



Enseignement secondaire classique
Classes supérieures
Section B
INFOR – Informatique
PROG – Programme
1CB_1CLB

Langue véhiculaire :	Luxembourgeois et français, terminologie de programmation en anglais
Nombre de leçons :	2
Nombre minimal de devoirs par semestre :	2
Dernière mise à jour par la CNES :	21/05/2021

Partie 1 – Techniques de programmation (20 heures)

Chapitre 1^{er} : Réutilisation de code – modules (4h)

- Connaissances
 - Vocabulaire technique (package, module, collection)
 - Notation
- Compétences
 - Installer des modules ou paquets
 - Lecture des documentations officielles des modules et des paquets
 - Utiliser des fonctions existantes (import, appel de fonctions etc.)
- Fichiers-textes : lecture, écriture, modification de fichiers-textes

Chapitre 2 : Récursivité (6h)

- Connaissances
 - Fonctions itératives vs fonctions récursives (exemples : pgcd, factorielle)
- Exercices d'entraînement
 - Fibonacci
 - Suite de Collatz
 - Recherche dans une liste

Chapitre 3 : Programmation Orientée Objet (10h)

- Connaissances
 - o Vocabulaire
 - o Définitions (classe, objet, attributs, méthodes, constructeur)
 - o Diagrammes UML (respecter scrupuleusement les conventions du manuel !)
- Compétences
 - o Utiliser des classes existantes
 - o Créer sa propre classe (p.ex. fractions)
- Exercices : création de nouvelles classes
- Application : la classe *Stack*
- Exercices divers de synthèse ; simulations

Partie 2 – Jeux et applications graphiques (20 heures)

Chapitre 4 : Pygame (20h)

- Connaissances
 - o Mise en place d'un programme Pygame
 - o Éléments graphiques (couleurs, fenêtre, ligne, rectangle, ellipse, cercle)
 - o Gestion des événements (quit, clavier, souris)
 - o Gestion du temps (clock)
 - o Insertion de textes dans la fenêtre Pygame (positionnement, formatage, polices)
- Exercices
 - o Triangle de Sierpinski
 - o Approximation de π par la méthode de Monte-Carlo
 - o Mouvement Brownien
 - o Ensemble de Mandelbrot
 - o Carré mobile
 - o Gestion de la souris
 - o Gestion de rectangles
 - o Balles
 - o Tic-Tac-Toe
 - o Particules en mouvement
 - o Affichage de textes
 - o Alien invasion
- Exercices divers de synthèse

Facultatif :

[Chapitre 5 : Initiation à l'algorithmique (6h)]

- Définition d'un algorithme, conditions
- Notation de Landau
- Mesures d'efficacité d'un algorithme (complexité temporelle)
- Exercices
- Application : Algorithmes de tri (*Selection Sort, Insertion Sort, Quicksort*)

[Chapitre 6 : Représentation graphique de données (4h)]

- Illustrations graphiques
 - Module matplotlib.pyplot
- Exercices
 - Représentation d'un polynôme
 - Représentation d'un polynôme, de sa dérivée et de sa dérivée seconde
 - Représentation d'un polynôme et de sa primitive
- Application
 - Méthode de Monte-Carlo pour approcher la valeur de π
- Illustration du fonctionnement des algorithmes de tri vus au cours

Recommandations pour les devoirs en classe

2 devoirs en classe par semestre.

L'énoncé sera formulé en français (avec terminologie de programmation en anglais).

Il est recommandé de chercher un équilibre entre les types de questions suivants :

- Problèmes de compréhension : extension/modification de programmes connus ;
- Problèmes ouverts : élaboration de nouveaux programmes en Python.
On pourra s'inspirer des exercices facultatifs disponibles.

Le recours aux diagrammes UML est facultatif dans l'énoncé des devoirs en classe.