



Fiche : Algorithmique et programmation

FICHE • M22 • TERMINALE

Spécialité de terminale, programme officiel. Cours original rédigé avec une IA. Le cours de ton enseignant fait foi.

Sommaire du cours

L'essentiel	1
Formules et définitions à connaître	2
Pièges à éviter	2
Je me teste	2

L'essentiel

- En Python, l'indentation fait partie de la syntaxe : elle délimite les blocs des boucles, des conditions et des fonctions.
- L'affectation $a = a + 1$ calcule d'abord $a + 1$ avec l'ancienne valeur, puis range le résultat dans a .
- Les types usuels sont `int`, `float`, `bool`, `str` et `list` ; le séparateur décimal est le point.
- `//` donne le quotient euclidien, `%` le reste, `/` un quotient décimal flottant, `**` la puissance.
- La boucle `for` s'utilise quand le nombre de tours est connu ; la boucle `while` quand on répète tant qu'une condition est vraie.
- `range(a, b)` donne les entiers de a à $b - 1$; les indices d'une liste vont de 0 à `len(L) - 1`.
- Une fonction renvoie un résultat avec `return`, qui termine immédiatement son exécution ; `print` ne fait qu'afficher.
- Une liste en compréhension s'écrit `[expression for x in L if condition]`.
- Un algorithme de seuil utilise une boucle `while` dont la condition est la négation de la condition d'arrêt.
- La dichotomie s'applique à une fonction continue qui change de signe sur $[a ; b]$: à chaque étape, l'intervalle est divisé par 2.
- Après n étapes de dichotomie, l'amplitude de l'encadrement vaut $(b - a)/2^n$.
- La méthode d'Euler approche la solution de $y' = F(x, y)$ par $y_{k+1} = y_k + h \times F(x_k, y_k)$ et $x_{k+1} = x_k + h$.
- Plus le pas h est petit, meilleure est l'approximation d'Euler, au prix de plus d'étapes.
- Le test `random() < p` simule un événement de probabilité p ; `randint(a, b)` inclut a et b .
- La fréquence observée sur n simulations se rapproche de la probabilité quand n grandit.
- Pour lire un algorithme, on dresse un tableau d'état des variables après chaque passage.



Formules et définitions à connaître

Outil	Écriture	À retenir
Somme par accumulation	$s = 0$ puis $s = s + \text{terme dans la boucle}$	initialiser à 0 (produit : à 1)
Parité	$n \% 2 == 0$	<code>==</code> teste, <code>=</code> affecte
Seuil	<code>while u < S</code> : mise à jour de u et de n	u et n avancent ensemble
Dichotomie	$m = (a + b)/2$; si $f(a) \times f(m) \leq 0$ alors $b = m$, sinon $a = m$	f continue, changement de signe
Nombre d'étapes	plus petit n tel que $(b - a)/2^n \leq e$	10 étapes pour 10^{-3} sur $[0 ; 1]$
Euler	$y_{k+1} = y_k + h \times F(x_k, y_k)$	nouveau y avec l'ancien x
Simulation	<code>random() < p</code>	vrai avec probabilité p
Fluctuation	$[p - 1/\sqrt{n} ; p + 1/\sqrt{n}]$	environ 95 % des échantillons, n grand

Pièges à éviter

- `range(1, n)` s'arrête à $n - 1$, et un parcours qui commence à l'indice 1 oublie le premier élément.
- Un `return` placé dans une boucle l'interrompt dès le premier passage.
- Une boucle `while` dont la variable testée n'est pas modifiée ne s'arrête jamais.
- Un `maximum` s'initialise au premier élément de la liste, pas à 0.
- Dans Euler, avancer x avant de calculer y fausse le calcul.
- On ne teste pas l'égalité de deux flottants : $0.1 + 0.2$ ne vaut pas exactement 0.3 en machine.

Je me teste

1. Que valent $17 // 5$ et $17 \% 5$? $\rightarrow 3$ et 2.
2. Quels entiers donne `range(2, 6)` ? $\rightarrow 2, 3, 4, 5$.
3. Que vaut `L[0]` pour `L = [4, 7, 1]` ? $\rightarrow 4$.
4. Quelle boucle pour un seuil ? $\rightarrow$ Une boucle `while`.
5. Différence entre `print` et `return` ? $\rightarrow$ `print` affiche, `return` transmet le résultat et termine la fonction.
6. Que donne `[k ** 2 for k in range(1, 4)]` ? $\rightarrow [1, 4, 9]$.
7. Combien d'étapes de dichotomie pour 10^{-3} sur un intervalle de longueur 1 ? $\rightarrow 10$, car $2^{10} = 1024$.
8. Relation d'Euler pour $y' = y$ avec un pas h ? $\rightarrow y_{k+1} = (1 + h) \times y_k$.
9. Que vaut $1,1^{10}$, approximation d'Euler de e avec $h = 0,1$? $\rightarrow$ Environ 2,594.
10. Comment simuler un événement de probabilité 0,3 ? $\rightarrow$ Tester `random() < 0.3`.