

**Programme de la
Discipline Informatique
Premier cycle INSA de Lyon
2014-2015**

PCC1A au premier semestre :

Prise en main et fonctionnement de l'ordinateur

Postes de travail et supports numériques (compétences « C2I »)

4 heures de CM + 14 heures

Connaissances :

- Environnement numérique de travail
- Éléments juridiques sur la propriété intellectuelle et le respect de la vie privée. Questions éthiques.
- Connaissances des outils de bases du travail collaboratif (Wiki, forum)

Questions abordées :

- Conduite de recherches documentaires, questions juridiques (notions)

Compétences :

- *Compétences D1 à D5*

Connaissance de l'ordinateur

2 heures CM

Connaissances :

- Structure et fonctionnement du processeur
- Principe de fonctionnement d'un système d'exploitation

Questions abordées :

- Expliquer aussi simplement que possible comment fonctionne un ordinateur sans entrer dans des détails techniques

Représentation et codage des nombres

1heure CM+ 4 heures TD

Connaissances :

- Principe de la représentation binaire, opérations arithmétiques en binaire, changement de base, structuration en octets et en mots.

Questions abordées :

- Élaboration du lien entre la théorie et la pratique manipulative des objets numériques (codage des nombres, débordements possibles, troncatures)

Compétences :

- *Manipuler le codage binaire, hexadécimal des nombres, maîtriser le changement de base*

PCC1A au 1^{er} puis 2nd semestre: Algorithmique JAVA

4CM + 29 heures TD

Représentation des variables

Connaissances :

- Types primitifs : types numériques, caractères, booléen.
- Types non primitifs : Chaînes de caractères, tableaux

Compétences :

- *Structurer les informations, choisir des types de données appropriées*

Instructions de base

Connaissances :

- Exprimer des conditions composées, des conditionnelles en cascade
- Séquences d'instructions répétitives et boucles (exemples)
 - Rechercher un élément dans un tableau
 - Calculer la limite d'une suite, réaliser une approximation numérique (à epsilon près)

Questions abordées :

- Sensibilisation aux questions de « schémas » algorithmiques

Compétences :

- *Comprendre un algorithme et expliquer ce qu'il fait*
- *Concevoir une solution algorithmique à un problème, savoir l'implanter (programmer) et savoir l'analyser*
- *Modifier un algorithme existant pour obtenir un résultat différent*

Développement sur machine

Connaissances:

- Affectation, séquence, test, boucle, déclaration.
- Opérateurs numériques, booléens, de chaînes de caractères (concaténation).
- Méthodes, main.
- Allocation mémoire et types de données primitifs et non primitifs.
- Passage de paramètres : référence, valeur.
- Principes des langages compilés et interprétés. Création des .java et des .class

Questions abordées :

- Recherche d'un « style » de programmation correct : exigences de clarté, exactitude, respect du guide de syntaxe java...
- Valorisation de la lisibilité des productions (commentaires, nommage, modularité, documentation, réutilisation de composants)
- Développement de l'autonomie dans la construction du programme : choix des méthodes et de leur entête.

Compétences :

- *savoir programmer un algorithme décrit en langue naturelle et comprendre un programme*
- *choisir un type de donnée en fonction d'un problème à résoudre*
- *concevoir l'entête d'une fonction (méthode), puis la fonction elle-même*
- *documenter une fonction, un programme*
- *mettre un programme au point en le testant*

Introduction à la Programmation orientée objet (POO)

2CM + 10 heures TD

Connaissances :

- Classes simples
 1. Classes, objets, encapsulation (public/privé)
 2. Données membres : attributs de classes et méthodes
- Algorithmique avancée sur des structures de données complexes (listes chaînées)

PCC2A au 1^{er} et 2nd semestre (année transitoire)

Programmation modulaire avec IDE et résolution numérique

6 heures TD

Connaissances :

- Calcul numérique
- Prise en main d'une IDE avec debugueur

Compétences (simplifiées) :

- *Etre capable de résoudre un problème numérique*
- *Être capable de concevoir une application avec une IDE*
- *Être capable d'utiliser un debugueur*

Programmation orientée objet (POO)

8CM +16 heures TD

Connaissances :

- Classes simples
 1. Classes, objets, encapsulation (public/privé)
 2. Données membres : attributs de classes et méthodes
- Algorithmique avancée sur des structures de données complexes (listes chaînées)
- Concepts avancées de classes
 1. polymorphisme
 2. héritage

- Programmation événementielle et classes graphiques (AWT/SWING)
 1. implémentation des écouteurs d'événements
 2. conception d'une IHM complète simple avec (à partir de Jdeveloper ou Eclipse) et sans IDE (Geany)

Compétences (simplifiées) :

- *Etre capable de résoudre numériquement un problème numérique*
- *Être capable de construire une classe complète sur une simple spécification textuelle*
- *Être capable de construire une hiérarchie complète à plusieurs niveaux*
- *Être capable de concevoir une application graphique et maîtriser la programmation par événements*
- *Être capable de concevoir une solution algorithmique avancée*
- *Etre capable de prendre en main un nouveau langage de programmation*

Applications graphiques

2CM +6 heures TD

Connaissances :

- Gestion des objets visuels et conception d'applications interactives (gestion du mouvement, de timer, des interactions)

Compétences (simplifiées) :

- *Être capable de concevoir une application graphique et d'interagir avec elle*

PCC2A au 2nd semestre

Découverte d'un nouveau langage de programmation

4 heures TD

Compétences :

- *Etre capable de prendre en main un nouveau langage de programmation*

Projets de développement orientés objet

14 heures TD

Connaissances :

- Pour les projets pluridisciplinaires (liées au projet de BQF ou en association avec d'autres disciplines) : connaissances spécifiques de la discipline associée

Compétences :

- Concevoir une application complète (des spécifications au test final) en groupe de 4 étudiants
- S'adapter à des problèmes ouverts pluridisciplinaires sans a priori

- Rechercher de la documentation adaptée (programmation android, importation d'une librairie spécifique au projet...)
- Réutiliser des codes sources de librairies importées
- Créer un SVN et partager les versions du projet (optionnel en PCC)
- Réinvestir les connaissances informatiques spécifiques et compétences du premier semestre
- Communiquer sur son travail au sein du groupe et vers les autres (démos et présentation de fin d'année)