

Initiation à l'algorithmique

— introduction générale —

Jacques TISSEAU

Enib–Cerv

enib©2009-2014

Remarque (Notes de cours : couverture)

Ce support de cours accompagne le chapitre 1 des notes de cours « Initiation à l'algorithmique ».

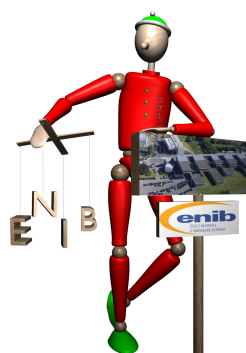


— Cours d'Informatique S1 —

Initiation à l'algorithmique

JACQUES TISSEAU

Ecole nationale d'ingénieurs de Brest
Centre européen de réalité virtuelle
tisseau@enib.fr



Ces notes de cours accompagnent les enseignements d'informatique du 1^{er} semestre (S1) de l'Ecole Nationale d'Ingénieurs de Brest (ENIB : www.enib.fr). Leur lecture ne dispense en aucun cas d'une présence attentive aux cours ni d'une participation active aux travaux dirigés.

Avec la participation de ROMAIN BÉCARD, STÉPHANE BONNEAUD, CÉDRIC BUCHE, GREG DESMUELLES, CÉLINE JOST, SÉBASTIEN KUBICKI, ERIC MAISEL, ALÉXIS NÉDÉLEC, MARC PARENTHOËN et CYRIL SEPTSEULT.

Définition

- **information automatique** (P. Dreyfus, 1962)
- **informatique** : science du traitement automatique de l'information

Matériel ↔ Logiciel

- **matériel** : ordinateurs (J. Perret, 1955)



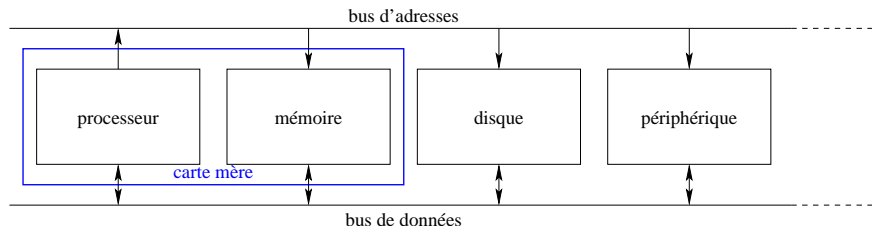
- **logiciel** : ensemble de programmes remplissant une fonction déterminée, permettant l'accomplissement d'une tâche donnée

Définitions

- **informatique** L'informatique est la science du traitement automatique de l'information.
- **matériel** Le matériel informatique est un ensemble de dispositifs physiques utilisés pour traiter automatiquement des informations.
- **logiciel** Le logiciel est un ensemble structuré d'instructions décrivant un traitement d'informations à faire réaliser par un matériel informatique.

Remarque (Cours Informatique S1)

Dans ce cours, nous ne nous intéresserons qu'aux aspects logiciels de l'informatique.



Architecture de Von Neumann

- élément central : processeur (unité arithmétique et logique, unité de contrôle)
- échanges avec les autres composants : stocker, récupérer et transférer des données
- bus d'adresse : désigner le composant
- bus de données : véhiculer l'information

Définitions

bit Un bit est un chiffre binaire (0 ou 1). C'est l'unité élémentaire d'information.

octet Un octet est une unité d'information composée de 8 bits.

TD (Unités d'information)

Combien y a-t-il d'octets dans 1 ko (kilooctet), 1 Go (gigaoctet), 1 To (téraoctet), 1 Po (pétaoctet), 1 Eo (exaoctet), 1 Zo (zettaoctet) et 1 Yo (yottaoctet) ?

TD (Stockage de données)

Donner l'ordre de grandeur en octets pour stocker en mémoire : une page d'un livre, une encyclopédie en 20 volumes, une photo couleur, une heure de vidéo, une minute de son, une heure de son.



Définitions

MIPS Le MIPS est l'unité de mesure qui représente un million d'instructions par seconde.

FLOPS Le FLOPS (Floating-point Operations Per Second) est l'unité de mesure qui représente le nombre d'opérations à virgule flottante par seconde.

TD (Puissance de calcul)

Donner l'ordre de grandeur en instructions par seconde des machines suivantes : le premier micro-ordinateur de type PC, une console de jeu actuelle, un micro-ordinateur actuel, Deep-Blue : l'ordinateur qui a « battu » Kasparov aux échecs en 1997, le plus puissant ordinateur actuel.



8086



80486



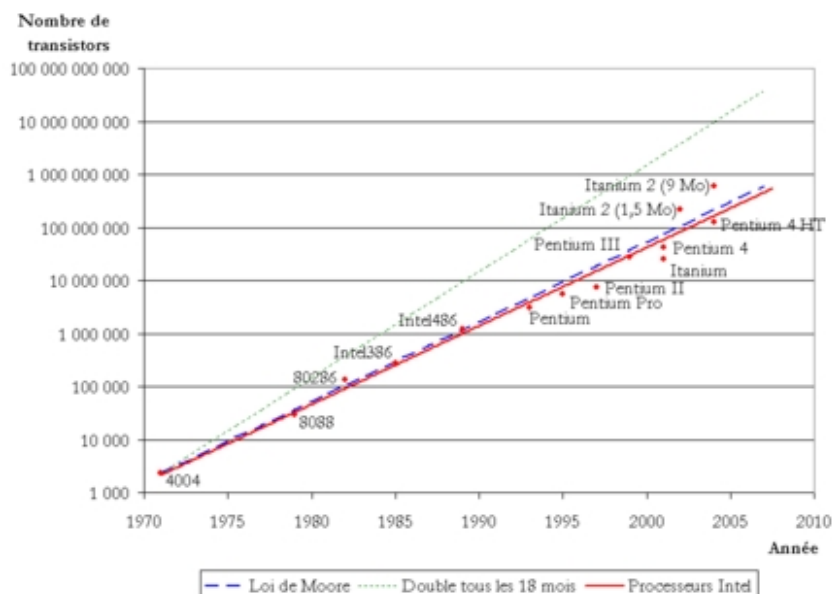
Pentium 4



Core Duo

Remarque (Loi de Moore)

Gordon Earl Moore est le co-fondateur avec Robert Noyce et Andrew Grove de la société Intel en 1968 (fabriquant n° 1 mondial de microprocesseurs). En 1965, il expliquait que la complexité des semiconducteurs doublait tous les dix-huit mois à coût constant depuis 1959, date de leur invention. En 1975, il précise sa « première loi » en affirmant que le nombre de transistors des microprocesseurs sur une puce de silicium double tous les deux ans (« deuxième loi »).



Algorithme

- mathématicien persan du 9^{ème} siècle **Al-Khwarizmi**
- méthode de calcul qui indique la démarche à suivre pour résoudre une série de problèmes équivalents en appliquant dans un ordre précis une suite finie de règles

Algorithmique

- art de construire des algorithmes
- validité, robustesse, réutilisabilité
- complexité, efficacité

Définitions

algorithme Un algorithme est une suite ordonnée d'instructions qui indique la démarche à suivre pour résoudre une série de problèmes équivalents.

algorithmique L'algorithmique est la science des algorithmes.

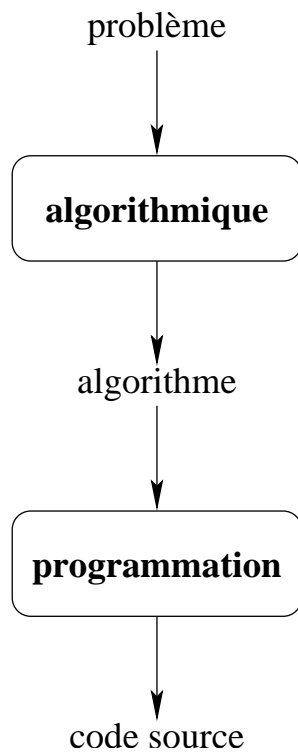
validité La validité d'un algorithme est son aptitude à réaliser exactement la tâche pour laquelle il a été conçu.

robustesse La robustesse d'un algorithme est son aptitude à se protéger de conditions anormales d'utilisation.

réutilisabilité La réutilisabilité d'un algorithme est son aptitude à être réutilisé pour résoudre des tâches équivalentes à celle pour laquelle il a été conçu.

complexité La complexité d'un algorithme est le nombre d'instructions élémentaires à exécuter pour réaliser la tâche pour laquelle il a été conçu.

efficacité L'efficacité d'un algorithme est son aptitude à utiliser de manière optimale les ressources du matériel qui l'exécute.



- Un algorithme exprime la structure logique d'un programme : il est indépendant du langage de programmation.

- La traduction de l'algorithme dans un langage de programmation dépend du langage choisi.

TD (Dessins sur la plage : exécution)

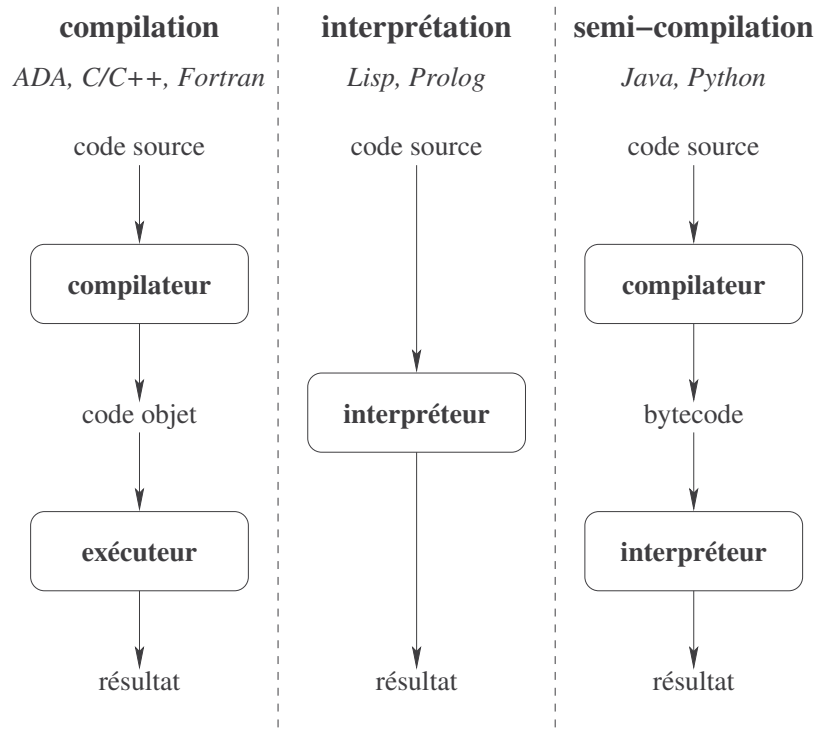
On cherche à faire dessiner une figure géométrique sur la plage à quelqu'un qui a les yeux bandés. Quelle figure géométrique dessine-t-on en exécutant la suite d'instructions suivante : avance de 10 pas, tourne à gauche d'un angle de 120° , avance de 10 pas, tourne à gauche d'un angle de 120° , avance de 10 pas ?

TD (Dessins sur la plage : conception)

Faire dessiner une spirale rectangulaire de 5 côtés, le plus petit côté faisant 2 pas de long et chaque côté fait un pas de plus que le précédent.

TD (Mon premier programme en Python)

Ecrire en Python l'algorithme de la spirale rectangulaire à 5 côtés.



Définitions

compilateur Un compilateur est un programme informatique qui traduit un langage, le langage source, en un autre, appelé le langage cible. Un compilateur sert le plus souvent à traduire un code source écrit dans un langage de programmation en un autre langage, habituellement un langage d'assemblage ou un langage machine. Le programme en langage machine produit par un compilateur est appelé code objet.

interpréteur Un interpréteur est un outil informatique (logiciel ou matériel) ayant pour tâche d'analyser et d'exécuter un programme écrit dans un langage source.

TD (Première utilisation de Python)

1. Lancer Python.
2. Utiliser Python comme une simple calculatrice.
3. Exécuter l'algorithme de la spirale rectangulaire à 5 côtés.
4. Quitter Python.

Objectifs

acquérir les notions fondamentales de l'**algorithmique** et les mettre en œuvre avec un langage opérationnel (**Python**).

Pré-requis

- I Bac scientifique

3 objectifs majeurs

1. instructions de base
2. procédures et fonctions
3. structures de données linéaires

Remarque (Objectifs thématiques)

Plus précisément, nous étudierons successivement :

1. *les instructions de base permettant de décrire les algorithmes : affectation, tests, boucles ;*
2. *les procédures et les fonctions qui permettent de structurer et de réutiliser les algorithmes ; on parlera alors d'encapsulation, de préconditions, de portée des variables, de passage de paramètres, d'appels de fonctions, de récursivité et de jeux de tests ;*
3. *les structures de données linéaires : tableaux, listes, piles, files, qui améliorent la structuration des données manipulées par les algorithmes. A cette occasion, on évaluera la complexité et l'efficacité de certains algorithmes utilisant ces structures linéaires.*

Ces différentes notions seront mise en œuvre à travers l'utilisation du langage Python.

Faire ses gammes

- apprentissage d'un langage algorithmique (semestre 1)
- pédagogie par objectifs

Jouer les grands classiques

- apprentissage des algorithmes classiques (semestres 1 et 2)
- pédagogie par l'exemple

Composer ses propres morceaux

- apprentissage de la conception d'algorithmes (semestre 2)
- pédagogie par problèmes

Remarque (Taxonomie de Bloom)

Benjamin Bloom (1913-1999) psychologue américain spécialisé en pédagogie.

S1	Connaître : définir, distinguer, acquérir, identifier, rappeler, reconnaître...
	Comprendre : traduire, illustrer, représenter, dire avec ses mots, distinguer, réécrire, réarranger, expliquer, démontrer...
	Appliquer : appliquer, généraliser, relier, choisir, développer, utiliser, employer, transférer, classer, restructurer...
S2	Analyser : distinguer, détecter, classer, reconnaître, catégoriser, déduire, discerner, comparer...
	Synthétiser : écrire, relater, produire, constituer, transmettre, modifier, créer, proposer, planifier, projeter, spécifier, combiner, classer, formuler...
	Evaluer : juger, argumenter, valider, décider, comparer...

Savoir-être

■ rigueur

respect des consignes, précision, exactitude
écrit ↔ image

■ persévérance

aller au bout des choses
finir ↔ papillonner

■ autonomie

pratique personnelle, autoformation
initiatives ↔ assistances

Définitions

rigueur La rigueur est la qualité de celui qui agit avec précision, exactitude, en respectant une logique inflexible.

persévérance La persévérance est la qualité de celui qui s'attache avec détermination et constance à mener à bien ce qu'il a entrepris.

autonomie L'autonomie est la qualité de celui qui est capable d'agir sans intervention extérieure.

TD (Rigueur : erreur de syntaxe en Python)

On considère la session Python suivante :

```
»> x = 3
»> y = x
    File "<stdin>", line 1
      y = x
      ^
SyntaxError: invalid syntax
»>
```

De quelle erreur de syntaxe s'agit-il ?

TD (Persévérance : dessins sur la plage)

Finir l'algorithme suivant qui cherche à dessiner un losange sur la plage :
avance de 10 pas, ...

Horaires

- Cours/TD : 21h (1× 1h30 chaque semaine) en salle banalisée
- TD : 21h (1× 3h toutes les 2 semaines) en salle informatique

Planning prévisionnel

Voir [site Web](#)

ENT – Informatique S1

Remarque (Exemple de planning prévisionnel)

Les enseignements d'Informatique S1 de l'ENIB s'étalent sur 14 semaines à raison de 3h de cours toutes les 2 semaines en alternance avec 3h de laboratoire toutes les 2 semaines.

	Cours	Laboratoire
1	<i>instructions de base</i>	—
2	—	<i>instructions de base</i>
3	<i>instructions de base</i>	—
4	—	<i>instructions de base</i>
5	<i>instructions de base</i>	—
6	—	<i>instructions de base</i>
7	<i>procédures et fonctions</i>	—
8	—	<i>procédures et fonctions</i>
9	<i>procédures et fonctions</i>	—
10	—	<i>procédures et fonctions</i>
11	<i>procédures et fonctions</i>	—
12	—	<i>procédures et fonctions</i>
13	<i>structures linéaires</i>	—
14	—	<i>structures linéaires</i>

Support de cours

- copie papier des transparents projetés pendant le cours
- plage de prise de notes

Notes de cours

- compléments au cours
- exercices corrigés

Site web

- planning prévisionnel
- exemples corrigés d'évaluations
- notes des élèves
- forum

Remarque (Travail personnel)

- *Les notes de cours comportent 259 pages structurées en 4 chapitres, 3 annexes, 4 index et une bibliographie. Elles proposent 47 définitions, 86 figures, 39 exemples, 79 remarques, 128 exercices et 5 contrôles types corrigés.*
- *En moyenne, au cours des 14 semaines que dure le cours d'informatique S1 de l'ENIB, le travail personnel hebdomadaire consiste donc à lire entre 15 et 20 pages de ce cours en retenant 3 à 4 définitions et en faisant entre 7 et 10 exercices.*

TD (Site Web d'Informatique S1)

Se connecter sur le site Web du cours d'informatique S1 de l'ENIB et vérifier que ce support de cours est bien disponible sur le site au format pdf.

Préparation

Cours autoformation, QCM

Laboratoire préparer les exercices

Participation

Par respect pour les autres, la **ponctualité** est de rigueur pour l'étudiant comme pour le professeur.

Cours présence attentive et soutenue

TD participation active et volontaire

Laboratoire chacun doit être « lecteur » et « écrivain »

Appropriation

Cours relire les notes le soir même (« fixer » les idées)

Laboratoire refaire les TD qui ont posé des problèmes

Remarque (Apprendre en faisant)

Dans tous les cas, l'expérience montre que :

1. *la seule présence, même attentive et active, aux séances de cours et de laboratoire ne suffit pas : il faut prévoir un temps de travail personnel qui, selon l'étudiant et la matière, peut aller de 50% à 150% du temps de présence en cours ;*
2. *la régularité dans le travail personnel est un gage d'apprentissage plus efficace.*

*Il n'y a pas de miracle, c'est votre travail personnel qui est le meilleur gage de vos apprentissages. **On apprend toujours mieux en faisant par soi-même.***

Remarque (Empathie numérique)

*Lorsque l'exemple est un algorithme, il faut systématiquement se mettre mentalement à la place de la machine qui va les exécuter (on parle alors d'« empathie numérique ») afin de vérifier le résultat obtenu. Pour cela, il faut être méthodique et rigoureux. Et petit à petit, à force de pratique, l'expérience fera qu'on « verra » le résultat produit par les instructions au fur et à mesure qu'on les écrit. **Naturellement, cet apprentissage est long, et demande des heures de travail patient.** Aussi, dans un premier temps, il faut éviter de sauter les étapes : la vérification méthodique, pas à pas, de chacun des algorithmes représente plus de la moitié du travail à accomplir.*

Exposé

- présentation orale
- durée 5'

Contrôle d'autoformation

- contrôle des connaissances acquises par auto-formation
- durée 30' (en début d'une séance de cours)

Contrôle d'attention

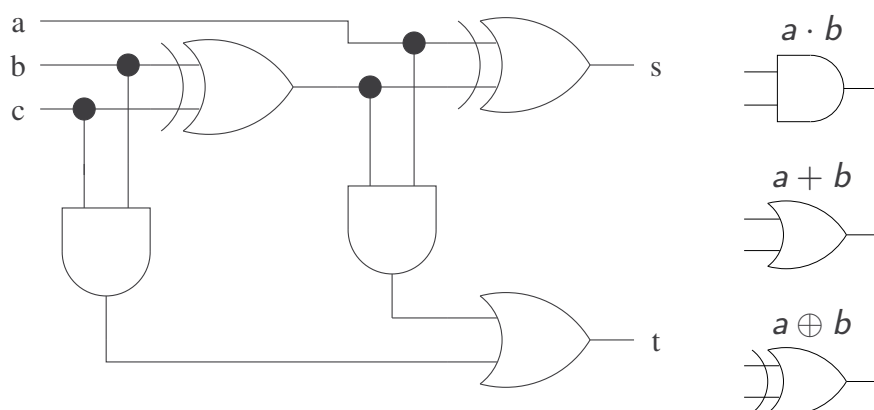
- QCM « à chaud » sur les points abordés pendant un cours
- durée 5' (en fin d'une séance de cours)



note de contrôle continu de TD (coefficient 1)

TD (Exemple de contrôle d'autoformation)

Etablir la table de vérité du circuit logique ci-dessous où a , b et c sont les entrées, s et t les sorties.



Contrôle de Laboratoire

- contrôle sur la préparation des exercices de laboratoire
- durée 15' (en début d'une séance de laboratoire)
- note de contrôle continu de laboratoire (coefficient 1)

Contrôle de synthèse

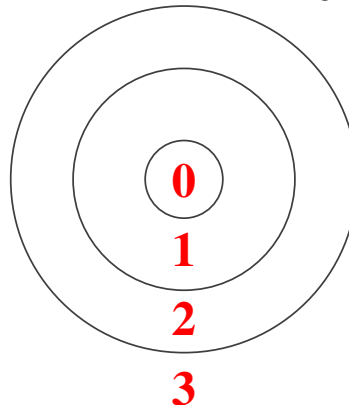
- contrôle des compétences acquises sur un thème donné
- durée 1h30 (en dehors de la grille horaire)
- 1 DS (coefficient 1)

TD (Exemple de QCM)

un seul item correct par question

1. *L'informatique est la science*
 - 1.1 *des signaux électriques porteurs d'information ou d'énergie*
 - 1.2 *du mouvement des systèmes matériels et de leurs déformations*
 - 1.3 *du traitement automatique de l'information*
 - 1.4 *de la commande des appareils fonctionnant sans intervention humaine*
2. *Le logiciel est*
 - 2.1 *la mémoire de l'ordinateur*
 - 2.2 *le traitement automatique de l'information*
 - 2.3 *l'ensemble des données manipulées par les instructions*
 - 2.4 *un ensemble structuré d'instructions décrivant un traitement d'informations à faire réaliser par un matériel informatique*
3. *La validité d'un algorithme est son aptitude*
 - 3.1 *à utiliser de manière optimale les ressources du matériel qui l'exécute*
 - 3.2 *à se protéger de conditions anormales d'utilisation*
 - 3.3 *à calculer le nombre d'instructions élémentaires nécessaires pour réaliser la tâche pour laquelle il a été conçu*
 - 3.4 *à réaliser exactement la tâche pour laquelle il a été conçu*

note \equiv distance à l'objectif

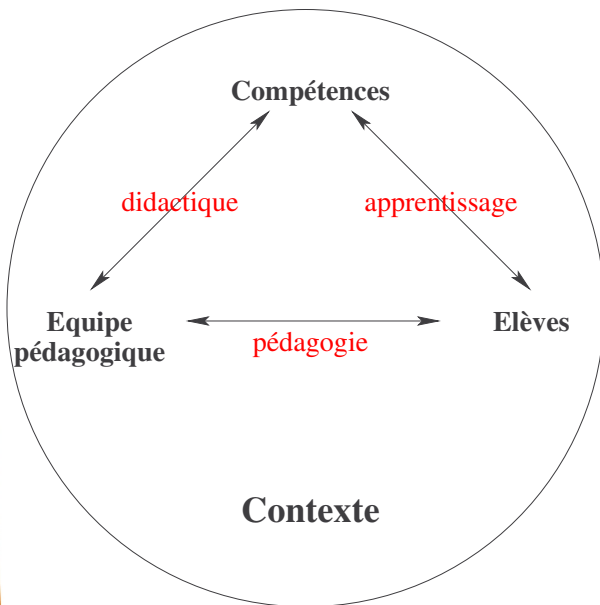


- 0 : « en plein dans le mille ! » → l'objectif est atteint
- 1 : « pas mal ! » → on se rapproche de l'objectif
- 2 : « tout juste sur la cible ! » → on est encore loin de l'objectif
- 3 : « même pas touchée ! » → l'objectif n'est pas atteint
- 4 : « même pas visée ! » → absence

Remarque (Notation)

Ainsi, et pour changer de point de vue sur la notation, le contrôle est réussi lorsqu'on a 0 ! Il n'y a pas non plus de 1/2 point ou de 1/4 de point : le seul barème possible ne comporte que 4 niveaux : 0, 1, 2 et 3. On ne cherche donc pas à « grappiller » des points :

- on peut avoir 0 (objectif atteint) et avoir fait une ou deux erreurs bénignes en regard de l'objectif recherché ;
- on peut avoir 3 (objectif non atteint) et avoir quelques éléments de réponse corrects mais sans grand rapport avec l'objectif recherché ;
- on a 4 en cas d'absence.



- « contrat pédagogique »
- 1 évaluation des enseignements par semestre (sur le site Web)
- améliorer les enseignements

Définitions

- pédagogie** La pédagogie désigne les méthodes et pratiques d'enseignement et d'éducation ainsi que toutes les qualités requises pour transmettre un savoir quelconque.
- didactique** La didactique se différencie de la pédagogie par le rôle central des contenus disciplinaires et par sa dimension épistémologique (la nature des connaissances à enseigner).
- apprentissage** L'apprentissage est l'acquisition de nouveaux savoirs ou savoir-faire, c'est-à-dire le processus d'acquisition de connaissances, compétences, attitudes ou valeurs, par l'étude, l'expérience ou l'enseignement.