

<b>Intitulé de l'UE</b>	<b>Programmation - Concepts avancés</b>
<b>Section(s)</b>	- (8 ECTS) Bachelier en Informatique orientation Réseaux et Télécommunications / Cycle 1 Bloc 1

<b>Responsable(s)</b>	<b>Heures</b>	<b>Période</b>
Erwin DESMET	84	Quad 2

<b>Activités d'apprentissage</b>	<b>Heures</b>	<b>Enseignant(s)</b>
<b>Algorithmie</b>	18h	<b>Johan DEPRETER</b>
<b>Programmation avancée - modélisation</b>	14h	<b>Erwin DESMET</b>
<b>Programmation avancée - théorie</b>	12h	<b>Erwin DESMET</b> Johan DEPRETER
<b>Programmation avancée - travaux pratiques</b>	40h	<b>Erwin DESMET</b> Johan DEPRETER Fabrice SCOPEL

<b>Prérequis</b>	<b>Corequis</b>

<b>Répartition des heures</b>
<b>Algorithmie</b> : 18h de théorie
<b>Programmation avancée - modélisation</b> : 14h de théorie
<b>Programmation avancée - théorie</b> : 12h de théorie
<b>Programmation avancée - travaux pratiques</b> : 40h d'exercices/laboratoires

<b>Langue d'enseignement</b>
<b>Algorithmie</b> : Français, Anglais
<b>Programmation avancée - modélisation</b> : Français
<b>Programmation avancée - théorie</b> : Français, Anglais
<b>Programmation avancée - travaux pratiques</b> : Français, Anglais

<b>Connaissances et compétences préalables</b>
<b>[T-PINI-206] Algorithmique</b> Maîtrise de la programmation itérative et impérative

**[T-PINI-203] Programmation avancée - théorie**

Avoir suivi L'UE Programmation - Base et Algorithmique

**[T-PINI-204] Programmation avancée - travaux pratiques**

Avoir suivi L'UE Programmation - Base et Algorithmique

**Objectifs par rapport au référentiel de compétences ARES**

Cette UE contribue au développement des compétences suivantes

**Objectifs de développement durable****Education de qualité**

Objectif 4 Assurer l'accès de tous à une éducation de qualité, sur un pied d'égalité, et promouvoir les possibilités d'apprentissage tout au long de la vie

- 4.4 D'ici à 2030, augmenter considérablement le nombre de jeunes et d'adultes disposant des compétences, notamment techniques et professionnelles, nécessaires à l'emploi, à l'obtention d'un travail décent et à l'entrepreneuriat.

**Paix, justice et institutions efficaces**

Objectif 16 Promouvoir l'avènement de sociétés pacifiques et ouvertes aux fins du développement durable, assurer l'accès de tous à la justice et mettre en place, à tous les niveaux, des institutions efficaces, responsables et ouvertes

- 16.7 Faire en sorte que le dynamisme, l'ouverture, la participation et la représentation à tous les niveaux caractérisent la prise de décisions.

**Acquis d'apprentissage spécifiques****[T-PINI-206] Algorithmique**

- Enumérer et définir les structures de données usuelles
- Expliquer le fonctionnement des algorithmes vus au cours
- Mettre en oeuvre les concepts vu au cours afin de résoudre des problèmes de programmation
- Choisir une structure de données (ou un paradigme de programmation) adaptées afin de résoudre un problème

**[T-PINI-203] Programmation avancée - théorie**

- Acquérir les bases nécessaires à la conception de programmes
- Appliquer les méthodologies de programmation
- Réalisation de diagrammes UML

**[T-PINI-204] Programmation avancée - travaux pratiques**

Au terme de ce laboratoire, l'étudiant sera capable de :

- Schématiser un diagramme UML modélisant un objet en utilisant des concepts de Programmation Orientée Objet (tels que accesseurs/mutateurs, visibilité, héritage,...);
- Etablir un code informatique à partir de ce diagramme UML vers un langage cible ;
- Etablir des classes et leurs composants
- Créer une application graphique à répondant à un cahier des charges spécifié (structuration, emploi de contrôles, gestion de l'apparence, ergonomie, gestion des sollicitations, interaction avec fichiers,...)

**Contenu de l'AA Algorithmie**

- Notions des structures de données usuelles : liste (doublement) chaînée/circulaire, ensemble, pile, file, dictionnaire, arbre, tas, graphe
- Algorithmes de manipulation des structures de données usuelles (ajout, suppression, modification)
- Algorithmes de tri : insertion, bulle, rapide, fusion, par tas
- Notions de récursivité (terminale, mutuelle, ...)
- Notions de la théorie des graphes (parcours, recherche de chemin, coloration)

### Contenu de l'AA Programmation avancée - modélisation

- UML

### Contenu de l'AA Programmation avancée - théorie

- Programmation événementielle
- Programmation objet

### Contenu de l'AA Programmation avancée - travaux pratiques

#### Apprentissage du langage Python dans le paradigme POO :

- Présentation et familiarisation à la modélisation UML
- Manipulation des contrôles principaux (python)
- Présentation et familiarisation au concept de "programmation événementielle"
- Présentation et familiarisation au concept de "programmation Orientée Objet"
- Réalisation de petits projets avec interfaces graphiques qui permettent de se confronter aux concepts cités ci-dessus. (Exemples de réalisations: calculatrice, éditeur de texte, gestion d'une clientèle basée sur un fichier, calendrier, jeu de société, ...)

### Méthodes d'enseignement

**Algorithmie** : cours magistral, travaux de groupes, approche interactive

**Programmation avancée - modélisation** :

**Programmation avancée - théorie** : cours magistral, approche par projets, approche interactive, approche par situation problème

**Programmation avancée - travaux pratiques** : travaux de groupes, approche par projets, approche avec TIC, utilisation de logiciels

### Supports

**Algorithmie** : copies des présentations, syllabus, notes de cours

**Programmation avancée - modélisation** :

**Programmation avancée - théorie** : syllabus

**Programmation avancée - travaux pratiques** : notes d'exercices, protocoles de laboratoires

### Ressources bibliographiques de l'AA Programmation avancée - modélisation

- Desmet Erwin, Notes de cours «Programmation », HEH - Campus Technique, 2023.

### Ressources bibliographiques de l'AA Programmation avancée - théorie

- Desmet Erwin, Notes de cours «Programmation avancées », HEH - Campus Technique, 2023.

### Ressources bibliographiques de l'AA Programmation avancée - travaux pratiques

- Chapelle Joakim, Depreter Johan et Desmet Erwin , Notes de cours «Programmation avancées », HEH - DST, 2023.

<b>Évaluations et pondérations</b>	
<b>Évaluation</b>	Note globale à l'UE
<b>Langue(s) d'évaluation</b>	Français, Anglais
<b>Méthode d'évaluation</b>	<p>En ce qui concerne la notation totale de cette UE, nous utiliserons la règle suivante :</p> <ul style="list-style-type: none"> <li>• Partie labo réussie mais pas le reste : 3/20</li> <li>• Partie théorique réussie mais pas le labo : 3/20</li> </ul> <p>Réussite de l'EI (réussites des 3 AA) -&gt; note pondérée (/20) selon la répartition ci-dessous :</p> <p>- Epreuve d'examen pour les 3 AA : Algorithmique 25 % - POO 20 % - Laboratoire 45%</p> <p>- Evaluation continue 10 % non rejouable.</p>
<b>Report de note d'une année à l'autre pour l'AA réussie en cas d'échec à l'UE</b>	
<p>Algorithmie : <b>non</b>            Programmation avancée - modélisation : <b>non</b>            Programmation avancée - théorie : <b>non</b>            Programmation avancée - travaux pratiques : <b>non</b></p>	

Année académique : **2024 - 2025**