



Enseignement secondaire classique
Classes supérieures – classes de 1 ^{re}
Section I
SCIPR – Science de la programmation
Programme

Langue véhiculaire :	français, allemand, anglais
Nombre de leçons :	3
Nombre minimal de devoirs par semestre :	2
Nombre minimal de devoirs oraux :	1 par an ¹
Dernière mise à jour par la CNES :	16/06/2025

Structuration des contenus (savoirs, savoir-faire)

N°	Notion	Contenu	Indications
1	Encapsulation	<ul style="list-style-type: none">Différence public, protected, private et package	Révision de 2 ^e
	Héritage	<ul style="list-style-type: none">Redéfinition de méthodesPolymorphismeClasses et méthodes abstraites	Révision de 2 ^e
2	Connexion à une base de données	<ul style="list-style-type: none">Bases de données	Connexion à une base de données MySQL ou SQLite avec JDBC en effectuant des requêtes à l'aide des classes Statement et PreparedStatement .
3	Gestion de versions décentralisée	<ul style="list-style-type: none">Source & Version control	Création et configuration d'un dépôt (p.ex. avec GitLab). Configuration d'un client (p.ex. Sublime Merge) Sont à traiter les opérations suivantes de clonage d'un dépôt, d'ajout de fichiers et de modification de fichier à l'aide des commandes commit , fetch , pull et push .
4	Algorithmes de tri	<ul style="list-style-type: none">Tri par sélection directeTri par bulles	Sont à trier les éléments d'une liste suivant un critère donné soit par ordre croissant, soit par ordre décroissant.
	Interface Comparable	<ul style="list-style-type: none">Utilisation de la méthode de tri prédéfinie	Implémentation de l'interface Comparable afin de pouvoir utiliser la méthode de tri prédéfinie Collections.sort() dans le but de pouvoir mesurer et comparer plus facilement par la suite

¹ Instruction ministérielle ES 2019-4 du 6 novembre 2019 concernant l'organisation des épreuves orales en classe de première

		<ul style="list-style-type: none"> • Spécification du type de l'objet à comparer en utilisant la généricité de l'interface 	les complexités temporelles de différentes implémentations d'algorithmes de tri.
	Algorithmes de recherche	<ul style="list-style-type: none"> • Recherche linéaire • Recherche dichotomique • Recherche du minimum et maximum 	Recherche d'une position ou d'un ou plusieurs objets suivant un critère donné.
	Complexité	<ul style="list-style-type: none"> • Complexité temporelle et spatiale • Notations de Landau (« Big O ») • Mesure empirique de la complexité temporelle d'un algorithme connu ou donné 	Il s'agit ici d'une introduction à la complexité temporelle et spatiale. En ce qui concerne la notation de Landau, il s'agit de familiariser les élèves à cette notation répandue sans entrer dans les détails. Sont à interpréter le cas optimal, moyen et pire des cas.
5	Format d'échange de données	<ul style="list-style-type: none"> • JSON 	Introduction à la sérialisation de données sous format JSON. Pour les entrées et sorties des libraires externes sont à utiliser. Exemples : GSON (https://github.com/google/gson)
6	Java Collection Interface	<ul style="list-style-type: none"> • Utilisation des classes HashSet et HashMap • Rendre les élèves attentifs au fait qu'il existe un choix entre plusieurs structures de données et que celles-ci offrent des opérations similaires avec différentes complexités • Savoir choisir judicieusement la meilleure implémentation pour une problématique donnée 	Sont à utiliser les implémentations HashSet et HashMap avec courte introduction aux fonctions de hachage à l'aide de la méthode hashCode() de la classe String . Ces classes implémentent les interfaces Set et Map . Il est à porter à connaissance des élèves que les structures prédéfinies dans Java sont accessibles tous de la même manière sans rentrer dans les détails de ce qu'est une interface.
	Listes chaînées	<ul style="list-style-type: none"> • Structures linéaires (liste chaînée simple) • Implémentation de l'interface Comparable 	<p>Réaliser une structure linéaire où chaque objet pointe vers un autre objet du même type. Implémentation des opérations de parcours, d'ajoute en queue, d'ajoute en tête, d'insertion à un endroit précis (p.ex. après un élément donné), de remplacement, de suppression d'un élément donné, de recherche, de détermination du nombre d'éléments (taille) et de test si la liste est vide.</p> <p>Implémentation de l'interface Comparable afin de pouvoir comparer plus facilement des objets dans l'implémentation de certaines méthodes des listes chaînées.</p> <p>Application de la liste chaînée pour programmer par exemple une pile (stack) et une file (queue).</p>
7	Récursivité	<ul style="list-style-type: none"> • Récursivité 	<p>Introduction au concept de la récursivité avec implémentation d'algorithmes simples (pgcd, factorielle ...).</p> <p>Comparaison de la complexité temporelle entre un algorithme itératif et sa version récursive.</p> <p>Application de la récursivité pour résoudre des problématiques réelles simples (p.ex. afficher tous les fichiers dans une structure de dossiers ou rechercher un fichier dans une structure de dossiers).</p>

Indications didactiques et méthodologiques

- L'environnement de développement NetBeans est recommandé pour le développement d'interfaces graphiques.
- Le logiciel Unimozzer est recommandé pour tester des classes et pour représenter les diagrammes de classe sous forme UML.
- L'accent est mis sur le travail autonome et réflexif des élèves : Pour chaque exercice en Java, l'élève est invité à modéliser lui-même les classes en choisissant et justifiant :
 - Les attributs et types d'attributs
 - Le constructeur avec paramètres éventuels
 - Les méthodes avec type de retour et paramètres éventuels
- L'apprentissage des méthodes de débogage fait partie intégrante du cours
- Les élèves sont tenus à se conformer aux conventions usuelles du langage de programmation (dénominations des éléments, présentation du code, indentations...)
- Dans la mesure du possible, les élèves s'informent en ligne, en consultant des manuels de logiciel et d'outil (p.ex. : JavaDoc, site et tutoriels Oracle...)
- Dans la mesure du possible, le chapitre « *Connexion à une base de données* » est à traiter en étroite collaboration avec la discipline « *Analyse et modélisation d'informations* ».

Il est essentiel pour la mobilité et l'équité des étudiants, le concept pédagogique, et l'appropriation des connaissances acquises, que les élèves soient équipés de leur propre ordinateur portable, afin de leur permettre de poursuivre leur travail en dehors des heures de cours.

Modalités de l'évaluation formative & sommative

Évaluation formative :

- Accompagnement des élèves dans la conceptualisation et la réalisation des productions
- Dialogue pédagogique au sujet de :
 - Tâches et activités des élèves
 - Rapports et résultats de travaux de groupe et individuels
- Évaluation par les pairs

Évaluation sommative :

- Devoirs en classe écrits, oraux et/ou pratiques (épreuves intégrées)
 - Peuvent tenir lieu de devoirs en classe, des réalisations pratiques, travaux de grande envergure ou en groupe
 - Évaluation des contenus de « défis » selon les consignes définies dans le cadre de la MAIOU
- Lors de l'évaluation sommative le processus de création et sa mise en perspective peuvent l'emporter sur le résultat final du projet.

Matériel pédagogique, ...

Matériel informatique, logiciels et ordinateur personnel