

Didactique de l'Informatique et NSI

Journée Recherche FST - 9 Juin 2023

LIM - axe EDIM

Histoire et didactique des mathématiques

- Dominique Tournès (LIM, Dpt de Mathématiques)
- Jean Berky Nguuala (LIM, CUFR de Mayotte)
- Luc Tiannot (LIM, INSPÉ de la Réunion)
- Jean-Jacques Salone (Associé LIM, CUFR de Mayotte)

Didactique de l'informatique

- Abal-Kassim Cheik Ahamed (Associé LIM, CUFR de Mayotte)
- Sophie Chane Lune (Doctorante LIM, Enseignante NSI)
- Christophe Declercq (LIM, INSPÉ et Dpt. Informatique)
- Sébastien Hoarau (LIM, Dpt. Informatique)
- Frédéric Rakotomalala (Postdoc, Université d'Antananarivo)

Le focus d'aujourd'hui

- 1 Nos motivations
- 2 Compétences
- 3 Activités (et instruments)

Comment enseigner la programmation à des débutants ?





Elie Milgrom, UCLouvain, FA2L nous rappelle¹

Principe n°1

L'objectif principal de tout enseignement (et de tout enseignant) est l'apprentissage réalisé par les étudiants.

¹Journée enseignement de la SIF (mai 2023)

Principe n°2

Construire un dispositif aligné pédagogiquement en explicitant les objectifs.

Principe n°3

Ce que fait l'étudiant détermine ce qu'il apprend.



Florent Bouchez Tichadou, Univ. Grenoble Alpes
Cours de programmation et algorithmique en L2 Informatique

Objectifs de l'APP1 : Ydana wheya

À l'issue de l'APP1, vous serez capable de :

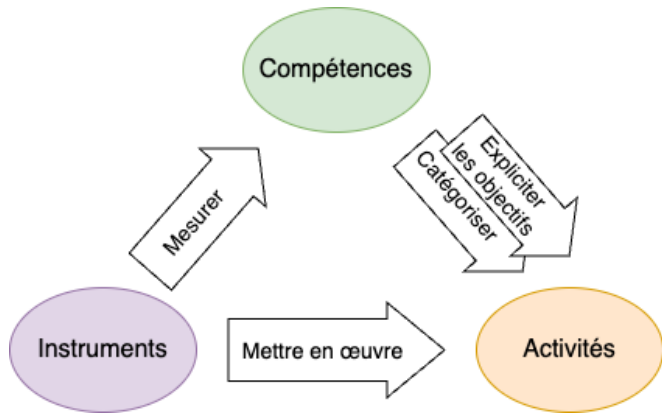
- Effectuer diverses manipulation sur les ensembles et séquences représentés par tableau avec longueur explicite : insertion, suppression, parcours ;
- Utiliser les ensembles et séquences pour implanter des listes d'associations (ou dictionnaires) ;
- Choisir le type de données approprié pour traiter un problème ;
- Analyser la complexité algorithmique d'algorithmes utilisant ces structures de données.

Quels dispositifs pour que les débutants apprennent à programmer ?

Sachant que *programmer* est entendue comme tâche complexe englobant :

- modéliser un problème
- anticiper des traitements
- mettre en œuvre via un langage de programmation (Python)

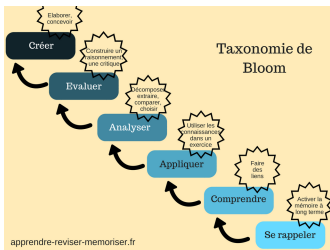
Nos outils... un tryptique



Constat

Il n'existe pas de référentiel de compétences pour les débutants en programmation.

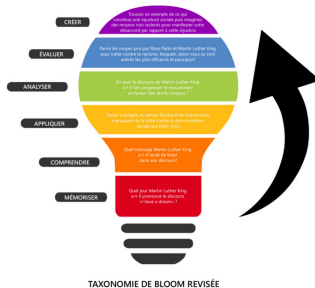
Compétences (2/5) Méta-compétences



LA TAXONOMIE DE BLOOM



Objectif cognitif	Connaissance	Compréhension	Application	Analyse	Synthèse	Évaluation
Verbes d'action	1. Citer 2. Définir 3. Définir 4. Définir 5. Définir 6. Définir 7. Définir 8. Définir 9. Définir Etc.	1. Expliquer 2. Illustrer 3. Parler 4. Répondre 5. Répondre 6. Traduire 7. Traduire 8. Traduire 9. Traduire Etc.	1. Appliquer 2. Adapter 3. Appliquer 4. Appliquer 5. Calculer 6. Choisir 7. Choisir 8. Choisir 9. Choisir Etc.	1. Décomposer 2. Identifier 3. Identifier 4. Identifier 5. Identifier 6. Identifier 7. Identifier 8. Identifier 9. Identifier Etc.	1. Comparer 2. Comparer 3. Comparer 4. Comparer 5. Comparer 6. Comparer 7. Comparer 8. Comparer 9. Comparer Etc.	1. Evaluer 2. Juger 3. Argumenter 4. Argumenter 5. Discuter 6. Discuter 7. Discuter 8. Discuter 9. Discuter Etc.



Contenus

- tableau indexé
- tableau donné en compréhension

Capacités attendues

- Lire et modifier les éléments d'un tableau grâce à leurs index.
- Construire un tableau par compréhension.
- Utiliser des tableaux de tableaux pour représenter des matrices : notation $a[i][j]$.
- Itérer sur les éléments d'un tableau.

Compétences (4/5) Extrait d'un travail en cours

Création d'un référentiel utilisable

La pensée informatique définie par J.W. Wing (2006) se décline en méta-compétences (C. Declercq, 2021) croisées ici avec les capacités attendues du BO :

	<i>Construire un tableau et accéder à ses éléments</i>	<i>Parcourir un tableau pour rechercher ou calculer des informations</i>	<i>Parcourir et modifier un tableau par des boucles imbriquées</i>
Évaluer	E1 : évaluer le résultat d'un programme utilisant un tableau	E2 : évaluer le résultat d'un programme itérant sur un tableau	E3 : évaluer les valeurs successives des indices avec des boucles imbriquées
Modéliser	M1 : modéliser une série d'informations par un tableau donné en extension	M2 : modéliser un traitement de tableau en choisissant d'itérer sur les indices ou les éléments	
Anticiper	A1 : anticiper les traitements à programmer pour accéder aux éléments d'un tableau par leurs indices et les modifier	A2 : anticiper un traitement nécessitant un parcours linéaire d'un tableau	A3 : anticiper un traitement de tableau nécessitant plusieurs parcours imbriqués

Définir des objectifs

Activités en *pair-programming* pour programmer le tri d'un tableau.

	<i>Construire un tableau et accéder à ses éléments</i>	<i>Parcourir un tableau pour rechercher ou calculer des informations</i>	<i>Parcourir et modifier un tableau par des boucles imbriquées</i>
Évaluer	E1 : évaluer le résultat d'un programme utilisant un tableau	E2 : évaluer le résultat d'un programme itérant sur un tableau	E3 : évaluer les valeurs successives des indices avec des boucles imbriquées
Modéliser	M1 : modéliser une série d'informations par un tableau donné en extension	M2 : modéliser un traitement de tableau en choisissant d'itérer sur les indices ou les éléments	
Anticiper	A1 : anticiper les traitements à programmer pour accéder aux éléments d'un tableau par leurs indices et les modifier	A2 : anticiper un traitement nécessitant un parcours linéaire d'un tableau	A3 : anticiper un traitement de tableau nécessitant plusieurs parcours imbriqués

Exemples d'activités

Inciter à modéliser et réviser quelques bases

Simulation d'une course *shifumi*

Programmer un tri par sélection

Activité en *pair-programming*

Découvrir la structure de Pile

Activité ludique : jeu du saute moutons

Présenter algorithmes gloutons et leurs limites

Activité ludique : jeu du sac à dos



- Basic Principles of Curriculum and Instruction, Ralph. W. Tyler, 1949
- Enhancing teaching through constructive alignment, John Biggs, 1996
- Computational thinking, Jeannette M. Wing, 2006

- Illustration *Apprendre* : Site Lutin Bazar
- Illustration *Questions* : Designed by Freepik

