des lancers successifs d'un dé équilibré jusqu'à ce que l'on obtienne 6.

• Voici :
– un algorithme qui effectue une simulation de cette expérience aléatoire et donne le nombre n de lancers effectués ;
– son codage par une fonction L écrite en langage Python.

Algorithme

$n \leftarrow 1$
$x \leftarrow$ nombre entier aléatoire de 1 à 6
Tant que $x \neq 6$
  | $n \leftarrow n+1$
  | $x \leftarrow$ nombre entier aléatoire de 1 à 6
Fin Tant que

Programme Python

```
from random import *

def L():
    n=1
    x=randint(1,6)
    while x!=6:
        n=n+1
        x=randint(1,6)
    return n
```

On importe le module random car on utilise la fonction randint

La condition « x différent de 6 » est notée : x!=6

• On exécute la fonction L, on obtient dans la console :

```
>>> L()
4
```

Cela signifie que pour cette expérience, on a lancé le dé 4 fois pour sortir le cheval de l'écurie.

INFO
Dans le langage Python, randint(1,6) renvoie un nombre entier aléatoire compris entre 1 et 6.
random() renvoie un nombre aléatoire de l'intervalle $[0\ ;\ 1[$.

Remarque :
Pour écrire ce programme, on peut aussi utiliser la fonction random.
En effet, pour x=random(), la probabilité de l'événement « x<1/6 » est 1/6 et celle de l'événement « x>=1/6 » est 5/6.

```
from random import *

def L():
    n=1
    x=random()
    while x>=1/6:
        n=n+1
        x=random()
    return n
```

INFO
Dans le langage Python, la fin de l'indentation marque la fin des instructions de la boucle.

Exemple 11

Déterminer un nombre d'années

• Louis a placé une somme de 1 500 € sur son livret d'épargne à un taux de 1,5 % par an.

• L'algorithme suivant détermine le nombre d'années N au bout duquel Louis disposera d'une somme supérieure à 1 600 €.

• L'algorithme est codé par une fonction A écrite en langage Python.

Algorithme	Programme Python

$S \leftarrow 1500$
$N \leftarrow 0$
Tant que $S \leqslant 1600$
  $S \leftarrow 1{,}015 \times S$
  $N \leftarrow N+1$
Fin Tant que
	`def A():`
	`    S=1500`
	`    N=0`
	`    while S<=1600:`
	`        S=1.015*S`
	`        N=N+1`
	`    return N`

Chaque année, le montant S du livret de Louis augmente de 1,5 %, il est donc multiplié par 1,015.

• On exécute la fonction A, on obtient dans la console :

```
>>> A()
5
```

Ainsi, au bout de 5 ans, Louis disposera d'une somme supérieure à 1 600 €.

Exemple 12

Tracer des points aléatoires dans un repère

• Une puce effectue des sauts aléatoires. Au départ, elle se situe à l'origine d'un repère du plan.

• Voici :

– un algorithme qui construit les points de coordonnées $(x\,;y)$ qui correspondent aux positions successives de la puce.

Les sauts ont lieu tant que $x \leqslant 10$ et $y \leqslant 10$.

– son codage en langage Python.

Algorithme	Programme Python

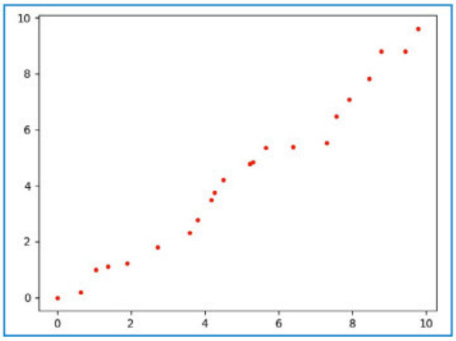
$x \leftarrow 0$
$y \leftarrow 0$
Tant que $x \leqslant 10$ et $y \leqslant 10$
  Tracer le point de coordonnées $(x\,;y)$
  $x \leftarrow x +$ nombre aléatoire de $[0\,;1[$
  $y \leftarrow y +$ nombre aléatoire de $[0\,;1[$
Fin Tant que
	`from pylab import *`
	`x=0`
	`y=0`
	`while x<=10 and y<=10:`
	`    plot(x,y,'r.')`
	`    x=x+random()`
	`    y=y+random()`
	`show()`

La condition x<=10 and y<=10 est vraie lorsque x<=10 et y<=10 sont vraies toutes les deux.

INFO

x<=10, y<=10 sont des variables booléennes.
Les opérateurs logiques and (et), or (ou) et not (non) s'appliquent aux variables de ce type.

• On exécute le programme, voici ci-contre un affichage des positions successives de la puce.



INFO

Dans le langage Python, on importe le module pylab pour réaliser des constructions graphiques. L'instruction plot(x,y,'r.') trace le point de coordonnées $(x\,;y)$ en rouge.
L'instruction show() affiche la figure à l'écran.

6 Le type liste

Exemple 13

Travailler avec des listes

• Une roue de loterie est divisée en dix secteurs identiques numérotés de 1 à 10. Une expérience aléatoire consiste à faire tourner la roue et à noter le numéro obtenu.

• Voici :

– un algorithme qui réalise n simulations de cette expérience aléatoire et donne pour résultat la liste des n numéros obtenus ;

– son codage par la fonction Loterie écrite en langage Python.

Algorithme

$L \leftarrow []$
Pour i allant de 0 à $n-1$
  $a \leftarrow$ nombre entier aléatoire de 1 à 10
  $L \leftarrow L + [a]$
Fin Pour

Programme Python

```
from random import *

def Loterie(n):
    L=[]
    for i in range(0,n):
        a=randint(1,10)
        L=L+[a]
    return L
```

L'instruction L=[] initialise la liste L à la liste vide.

• Par exemple dans la console, on obtient :

```
>>> Loterie(5)
[4, 8, 2, 10, 9]
```

Ainsi on a simulé 5 fois l'expérience aléatoire et obtenu les numéros 4, 8, 2, 10 et 9.

• On souhaite également obtenir la somme des n numéros obtenus.

• Pour cela, on écrit une fonction Somme qui calcule et renvoie la somme des éléments d'une liste.

• Voici son algorithme et le codage en langage Python à la suite du précédent.

Algorithme

$n \leftarrow$ Longueur (L)
$S \leftarrow 0$
Pour i allant de 0 à $n-1$
  $S \leftarrow S + L[i]$
Fin Pour

Programme Python

```
from random import *

def Loterie(n):
    L=[]
    for i in range(0,n):
        a=randint(1,10)
        L=L+[a]
    return L

def Somme(L):
    n=len(L)
    S=0
    for i in range(0,n):
        S=S+L[i]
    return S
```

• Par exemple, dans la console, on obtient :

```
>>> Somme(Loterie(10))
55
```

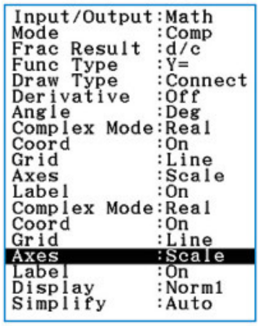
Ainsi on a réalisé 10 simulations de l'expérience et obtenu pour somme 55.

INFO

Une liste L de nombres est une suite ordonnée de nombres, on note par exemple L=[1, 7, 3, 5].
La fonction len renvoie le nombre n d'éléments d'une liste L.
Ces éléments sont notés L[0], L[1], …, L[n-1].
Pour ajouter un élément x à la fin d'une liste L, on écrit en langage Python : L=L+[x] ou L.append(x).

1 Calculatrice Casio Graph 90+E

A Réglage des paramètres



MENU Exe-Mat SHIFT MENU (SET UP)

On peut régler en particulier l'unité de mesure des angles (Angle) et le format d'affichage des nombres (Display).

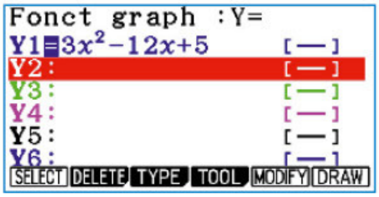
Le réglage d/c pour Frac Result permet un affichage normal des fractions.

B Fonctions

f est la fonction définie sur $\mathbb{R}$ par $f(x) = 3x^2 - 12x + 5$.

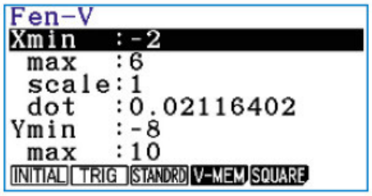
① Saisir la fonction

MENU Graphe Saisir l'expression de $f(x)$ à l'aide de la touche X,θ,T puis EXE



② Régler la fenêtre

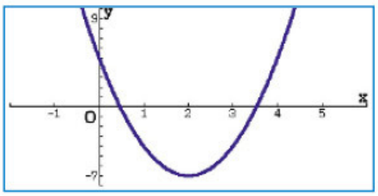
SHIFT F3 (V-Window) Saisir les valeurs



EXIT

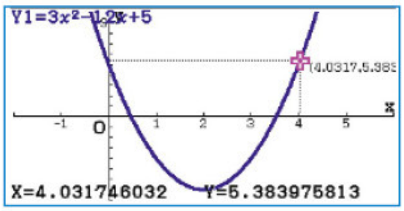
③ Tracer la courbe

F6 (DRAW)



④ Lire les coordonnées de points de la courbe

SHIFT F1 (Trace) ▶ et ◀

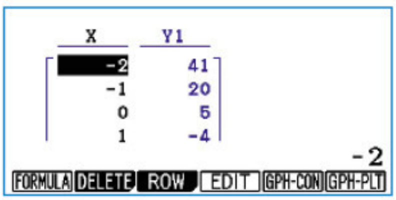


⑤ Tabuler la fonction

MENU Table F5 (SET)

On règle le début, la fin et le pas de la table.

EXIT F6 (TABLE)



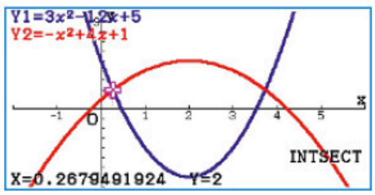
⑥ Étudier l'intersection de deux courbes

g est la fonction définie sur $\mathbb{R}$ par :

$$g(x) = -x^2 + 4x + 1$$

On trace les courbes représentatives de *f* et *g* dans un même repère.

SHIFT F5 (G-SOLVE) F5 (INTSECT)



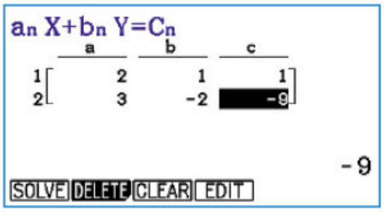
Pour passer d'un point à l'autre : ▶ ◀

C Résoudre un système d'équations

① **Saisir le système** $\begin{cases} 2x + y = 1 \\ 3x - 2y = -9 \end{cases}$

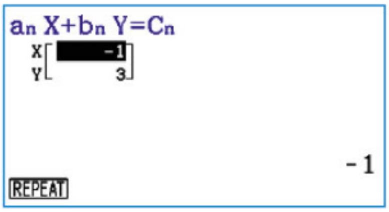
MENU Équation F1 |SIMUL| F1 (2)

Saisir les coefficients du système



② **Résoudre le système**

F1 (SOLVE)



D Statistiques

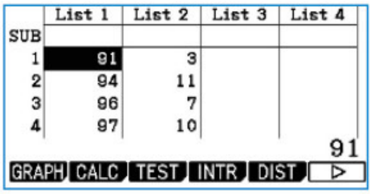
Voici une série statistique.

Valeur	91	94	96	97	101	104
Effectif	3	11	7	10	9	10

① **Saisir la série statistique**

MENU Statistique Si besoin, vider les listes : F6 (▷) F4 (DEL-ALL)

Saisir les valeurs dans List 1 et les effectifs dans List 2



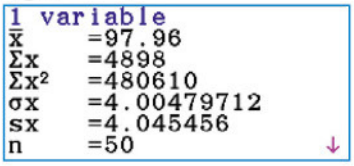
② **Définir les listes**

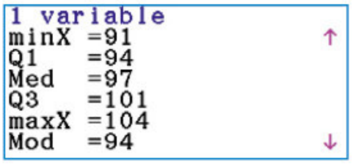
F2 (CALC) F6 (SET)

Renseigner les deux premières lignes comme sur l'écran ci-dessous, puis EXIT EXIT pour revenir à la saisie.

③ **Afficher les paramètres statistiques**

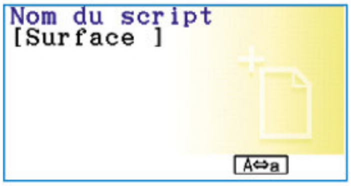
F2 (CALC) F1 (1-VAR)





Remarque : l'écart-type de la série est noté σx.

E Écrire un programme en langage Python

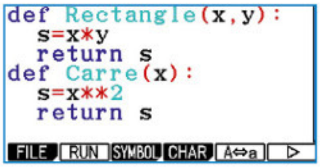


① **Donner le nom du script**

MENU Python F3 (NEW)

Saisir le nom du script EXE

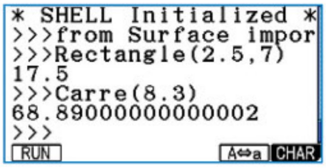
② **Saisir le programme dans l'éditeur**



③ **Exécuter le programme dans la console**

F2 (RUN) Enregistrer le fichier

Saisir l'appel à une fonction



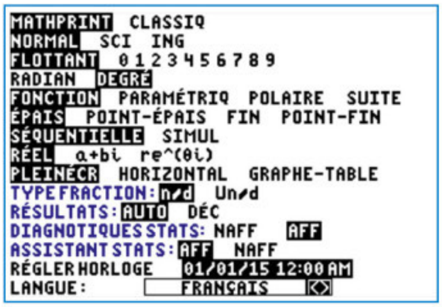
2 Calculatrice TI-83 Premium CE

A Réglage des paramètres

mode

On peut régler en particulier l'unité de mesure des angles (RADIAN DEGRÉ) et le format d'affichage des nombres (NORMAL SCI FLOTTANT…)

Le réglage n/d pour TYPE FRACTION permet un affichage normal des fractions.

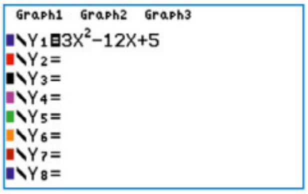


B Fonctions

f est la fonction définie sur $\mathbb{R}$ par $f(x) = 3x^2 - 12x + 5$.

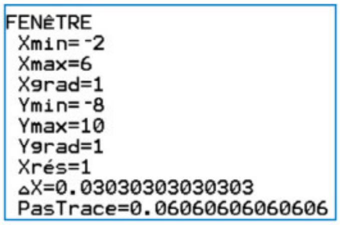
① Saisir la fonction

f(x) Saisir l'expression de $f(x)$ en utilisant la touche X,t,θ,n puis entrer



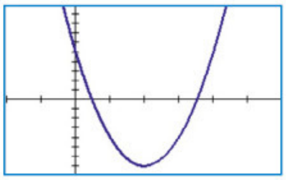
② Régler la fenêtre

fenêtre Saisir les valeurs.



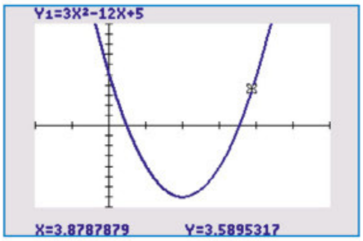
③ Tracer la courbe

graphe



④ Lire les coordonnées de points de la courbe

trace ► et ◄



⑤ Tabuler la fonction

2nde fenêtre (déf table f2)

On règle le début et le pas de la table.

2nde graphe (table f5)

X	Y1
-2	41
-1	20
0	5
1	-4
2	-7
3	-4
4	5
5	20
6	41
7	68
8	101

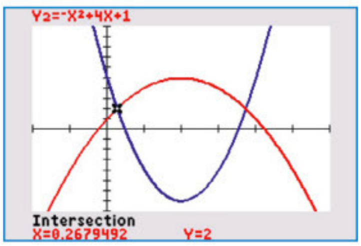
⑥ Étudier l'intersection de deux courbes

g est la fonction défnie sur $\mathbb{R}$ par :

$$g(x) = -x^2 + 4x + 1$$

On trace les courbes représentatives de f et g dans un même repère.

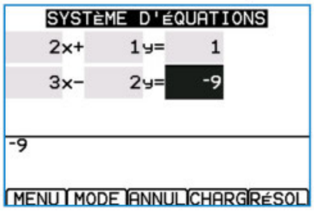
2nde trace (calcul f4) (5:intersection), on renseigne la 1re fonction, la 2e fonction et on choisit une valeur initiale pour obtenir un point d'intersection.



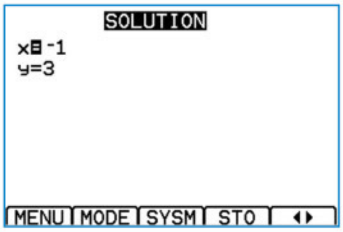
C Résoudre un système d'équations

① **Saisir le système** $\begin{cases} 2x + y = 1 \\ 3x - 2y = 9 \end{cases}$

résol (2 : PlySmlt2) (2 : SOLVEUR SYST D'ÉQUATIONS)
Choisir le nombre d'équations (2) et le nombre d'inconnues (2)
graphe (SUIV.) et saisir les coefficients du système



② **Résoudre le système**
graphe (RÉSOL)

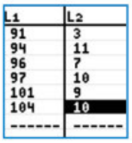


D Statistiques

Voici une série statistique.

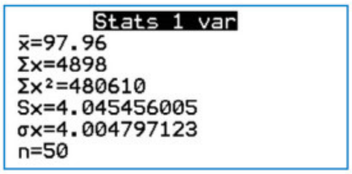
Valeur	91	94	96	97	101	104
Effectif	3	11	7	10	9	10

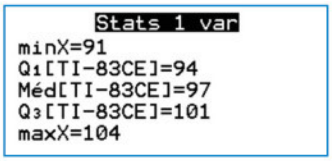
① **Saisir la série statistique**
stats (EDIT) (1 : Modifier) Si besoin, vider les listes en allant sur leur nom puis annul entrer.
Saisir les valeurs dans L_1 et les effectifs dans L_2.



② **Définir les listes**
stats ▷ (CALC) (1 : Stats 1 Var)
Renseigner les deux premières lignes comme sur l'écran ci-dessous en utilisant :
2nde stats (listes)

③ **Afficher les paramètres statistiques**
▼ (Calculer) entrer



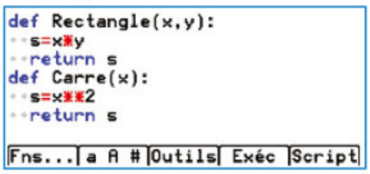


Remarque : l'écart-type de la série est noté σx.

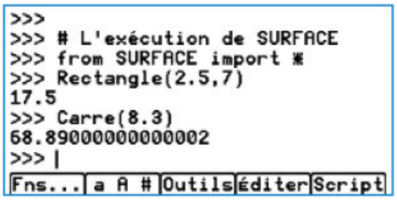
E Écrire un programme en langage Python

① **Donner le nom du script**
2nde resol (apps) ▼ (PyAdaptr) entrer zoom (f3) (Nouv)
Saisir le nom du script graphe (ok)

② **Saisir le programme dans l'éditeur**



③ **Exécuter le programme dans la console**
trace (f4) (Exéc)



3 Calculatrice NumWorks

A Réglage des paramètres

Parametres OK

On peut régler en particulier l'unité de mesure des angles (Unite d'angle) et le format d'affichage des nombres (Format resultat).

Dans Format ecriture, on peut choisir : Naturelle $1+\frac{2}{3}$ ou En ligne 1 + 2/3.

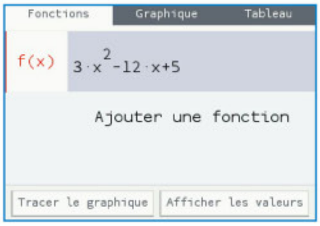
B Fonctions

f est la fonction définie sur $\mathbb{R}$ par $f(x) = 3x^2 - 12x + 5$.

① Saisir la fonction

Fonctions OK (Ajouter une fonction) OK

Saisir l'expression de $f(x)$ en utilisant la touche x,n,t puis OK

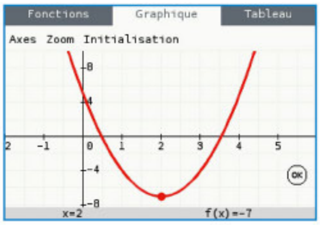


② Régler la fenêtre

Graphiques OK Axes OK Saisir les valeurs

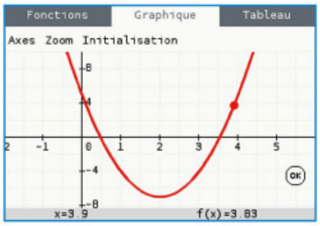
Valider OK

③ Tracer la courbe



④ Lire les coordonnées de points de la courbe

▶ et ◀

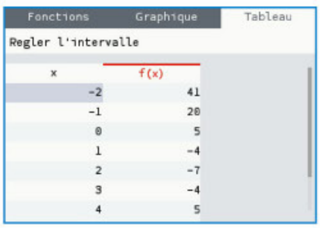


⑤ Tabuler la fonction

Tableau OK Régler l'intervalle OK

On règle le début, la fin et le pas de la table

Valider OK



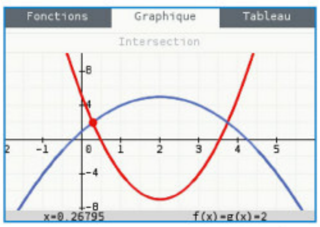
⑥ Étudier l'intersection de deux courbes

g est la fonction définie sur $\mathbb{R}$ par :

$$g(x) = -x^2 + 4x + 1.$$

On trace les courbes représentatives de f et g dans un même repère.

OK Calculer OK Intersection OK



Pour passer d'un point à l'autre ▶ et ◀

C Résoudre un système d'équations

① **Saisir le système** $\begin{cases} 2x + y = 1 \\ 3x - 2y = 9 \end{cases}$

OK | Ajouter une équation | OK

Choisir le modèle, saisir l'équation et faire de même pour une 2^e^ équation

```
{ 2·x+y=1
{ 3·x-2·y=-9
Ajouter une equation
Resoudre le systeme
```

② **Résoudre le système**

Résoudre le système | OK

Solution

x	-1
y	3

D Statistiques

Voici une série statistique.

Valeur	91	94	96	97	101	104
Effectif	3	11	7	10	9	10

① **Saisir la série statistique**

Statistiques | OK | Donnees | OK Si besoin vider les listes Valeurs V1 | OK | Effacer la colonne | OK

Saisir les valeurs V1 et les effectifs N1

Donnees | Histogramme | Boite | Stats

Valeurs V1	Effectifs N1	Valeurs
91	3	
94	11	
96	7	
97	10	
101	9	
104	10	

② **Afficher les paramètres**

Stats | OK

Donnees | Histogramme | Boite | Stats

	V1/N1
Effectif total	50
Minimum	91
Maximum	104
Etendue	13
Moyenne	97.96
Ecart type	4.004797
Variance	16.0384
Premier quartile	94
Troisieme quartile	101
Mediane	97
Ecart interquartile	7
Somme	4898
Somme des carres	480610
Ecart type echantillon	4.045456

E Écrire un programme en langage Python

① **Donner le nom du script**

Python | OK | Ajouter un script | OK

Saisir le nom du script | OK

```
mandelbrot.py ...
polynomial.py ...
surface.py ...
Console d'execution
```

② **Saisir le programme dans l'éditeur**

surface.py | OK

```python
from math import *
def Rectangle(x,y):
  s=x*y
  return s
def Carre(x):
  s=x**2
  return s
```

③ **Exécuter le programme dans la console**

OK | surface.py | ... | OK

Executer le script | OK

```
>>> from surface import *
>>> Rectangle(2.5,7)
17.5
>>> Carre(8.3)
68.89000000000002
```

4 GeoGebra

A Géométrie plane

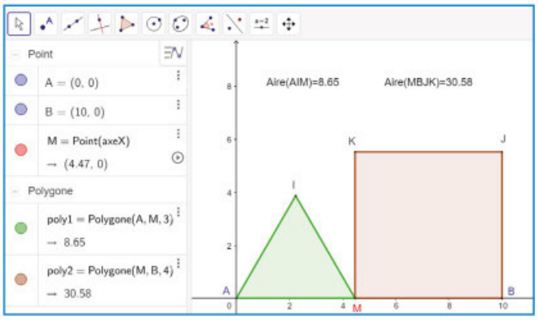
Exemple 1

Comparer des aires et conjecturer

Dans un repère, on donne les points A(0 ; 0) et B(10 ; 0). M est un point variable du segment [AB]. On construit le triangle équilatéral AIM et le carré MBJK.

On définit A et B sur la ligne Saisie de la fenêtre Algèbre
Saisie: **A=(0,0)**
Saisie: **B=(10,0)**

On définit le point M sur [AB].
Point sur Objet



On construit le triangle équilatéral AMI et le carré MBJK.
Polygone régulier

On déplace le point M et on **conjecture** sa position sur [AB] telle que les aires de AIM et MBJK soient égales.

Dans la fenêtre Algèbre, on lit l'aire de poly1=AMI et poly2=MBJK.
On peut également afficher ces aires dans la fenêtre Graphique.

Exemple 2

Construire une figure et conjecturer

ABC est un triangle, I est le milieu du côté [BC].

k est un nombre réel de l'intervalle [0 ; 1] et M le point défini par $\overrightarrow{AM} = k\overrightarrow{AB}$.

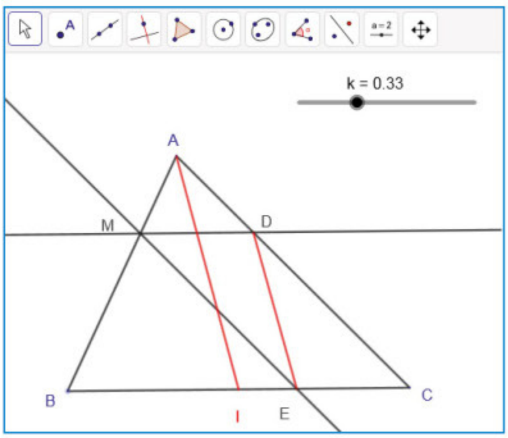
On trace par le point M :

– la parallèle à (BC), elle coupe le côté [AC] en D ;

– la parallèle à (AC), elle coupe le côté [BC] en E.

On définit le curseur k et ses propriétés

On définit M sur la ligne de saisie.
Saisie: **M=A+k*Vecteur(A,B)**



On construit le milieu I de [BC].
Milieu ou centre

On trace les parallèles et leur intersection avec les côtés.
Parallèle
Intersection

On déplace le curseur k afin de **conjecturer** la position du point M sur [AB] telle que les droites (AI) et (DE) soient parallèles.

B Courbe représentative d'une fonction

Exemple 3

Étudier une intersection

f est la fonction définie sur $\mathbb{R}$ par $f(x) = -x^3 + 2x^2 + 1$.

Dans un repère, on trace la courbe représentative $\mathscr{C}$ de la fonction f et on étudie l'intersection de cette courbe avec la droite d d'équation $y = 2$.

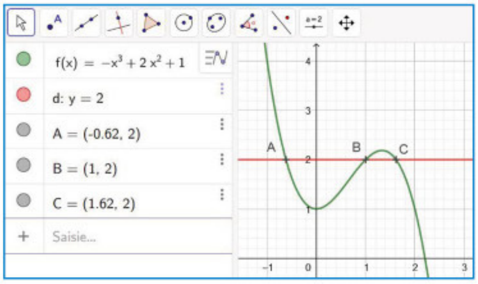
On entre l'expression de f et l'équation de d sur la ligne de saisie de la fenêtre Algèbre.

Saisie: f(x)=-x^3+2*x^2+1

Saisie: y=2

On peut également tracer la courbe sur un intervalle, par exemple :

Saisie: f(x)=Fonction(-x^3+2*x^2+1, 0, +∞)



On crée les points d'intersection de $\mathscr{C}$ et d.

Intersection

On lit les coordonnées des points d'intersection dans la fenêtre Algèbre.

On peut également faire apparaître un fond quadrillé

Graphique
Axes
Grille

C Calcul formel

On ouvre la fenêtre de Calcul formel [Calcul formel]

Exemple 4

Développer, factoriser

- Développer : $(3x-2)^2 + (x+6)^2$
- Factoriser : $4x^2 + 8x - 12$

On peut également saisir l'expression puis utiliser les touches rapides du menu [(( ))] et [15 / 3·5]

1	Développer$\left((3x-2)^2 + (x+6)^2\right)$
	→ $10\,x^2 + 40$
2	Factoriser$(4x^2 + 8x - 12)$
	→ $4\,(x-1)\,(x+3)$

Exemple 5

Résoudre une équation, une inéquation

- Résoudre l'équation : $\dfrac{2x-1}{3} = \dfrac{3x+5}{4}$.
- Résoudre l'inéquation : $3x + 7 > 0$.

On peut également saisir l'équation ou l'inéquation puis utiliser la touche rapide du menu [x=]

1	Résoudre$\left(\dfrac{2\,x-1}{3} = \dfrac{3\,x+5}{4}\right)$
	→ $\{x = -19\}$
2	Résoudre$(3\,x + 7 > 0)$
	→ $\left\{x > -\dfrac{7}{3}\right\}$

Exemple 6

Utiliser les nombres premiers en arithmétique

- Tester si le nombre 197 est premier ou non.
- Déterminer la décomposition en produit de facteurs premiers de 8 076.

1	EstPremier(197)
	→ true
2	FacteursPremiers(8076)
	→ {2, 2, 3, 673}

5 Tableur

A Affichage des nombres

Exemple 1

On calcule l'aire d'un disque de rayon 10.
Le résultat est affiché sous différents formats.

On saisit : =PI()*10^2

	A	
1	314,1592654	format standard
2	314,16	format avec 2 décimales
3	3,14E+02	format scientifique avec 2 décimales

Exemple 2

La classe compte 17 filles et 15 garçons.
On calcule la proportion de filles dans la classe.

On saisit : =17/32

	A	
1	0,53125	format standard
2	53,13%	format pourcentage avec 2 décimales

Dans les différents formats, on peut fixer le nombre de décimales avec lequel les nombres sont affichés.

B Des fonctions en arithmétique

- ABS(x) renvoie la valeur absolue du nombre x.
- ENT(x) renvoie la partie entière du nombre x.
- MOD(a;b) renvoie le reste de la division euclidienne de a par b avec $a \in \mathbb{N}$ et $b \in \mathbb{N}^*$.

	A	
1	7	=ABS(-7)
2	3	=ENT(3,92)
3	2	=MOD(128;3)

C Une feuille de calcul en statistique

Exemple 3

Une urne contient 6 billes jaunes et 4 billes bleues. On tire au hasard et avec remise 2 billes de l'urne et on compte le nombre de billes jaunes obtenues.
On simule un échantillon de taille 100 de l'expérience et on détermine la fréquence de chaque issue.

A1 — Expériences

Accès aux fonctions

	A	B	C	D	E	F	G
1	Expériences	1er tirage	2e tirage	Total		Nombre de jaunes	Fréquence
2	1	1	0	1		0	0,19
3	2	1	1	2		1	0,55
4	3	1	1	2		2	0,26
5	4	0	1	1		Total	1
6	5	1	1	2			

=A2+1

=SI(ALEA.ENTRE.BORNES(1;10)⩽6;1;0)

=SOMME(B2:C2)

=NB.SI(D$2:D$101;F2)/100

=SOMME(G2:G4)

ALEA.ENTRE.BORNES(1;10) renvoie un nombre entier aléatoire compris entre 1 et 10.

- NB.SI(D$2:D$101;F2) compte le nombre de fois où la valeur de la cellule F2 est présente dans le domaine D$2:D$101.
- Dans D$2, le signe $ fixe la ligne, ainsi elle n'est pas modifiée lors de la recopie vers le bas.

Une **proposition** mathématique est un énoncé qui est soit **vrai**, soit **faux**.

Exemples

- « 1 025 est un nombre impair » est une proposition vraie.
- « 275 est divisible par 3 » est une proposition fausse.
- « $x^2 \leqslant 16$ » est une proposition vraie pour certaines valeurs du nombre réel x et fausse pour les autres valeurs

1 Les connecteurs logiques « et », « ou »

A Conjonction

La **conjonction** de deux propositions P et Q est la proposition, notée **P et Q**, qui est vraie uniquement lorsque les propositions P et Q sont toutes les deux vraies.

Exemples

- a désigne un nombre réel.
« $a^2 \geqslant 0$ et $|a| \geqslant 0$ » est une proposition vraie.
- « $5 < 2^3$ et $2^3 < 7$ » est une proposition fausse. En effet, la proposition « $2^3 < 7$ » est fausse.

B Disjonction

La **disjonction** de deux propositions P et Q est la proposition, notée **P ou Q**, qui est :
– vraie lorsque l'**une au moins** des propositions P et Q est vraie ;
– fausse lorsque les propositions P et Q sont toutes les deux fausses.

Exemples

- « 65 est divisible par 2 ou 262 est divisible par 2 » est une proposition vraie.
En effet, la proposition « 262 est divisible par 2 » est vraie.
- « 25 est premier ou 49 est premier » est une proposition fausse. En effet, chacune des propositions « 25 est premier », « 49 est premier » est fausse.

2 La négation « non »

La **négation** d'une proposition P est la proposition, notée **non P**, qui est vraie lorsque P est fausse et fausse lorsque P est vraie.

Exemples

- a désigne un nombre réel.
La négation de la proposition « $a < 5$ » est la proposition « $a \geqslant 5$ ».
- x désigne un nombre réel.
La négation de la proposition « $x \neq 13$ » est la proposition « $x = 13$ ».

3 L'implication « Si ..., alors ... »

A Implication

• Une **implication** est une proposition de la forme « **Si P, alors Q** » où P est une proposition appelée **hypothèse** et Q une proposition appelée **conclusion**.
• On suppose la proposition P vraie, alors :
– si la proposition Q est vraie, l'implication « Si P, alors Q » est vraie ;
– si la proposition Q est fausse, l'implication « Si P, alors Q » est fausse.

Exemple

x désigne un nombre réel.
« Si $x = 3$, alors $x^2 = 9$ » est une implication vraie.
« Si $x = -13$ alors $x^2 < 0$ » est une implication fausse.

B Réciproque

La **réciproque** de l'implication « Si P, alors Q » est l'implication « **Si Q, alors P** ».

Exemple

x désigne un nombre réel.
La réciproque de l'implication « Si $x = 3$, alors $x^2 = 9$ » est l'implication :
« Si $x^2 = 9$, alors $x = 3$ ».
Cette réciproque est fausse pour $x = -3$, en effet $(-3)^2 = 9$ et $-3 \neq 3$.

C Contraposée

• La **contraposée** de l'implication « Si P, alors Q » est l'implication « **Si non Q, alors non P** ».
• Une implication et sa contraposée sont soit toutes les deux vraies, soit toutes les deux fausses.

Exemple

x désigne un nombre réel.
La contraposée de « Si $x = 3$, alors $x^2 = 9$ » est l'implication « Si $x^2 \neq 9$, alors $x \neq 3$ » et cette implication est vraie.
En effet, si on suppose $x^2 \neq 9$ alors x ne peut pas être égal à 3.

4 L'équivalence « ... si, et seulement si ... »

Une **équivalence** est la conjonction de deux implications réciproques :
« Si P, alors Q » et « Si Q, alors P ».
On la note « **P si, et seulement si, Q** » ou « **P équivaut à Q** ».

Exemple

ABCD est un quadrilatère.
« ABCD est un parallélogramme si, et seulement si, ses diagonales se coupent en leur milieu » est une proposition vraie.

5 Condition nécessaire suffisante

A Lorsqu'une implication « Si P, alors Q » est vraie, on dit que :

- Q est une **condition nécessaire à P**. En effet, il faut que la proposition Q soit vraie pour que la proposition P soit vraie.
- P est une **condition suffisante à Q**. En effet, il faut que la proposition P soit vraie pour que la proposition Q soit vraie.

Exemple

ABCD est un quadrilatère. « Si ABCD est un losange, alors ABCD est un parallélogramme » est une implication vraie.
« ABCD est un parallélogramme » est une condition nécessaire à « ABCD est un losange » et « ABCD est un losange » est une condition suffisante à « ABCD est un parallélogramme ».

B Lorsqu'une équivalence « P si, et seulement si, Q » est vraie, P est une **condition nécessaire et suffisante** à Q.

Exemple

x et y désignent deux nombres réels.
L'équivalence « Le produit xy est nul si, et seulement si, l'un des facteurs x ou y est nul » est vraie. La proposition « Le produit xy est nul » est une condition nécessaire et suffisante à la proposition « L'un des facteurs x ou y est nul ».

6 Les quantificateurs

A Le quantificateur « Pour tout », « Quel que soit »

- L'expression « Pour tout » est appelée **quantificateur universel**.
- On l'emploie pour exprimer que tous les éléments d'un ensemble vérifient une certaine propriété. On dit aussi « Quel que soit ».

Exemple

« Pour tout nombre réel x, $x^2 \geqslant 0$ » est une proposition vraie.

B Le quantificateur existentiel « Il existe »

- L'expression « Il existe » est appelée **quantificateur existentiel**.
- On l'emploie pour exprimer qu'au moins un élément d'un ensemble vérifie une certaine propriété.

Exemple

« Il existe un nombre entier naturel non nul n tel que $\frac{1}{n}-\frac{1}{n+1}<0{,}01$ » est une proposition vraie.

En effet, pour $n=10$, $\frac{1}{10}-\frac{1}{11}=\frac{1}{110}$. Or, $\frac{1}{110}\approx 0{,}0091$ donc $\frac{1}{10}-\frac{1}{11}<0{,}01$.

7 Raisonnements

A Démonstration par disjonction des cas

Pour démontrer qu'une proposition est vraie pour tout x d'un ensemble E, on peut démontrer qu'elle est vraie pour tout x de A (partie de E) puis qu'elle est vraie pour tout x de E n'appartenant pas à A.

Énoncé : Démontrer que pour tout nombre n de $\mathbb{Z}$, $n(n+3)$ est pair.

Solution

1^er^ cas : n est pair, c'est-à-dire $n = 2k$ avec $k \in \mathbb{Z}$.
Alors $n(n+3) = 2k(2k+3) = 2\mathrm{K}$ avec $\mathrm{K} = k(2k+3)$. Or, $\mathrm{K} \in \mathbb{Z}$ donc $n(n+3)$ est pair.
2^e^ cas : n est impair, c'est-à-dire $n = 2k+1$ avec $k \in \mathbb{Z}$.
Alors $n(n+3) = (2k+1)(2k+4) = 2(2k+1)(k+2) = 2\mathrm{K}'$
avec $\mathrm{K}' = (2k+1)(k+2)$. Or, $\mathrm{K}' \in \mathbb{Z}$ donc $n(n+3)$ est pair.
Finalement, pour tout nombre n de $\mathbb{Z}$, $n(n+3)$ est pair.

B Démonstration par contraposée

Une implication et sa contraposée sont soit toutes les deux vraies, soit toutes les deux fausses. Pour démontrer « Si P, alors Q », on peut démontrer « Si non Q, alors non P ». Ainsi, on suppose non Q vraie et on prouve non P vraie.

Énoncé : Démontrer que si le carré d'un nombre n de $\mathbb{Z}$ est pair, alors n est pair.

Solution

On démontre la contraposée « Si n est impair, alors n^2 est impair ».
On suppose n impair, c'est-à-dire $n = 2k+1$ avec $k \in \mathbb{Z}$.
Alors $n^2 = (2k+1)^2 = 4k^2 + 4k + 1 = 2(2k^2 + 2k) + 1 = 2\mathrm{K} + 1$
avec $\mathrm{K} = 2k^2 + 2k$. Or, $\mathrm{K} \in \mathbb{Z}$ donc n^2 est impair ».
La contraposée est vraie donc la proposition de l'énoncé est vraie.

C Démonstration par l'absurde

Pour démontrer qu'une proposition P est vraie, on suppose que la proposition non P est vraie et on en déduit une contradiction.

Énoncé : Démontrer que $\sqrt{2} \neq 1{,}414$.

Solution

On suppose que $\sqrt{2} = 1{,}414$. Deux nombres égaux ont le même carré donc $2 = 1{,}414^2$. Or $1{,}414^2 = 1{,}999396$, on obtient ainsi une contradiction.
Donc la proposition $\sqrt{2} = 1{,}414$ est fausse.

D Démonstration par contre-exemple

$\mathrm{P}(x)$ est une proposition définie pour les éléments x d'un ensemble E.
Pour démontrer que la proposition « Pour tout x de E, $\mathrm{P}(x)$ » est fausse, il suffit de trouver un contre-exemple, c'est-à-dire un élément x de E pour lequel $\mathrm{P}(x)$ est fausse.

Énoncé : La proposition « Pour tout nombre réel $x > 0$, $\frac{1}{x} \leqslant x$ » est-elle vraie ou fausse ?

Solution

Pour $x = 0{,}5$, $\frac{1}{x} = 2$ donc $\frac{1}{x} > x$ et la proposition énoncée est fausse.