# Introduction à la programmation informatique

TEM0013-CTEA_

Domaine : **Sciences de l'ingénieur et technologie** Grade académique : **Master en sciences de l'ingénieur industriel, orientation électromécanique - Arlon**
Niveau de certification : **7** Cycle : **Deuxième cycle** Année académique : **2026-2027** ( Q1 Q2 )

| CRÉDITS ECTS | VOLUME HORAIRE |
|---|---|
| **4** | **45 h** |
| PRÉREQUIS<br>*Aucun* | CO-REQUIS<br>*Aucun* |

| IMPLANTATION | DIRECTEUR DE DÉPARTEMENT DU DOMAINE | RESPONSABLE DE L'UE | ENSEIGNANTS INTERVENANTS |
|---|---|---|---|
| Département technique d'Arlon | Benoît Dujardin | Florent MIGNON | Florent MIGNON |

PROGRAMME D'ÉTUDES

| Programme | Bloc | Caractère |
|---|---|---|
| CC-Sciences industrielles | 1 | Choix individuel |

## Partie 1 Contribution de l'UE au profil d'enseignement du programme et acquis d'apprentissage spécifiques (AAS) sanctionnés par l'évaluation

### CONTEXTUALISATION

Cette UE contribue à l'atteinte des acquis d'apprentissage terminaux suivants :

· Mobiliser des concepts des sciences fondamentales afin de résoudre des problèmes spécifiques aux sciences et techniques de l'ingénieur.

· Mettre en œuvre des techniques d'algorithmique et de programmation et utiliser les outils numériques spécifiques aux sciences et techniques de l'ingénieur.

· Agir de façon réflexive, autonome et responsable.

### ACQUIS D'APPRENTISSAGE SPÉCIFIQUES (AAS)

Au terme de l'Unité d'Enseignement, l'étudiant sera capable de :

- Concevoir des algorithmes résolvant des tâches simples
- Comprendre les concepts fondamentaux de la programmation en Python
- Appliquer les structures de contrôle pour résoudre des problèmes
- Créer des fonctions et des classes en utilisant la programmation orientée objet
- Manipuler des structures de données pour stocker et organiser des informations
- Gérer les erreurs et exceptions pour garantir la stabilité des programmes
- Utiliser des bibliothèques et modules Python pour étendre les fonctionnalités
- Transcrire un algorithme en son programme équivalent en Python
- Modéliser en langage Python une situation simple du monde réel
- Structurer un problème complexe en sous-problèmes
- Analyser un code pour en décrire le résultat ou le corriger
- Argumenter la validité de décision de conception de base dans un programme orienté-objet
- Tester l'adéquation du résultat d'un programme par rapport à la tâche visée
- Réaliser de façon autonome une application de petite taille au moyen du langage Python et en utilisant les concepts fondamentaux de la programmation orientée objet

### OBJECTIFS DE DÉVELOPPEMENT DURABLE

Document annexe à la fiche UE accessible via un lien explicitant en quoi le programme d'études s'inscrit dans une politique de développement durable. Une ou plusieurs UE spécifique(s) pourraient être ciblée(s) dans cette annexe.

## Partie 2 Description de l'unité d'enseignement

### OBJECTIFS

L'UE *"Introduction à la programmation informatique"* vise à fournir aux étudiants une compréhension solide des concepts de base de la programmation en utilisant le langage Python.

### CONTENU

L'UE couvrira les sujets suivants (sous réserve de temps):

- Introduction à Python et Environnement de Développement
- Variables, Types de Données et Opérations Fondamentales
- Structures de Contrôle: Conditions et Boucles
- Fonctions et Modularité dans Python
- Structures de Données: Listes, Tuples, Dictionnaires
- Programmation Orientée Objet (POO) en Python
- Gestion des Erreurs et Exceptions
- Manipulation de Fichiers et Entrées/Sorties
- Introduction aux Bibliothèques et Modules Python
- Applications Pratiques: Traitement de Données, Visualisation, etc.

### SOURCES ET RÉFÉRENCES

Cordeau, Bob. "Python 3 : Apprendre à programmer dans l'écosystème Python Ed. 2." Dunod, 2020.

Swinnen, Gérard. "Apprendre à programmer avec Python 3 : Avec 60 pages d'exercices corrigés ! 3e édition." Eyrolles, 2012.

Beury, Jean-Noël. "Informatique avec Python - Classes prépas scientifiques : Exercices incontournables." Dunod, 2018.

### SUPPORTS DE COURS

| Activité d'apprentissage | Support |
|---|---|
| **ASCA2013** — Introduction à la programmation informatique | Tous les documents nécessaires au cours sont disponibles sur Moodle (PowerPoint, les liens web, les outils à télécharger et installer).<br>Certaines sources, documentation ou références en ligne sont en anglais. |

## Partie 3 Méthodologie mise en œuvre et éventuel découpage de l'UE

### DESCRIPTION DES ACTIVITÉS D'APPRENTISSAGE (AA) AU REGARD DE LEUR STATUT

| Activité d'apprentissage | Heures | Langue d'enseignement | Langue d'évaluation |
|---|---|---|---|
| **ASCA2013** — Introduction à la programmation informatique | 45 h | Français | Français |

### MÉTHODES D'ENSEIGNEMENT ET D'APPRENTISSAGE

| Activité d'apprentissage | Méthodes d'enseignement et d'apprentissage |
|---|---|
| **ASCA2013** — Introduction à la programmation informatique | Cours théorique expliqué par des exemples concrets avec des exercices à faire en classes, des exercices à faire en autonomie pour les séances suivantes, et des exercices de synthèses évalués. |

### DÉTERMINATION DE LA COHÉRENCE PÉDAGOGIQUE EN CAS DE REGROUPEMENT D'AA FAISANT L'OBJET D'UNE ÉVALUATION DISTINCTE

Sans objet

## Partie 4 Évaluation de l'unité d'enseignement

### STRUCTURE DU DISPOSITIF GÉNÉRAL D'ÉVALUATION (EN LIEN AVEC LE PROGRAMME D'ÉTUDES)

Travaux, interrogations, examen en session.

UE remédiable.

### MÉTHODOLOGIE DE CONSTRUCTION DE LA NOTE FINALE DE L'UE

La note finale de l'UE correspond à celle de l'activité d'apprentissage.

### LA PONDÉRATION DE L'ÉVALUATION DE L'UE À DES FINS DE DÉLIBÉRATION LORS DU CALCUL DE LA MOYENNE

Au sein du programme d'études, lors du calcul de la moyenne, la pondération de l'évaluation de l'unité d'enseignement est effectuée sur base des crédits affectés à cette unité d'enseignement et évalués

### MODALITÉS D'ÉVALUATION PAR ACTIVITÉ D'APPRENTISSAGE

| Activité d'apprentissage | Modalités d'évaluation |
| --- | --- |
| **ASCA2013** — Introduction à la programmation informatique | Q1<br><br>Travaux:<br><br>Interros:<br><br>Questionnaires et/ou exercice(s) sur PC (**valeur 20%**)<br><br>Q2<br><br>Travaux:<br><br>Interros:<br><br>Exercice(s) sur PC (**valeur 10%**)<br><br>Examen (JUIN):<br><br>Examen 30 min QCM + 1h30 exercices sur PC (**valeur 70%**)<br><br>Q3<br><br>Examen (AOUT):<br><br>Examen 30 min QCM + 1h30 exercices sur PC (**valeur 100%**) |

**MODALITÉS CONCRÈTES D'ÉVALUATION**

Une mention sur la fiche informera les étudiants du fait que les précisions sur les épreuves seront disponibles sur Moodle au plus tard 6 semaines avant l'évaluation.