



Introduction à la Programmation (CS-107)

Petit tour d'horizon...

1 Introduction

Ce petit document a pour but de vous informer sur la pédagogie du cours "Introduction à la programmation" (CS-107), son mode de fonctionnement et sur divers autres aspects liés à son organisation.

2 Un cours pourquoi ? pour qui ?

Deux constats habituels au sujet de ce cours :

- c'est un cours qui suscite souvent, et à juste titre, de fortes attentes,
- mais, qui s'adresse à des étudiants ayant des niveaux très divers en programmation¹

L'idéal voudrait donc que ce cours puisse répondre aux exigences des débutants comme des plus forts. Exercice plutôt «sportif» s'il en est ...

Pour que les choses soient clairement dites dès le départ : ce cours est en premier lieu **organisé pour les débutants** et ne demande pas de connaissances préalables en informatique ; ce qui ne veut pas dire qu'il ne soit pas exigeant !

L'objectif premier est de faire acquérir à tous les étudiants **une base commune en programmation**, et en particulier en programmation orientée objet. Cette dernière est une forme de programmation considérée comme à la fois naturelle et rigoureuse. Le but est de permettre ensuite aux étudiants de bien aborder les projets qu'ils devront mener dans la suite de leurs études à l'EPFL.

Les notions fondamentales vues au semestre d'automne seront consolidées au semestre de printemps, notamment par la réalisation d'un projet. Au terme de ces deux semestres, chaque étudiant sera capable

¹ eh oui ... contrairement à un futur ingénieur en génie civil qui n'a eu que peu de chances de construire un pont dans sa chambre quand il était au gymnase, la disponibilité des moyens techniques fait que les futurs ingénieurs en informatique ou en systèmes de communication ont parfois commencé à «jouer» très tôt avec la programmation !

de manipuler aisément un ordinateur et de programmer de manière indépendante. Les concepts de base de la programmation orientée objet seront par ailleurs acquis.

Certains pourraient se dire, mais alors "ce cours ne me sert à rien si je programme déjà depuis le berceau !". Si vous êtes dans ce cas, il est vraisemblable que vous bailliez aux corneilles lorsque nous vous présenterons doctement ce qu'est un **if-then-else**.

Cela dit, la difficulté des notions abordées croît assez rapidement en cours de semestre. Par ailleurs le cours met l'accent sur les concepts plutôt que sur les détails techniques. Le but étant de fournir les bases formelles qui font souvent défaut dans un apprentissage autodidacte et ainsi de **consolider l'acquis des non-débutants**. A part la programmation en tant que telle, le cours vous permettra également de mettre en pratique certaines notions plus théoriques acquises dans le cadre du cours AICC (la notion de complexité par exemple).

Attention, même si vous êtes déjà à l'aise en programmation, **il demeure important de bien connaître le contenu du cours** (transparents, genre d'exercices, jargon utilisé, concepts abordés) pour mener à bien les **tests**.

Le projet du deuxième semestre vous donnera l'occasion de montrer vos compétences et de vous faire plaisir en programmant des applications de plus grande envergure.

Et Java ?

Au risque de décevoir certains d'entre vous, ce cours n'a pas pour objectif principal de vous transformer en expert Java « hightech ». De nombreux cours et tutoriels, référencés sur le site du cours s'y appliquent déjà.

Le but est plutôt :

1. d'enseigner les notions fondamentales communes à la plupart des langages objet généralistes,
2. et de les illustrer au moyen de Java.

Les notions sont par ailleurs abordées comme si Java était votre premier langage de programmation (aucune expérience n'est supposée, ni en Java, ni en d'autres langages). A noter qu'une telle démarche n'est pas originale et que la présentation du matériel sous cette forme se retrouve dans de nombreux livres sur les bases de Java (voir références sur la page web du cours).

Cette approche fait que les aspects plus spécifiques à Java ne sont délibérément abordés que tardivement dans la progression du cours : l'API de Java, c'est à dire tout le matériel prédéfini que Java met spécifiquement à disposition du programmeur sont vus qu'en fin de semestre et/ou au semestre de printemps.

Cela dit, au terme de ce cours vous aurez néanmoins acquis tout le bagage de base nécessaire pour vous transformer en « gourou » Java si le coeur vous en dit.

A part Java ...

Ce cours a aussi pour objectif de vous sensibiliser à l'importance des aspects *methodologiques* liés à la programmation. Ainsi, s'il s'intéresse en priorité au **quoi** (les concepts : structures de contrôle, modularisation, abstraction et encapsulation, polymorphisme, ...) et au **pourquoi** (la raison de leur existence), il va également vous donner des pointeurs sur le **comment** (algorithmique, règles et conseils pour produire de bon programmes etc ...). Les cours de programmation de première année ne sont ni des cours d'algorithmique, ni des cours de génie logiciel, mais ils vous donneront un avant-goût de ces disciplines fondamentales.

Pour remplir les objectifs multiples de ce cours, les outils pédagogiques suivants sont mis à votre disposition :

- La page Moodle du cours : <https://moodle.epfl.ch/course/view.php?id=14847> sur laquelle vous trouverez, au fil des semaines, les transparents du cours, les séries d'exercices et leur corrigé ainsi que des références utiles pour Java et la programmation de façon générale. **Il s'agit du point d'entrée principal pour toutes les ressources utiles et il est important d'en consulter systématiquement le contenu.**
- Du matériel en ligne (vidéos, quizzes), accessible via deux MOOCs² complétant les ressources classiques du cours.
- Un forum de discussion Ed (accessible aussi depuis le site Moodle). Ce forum vous permettra d'interagir avec l'équipe du cours, notamment pendant les TP en ligne et aussi de poser vos questions sans avoir à attendre les séances « officielles ».
- Des informations générales sur l'environnement de programmation.

Important : c'est la page Moodle du cours qui sert de point d'entrée à toute la documentation nécessaire (transparents, exercices, ressources du MOOC etc). N'allez donc **pas chercher** cette documentation en partant **d'ailleurs** (comme le site de Coursera qui héberge les MOOC par exemple).

Ces différents outils sont décrits plus en détail dans ce qui suit.

3 Nous

Le cours « Introduction à la Programmation » est assuré cette année par :

- Jamila Sam, enseignante de la faculté I&C, en charge par ailleurs de cours de programmation (en C++ cette fois) à vos collègues de la section des sciences de la vie.
- Rafael Pires, enseignant de la faculté I&C, en charge également du Cours ICC au semestre de printemps.

Le cours CS-107 sert de pré-requis au cours "Pratique de la programmation orientée-objet" (CS-108), enseigné au semestre de printemps par Michel Schinz.

Plusieurs assistants-doctorants participent au cours. Ils seront aidés par 18 assistantes.s-étudiante.s en informatique et en systèmes de communication.

Cette équipe dévouée est à votre disposition pour vous aider à progresser en programmation. Vous êtes encouragés à profiter pleinement des ressources disponibles et à **poser toute question vous semblant nécessaire**.

Les rôles respectifs de chacun sont définis comme suit :

- Jamila Sam donne la plupart des cours et est responsable de la logistique générale, de l'organisation des séances de travaux pratiques et des tests, ainsi que du mini-projet noté (énoncés, transparents, organisation).
- Rafael Pires est responsables du mini-projet auto-noté (énoncés, transparents et organisation) et donnera quelques cours.
- Les assistants-doctorants sont responsables du déroulement des séances d'exercices ainsi que de diverses autres tâches. Ils répondront à vos questions pendant les séances bien sûr, mais aussi sur le forum de discussion.
- Les assistants-étudiants répondront à vos questions lors des séances d'exercices et sur le forum Ed.

La liste des complètes des assistants étudiants et doctorants est disponible sur le site Moodle du cours.

Comment nous contacter ?

- Pour toute question sur le cours, les séries ou d'ordre général : le forum Ed du cours accessible via le site moodle du cours (<https://moodle.epfl.ch/course/view.php?id=14847>).

2. MOOC pour "Massive Online Open course"

- Pour toute question administrative relative au cours (intégration au cours des étudiants hors SIN et SSC, organisation de tests de rattrapage etc.) : Jamila Sam (jamila.sam@epfl.ch, INR235, tel. 693 52 09).
- Tout problème relatif aux mini-projets : Barbara Jobstmann (cs107@epfl.ch) et Jamila Sam (cs107@epfl.ch)
- Tout problème sur le contenu du matériel pédagogique ou le déroulement des TP : les assistants-doctorants.

4 Vous

Le cours est obligatoire pour tous les étudiants en 1ère année SIN et SSC ainsi que pour certains étudiants d'échange. Les étudiants d'autres sections et de l'UNIL sont les bienvenus sous condition de se présenter à nous au début du semestre.

Si vous redoublez et que vous avez des doutes sur quels cours devoir suivre, veuillez consulter Madame Martine Emery (martine.emery@epfl.ch) ou Madame Cécilia Bigler (cecilia.bigler@epfl.ch).

5 Comment le cours est-il donné ?

Le cours CS-107 est enseigné sous la forme de cours ex cathedra (projeté en visio-conférence) et de séances d'exercices (en présentiel). Durant 10 semaines du semestre, **il sera couplé à deux cours en ligne** (des "MOOCs", pour Massive³ Open Online Course).

Les paragraphes suivants vous expliquent le rôle respectif de chacun de ces éléments pédagogiques.

5.1 MOOCs

Les deux MOOCs utilisés seront les suivants :

- Initiation à la programmation (en Java) :
<https://www.coursera.org/learn/initiation-programmation-java/>
pendant les 5 premières semaines de cours.
- Introduction à la programmation orientée-objet :
<https://www.coursera.org/learn/programmation-orientee-objet-java/>
depuis la semaine 7 du cours.

Ces deux MOOCs sont évidemment ouvert à une plus large audience que celle de l'EPFL.

Pour CS-107, le rôle de ces MOOCs sera de fournir un certain nombre de ressources supplémentaires (vidéos, quizzes et devoirs) qui seront exploitées durant 10 semaines du calendrier.

Pour accéder à ces ressources, il suffit de vous inscrire.

La démarche suggérée pendant les semaines couplées aux MOOCs est la suivante :

- visionner les vidéos suggérées par le calendrier, parcourir rapidement les transparents fournis et faire les quizzes relatifs avant de venir en cours ;
- commencer les exercices faciles (niveau 0 et 1, voir explications plus bas) avant de venir en TP ;
- assister au cours ex-cathedra (soit en présentiel soit en ligne), qui sera en général moins long, et dont le rôle sera de donner un résumé des concepts⁴ vus dans les vidéos ainsi que quelques approfondissements ;

3. une moyenne de 20'000 inscrits par MOOC en général

4. il faut pour cela que vous les ayez vus au travers des vidéos ou transparents

- participer aux séances d'exercices en salle de TP (qui impliqueront entre autres de programmer les devoirs du MOOC).

A noter que cette démarche s'applique principalement à ceux d'entre vous qui débutent avec la programmation. Pour ceux qui ont déjà des bases, l'étude des transparents et la participation aux séances d'exercices peut suffire.

5.2 BOOC

La démarche décrite ci-dessus s'applique principalement à celles et ceux d'entre vous qui débutent avec la programmation. Si vous avez déjà des bases, le cours fournit comme supports alternatifs :

- des BOOCs : il s'agit de document pdf multi-média résumant les MOOCs et permettant d'accéder à des morceaux choisis des vidéos ;
- des transparents compilant de façon complète le contenu du cours (hors compléments).

Libre à vous de choisir le support qui semble le plus adapté à votre niveau. La vérification des acquis au travers des séances d'exercices est très fortement recommandée.

Les BOOCs constituent par ailleurs un bon moyen de réviser pour tous en vue de l'examen final.

5.3 Cours ex cathedra

Les cours ex cathedra réunissent les étudiants dans un auditoire (et seront projetés simultanément en visio-conférence). Ils durent :

- une à deux heures pendant les semaines couvertes par le MOOC ;
- et deux heures complètes pour les quelques séances non couplées à l'un des deux MOOCs.

Pendant les semaines couplées aux MOOCs, le rôle des séances ex-cathedra est simplement de résumer les concepts présentés dans les vidéos et d'aborder quelques approfondissements.

L'examen et les mini-projets porteront sur les matières qui sont enseignées en cours ex-cathedra (avec apport des vidéos). Il est important de connaître le matériel et le style d'enseignement pour arriver à bien se préparer.

Afin de faciliter la tâche des étudiants qui désirent suivre le cours à distance, ainsi que la prise des notes des étudiants présents, les ressources seront disponibles sur le site web du cours quelques jours avant le cours en question. Des compléments oraux, sous formes d'exemples additionnels ou de discussions, peuvent être apportés aux transparents pendant le cours ex-cathedra ou sur le forum de discussion du cours.

5.4 Séances d'exercices

Chaque cours est complété d'une séance d'exercices où les notions théoriques seront mises en pratique. Il est prévu trois heures d'exercices par semaine, mais le temps requis pour résoudre les exercices peut varier, parfois considérablement, en fonction des connaissances préalables et de la préparation de l'étudiant.

Les séances d'exercices sont entièrement pratiques. Elles se déroulent dans une salle d'ordinateurs ou en ligne.

Plusieurs assistants sont présents pendant la séance pour vous aider. Ils répondront à tout type de question. Vous êtes encouragés à discuter de vos programmes et de vos problèmes éventuels avec les assistants, même si le but n'est pas qu'ils fassent la totalité de la série avec vous ! Ils ne s'imposent pas, mais attendent d'être sollicités par les étudiants. Profitez pleinement de leur présence !

Plus d'informations sur le déroulement des séances d'exercice (notamment pour la partie en ligne) seront données lors du premier cours.

Pour chaque séance, il y a une série d'exercices à résoudre de manière indépendante. L'énoncé sera mis à disposition sur le site Moodle du cours une semaine avant le cours en question. Le corrigé sera également mis à disposition (mais un peu plus tard).

Remarque importante : Il est vivement recommandé d'être présents aux séances d'exercices.

- cela vous permet d'assurer la régularité de votre progression qui est une clé essentielle de la réussite au cours,
- cela vous permet de bénéficier de l'aide des assistants,
- et cela nous permet de vous connaître, ce qui peut avoir une certaine importance si l'on doit justifier vos compétences en conférence des notes au terme de l'année.

5.5 Charge de travail

Le cours CS-107 prévoit deux heures de cours et trois heures d'exercices (ce qui implique une charge totale hebdomadaire de 10 heures en comptant le travail personnel).

Pendant les semaines couplées aux MOOCs, la répartition de ces heures telle qu'anticipée sera la suivante :

- 1 heure de cours ex-cathedra : récapitulation et approfondissements ;
- 3 heures d'exercices en salle de TP ;
- 5 à 6 heures hors séances de contact :
 - 1 :30-1 :45 sur les vidéos de la semaine suivante
 - 0 :15-0 :30 sur les quizzes de la semaine suivante
 - 3 à 4 heures pour préparer la série d'exercices de la semaine à venir, finaliser celle de la semaine précédente (ou faire les mini-projets) et faire les devoirs du MOOC.

5.6 Séances de soutien

La section IC vous offre gracieusement le budget pour quelques heures de soutien. Ces appuis visent à apporter de l'aide et à redonner les explications complémentaires nécessaires à celles et ceux d'entre vous qui en ont le plus besoin. Deux à quatre assistants-étudiants y seront présents chaque semaine. A noter que ces heures ne sont pas une prolongation systématique du TP. Elles sont en priorité réservées à des compléments d'explications sur des points du cours qui ne seraient pas bien compris. **Il vous sera demandé de vous inscrire à ces appuis en début de semaine afin que nous puissions anticiper les besoins en encadrement.**

6 Choix pédagogiques

Le principe pédagogique fondamental de ce cours est de **donner à chacun les moyens de progresser à son niveau**, et ce, en raison de la grande hétérogénéité de vos niveaux en programmation.

Ce principe a conduit à introduire plusieurs éléments :

- un **accès diversifié** au contenu : transparent du cours, vidéos, quizzes, exercices, fiches résumé, mini-références, livre conseillé et références externes (« en ligne » et bibliographiques) sont les différents supports mis à disposition ;
- un **accès hiérarchisé par niveau** des élèves : 4 niveaux pour les exercices.

6.1 Catégorisation des exercices par niveaux

Les séries d'exercices sont en général surdimensionnées : elles contiennent souvent plus de matériel qu'il n'est faisable en trois heures, pour un programmeur débutant.

Considérez par ailleurs les trois heures de pratique à l'emploi du temps comme des heures d'assistance : nous sommes là pour vous aider. **Organisez donc votre travail de sorte à faire chez vous les choses qui vous semblent abordables et réserver les aspects difficiles/ les questions pour les séances pratiques du vendredi.** Pensez aussi à utiliser le forum Ed du cours y compris en dehors des séances du vendredi (voir plus bas).

Dans le but de vous aider à choisir, les exercices sont catégorisés par niveau de difficulté (de 0 à 3).

- **Niveau 0** : Reprise pas à pas d'un exemple du cours. Les exercices de ce niveau ne sont pas systématiquement présents. Il peuvent sans problème être sautés par tous ceux qui estiment avoir une suffisamment bonne compréhension de la programmation de base et du cours.
- **Niveau 1** : Ces exercices élémentaires devraient pouvoir être faits par tous dans un temps raisonnable (30 à 45 minutes maximum). Ils permettent de travailler les bases.
- **Niveau 2** : Ces exercices plus avancés devraient être abordés par tous, sans forcément être finis. La reprise de l'exercice avec la correction devrait constituer un bon moyen de progresser.
- **Niveau 3** : Ces exercices d'un niveau avancé sont pour les plus motivés/habiles d'entre vous. Ils n'existent pas systématiquement dans chaque série. Ils peuvent *dans un premier temps* être ignorés mais doivent être repris, si nécessaire avec la correction, lors des révisions.
- **Les devoirs du MOOC** : leur note ne comptera pas pour vous, mais il s'agit de matériel d'entraînement aux tests qu'il est vivement recommandé de pratiquer.

Notez que les niveaux sont déterminés en fonction du moment où la série est prévue. Il est clair qu'un même exercice donné plus tard au cours de l'année (par exemple au moment du test de fin de semestre) serait considéré comme plus facile !

Les séries ne sont pas notées et il ne vous est pas demandé de les rendre. Elles sont à considérer comme une aide à l'apprentissage et comme une préparation aux tests. Si vous n'arrivez pas à terminer une série d'exercices pendant la semaine, il est impératif d'en consulter le corrigé et en étudier les détails lorsque vous vous préparez aux tests.

Si, par contre, il vous reste du temps, complétez la série par un peu de curiosité et d'expérimentation personnelle : ajoutez à votre gré d'autres fonctionnalités à vos programmes, consultez la documentation, etc. En règle générale, toute manipulation sérieuse sur l'ordinateur augmentera vos connaissances et facilitera votre séjour à l'EPFL.

7 Eléments d'auto-évaluation et éléments notés

Le cours vous propose à mi-semestre un mini-projet dont la note ne compte pas mais qui vous permet de vous auto-évaluer.

Les connaissances que vous avez acquises lors de ce cours seront quant à elles évaluées à l'aide d'un mini-projet noté (à coder sur le dernier mois du semestre) et d'un examen écrit qui aura lieu lors de la dernière semaine du semestre. Le mini-projet noté se fait à deux. L'examen est individuel.

Le calcul de vos notes se fait avec les pondérations suivantes :

- 60% pour le mini-projet noté ;
- et 40% pour l'examen

Notez que l'examen comportera des questions relatives aux mini-projets.

La moyenne est transmise au service académique arrondie au quart de point le plus proche.

Chacun des modules de programmation de cette année, à savoir, "Introduction à la Programmation" (CS-107, semestre automne) et "Pratique de la programmation orientée-objet" (CS-108, semestre de printemps) vous vaudra une note séparée.

Importants : les devoirs notés du MOOC servent d'exercices supplémentaires qu'il est vivement recommandé de faire. Il n'intervient cependant pas dans l'évaluation de CS-107.

7.1 Mini-projets

Les mini-projets ont pour but de démontrer que l'outillage de programmation dont vous disposerez vous permettra rapidement de vous attaquer à la *résolution de problèmes concrets*.

À la mi-semestre, un **mini-projet en auto-évaluation** (dont la note ne compte pas) vous sera proposé. Il vous permettra d'acquérir les compétences attendues sur les bases de la programmation non orientée-objet et de vous exercer à travailler en binôme au moyen d'outils de gestion de projets. Un correcteur automatique, que vous pourrez lancer autant de fois que souhaité, vous attribuera une «note» sur la réalisation de ce mini-projet et vous permettra ainsi d'**auto-évaluer votre niveau**. Le but à vous fixer est que votre groupe atteigne au moins la moyenne dans les 2 semaines imparties pour réaliser le mini-projet. Ce dernier comportera un petit challenge et les dix premiers groupes à le résoudre recevront une modeste récompense symbolique :)

Une fois la date butoir atteinte, vous pourrez connaître votre classement dans la volée (basé sur le nombre de points atteints lors de la dernière soumission en date).

Important : ni le nombre de points ni le classement du mini-projet en auto-évaluation ne seront utilisés par l'équipe du cours. Il s'agit simplement d'une information **pour vous**, afin que vous puissiez estimer votre niveau (compétence, implication, réussite de travail en équipe) à mi-parcours du cours. **Il est donc fondamental, pour que l'exercice soit profitable, que vous réalisiez vous même le travail de bout en bout.**

Le **mini-projet** noté sera quant à lui complètement corrigé par l'équipe enseignante il portera sur tous les concepts du cours avec un accent particulier sur la programmation orientée-objet. Vous aurez quatre semaines pour le réaliser.

7.2 Examen écrit

L'examen écrit se déroule dans des auditoriums pour tous les étudiants en même temps.

Tout ce qui a été enseigné aux cours (complétés des vidéos), ainsi que les séries d'exercices peuvent faire l'objet de question. Pour vous préparer, lisez et assimilez tous les transparents (qui résument le contenu des vidéos et apportent certains compléments), révisez toutes les séries d'exercices et étudiez les solutions proposées dans les corrigés. Vous pouvez également consulter les énoncés et les corrigés du test de l'année précédente sur le site web du cours, tout en sachant que la matière enseignée peut changer chaque année. Les feuilles blanches pour répondre au test seront fournies. Vous devez rédiger de façon lisible au stylo indélébile noir ou bleu. Vous pouvez apporter une quantité raisonnable de documents aux tests, à savoir 1-3 livres de référence, les transparents du cours, les énoncés et les corrigés des séries, et une quantité raisonnable de notes de cours. Les documents jugés excessifs ou inéquitables peuvent être retirés lors du test. En cas de doute, demandez l'autorisation de l'assistant responsable avant le test.

Il n'est pas permis d'échanger des objets (documents, stylos, gommes, etc) entre étudiants pendant le test. Les équipements électroniques (à part les traducteurs pour les étudiants non-francophones) ne sont pas permis. Il n'est pas permis de sortir de la salle pendant le test sans avoir rendu sa copie. La carte d'étudiant sera contrôlée pendant le test.

En cas d'absence, le note de zéro est attribuée, sauf si vous présentez un certificat de maladie (copie) ou un ordre de marche. Dans ce cas, une **épreuve de rattrapage** sera organisé pour vous.

Le corrigé sera mis à disposition sur le site web du cours après l'examen. Votre copie corrigée sera consultable lors de séances programmées à cet effet. Les résultats préliminaires seront publiés sur le site web du cours. Vous aurez ensuite quelques jours pour faire vos éventuelles réclamations concernant les corrections. Après cette date, votre note pour le test en question sera définitive.

Les tentatives de fraude, même légères ou ratées, ne seront pas tolérées. Tous les étudiants impliqués dans la tentative auront 0 pour le test en question. Le règlement disciplinaire de l'EPFL est disponible au service académique.

8 Forum Ed

Le cours met donc à votre disposition un forum de discussion (*forum Ed* : <https://edstem.org/eu/courses/1526/discussion/>) accessible aussi depuis Moodle. Ce forum

permettra à ceux d'entres vous qui le souhaitent, de poser des questions sur le contenu du cours (au sens large) pendant toute la semaine.

Ainsi, si vous rencontrez des difficultés à résoudre un exercice ou à procéder à une installation nécessaire pendant la semaine, vous êtes vivement encouragés à nous décrire votre problème, si basique soit-il, pour que nous vous aidions à le surmonter. Il n'y a pas de question ridicule (vous pouvez d'ailleurs poster anonymement ;-)), et même si certains semblent nettement meilleurs que vous (ce ne sont d'ailleurs pas toujours ceux qui finissent le mieux l'année!), dites-vous bien que si quelque chose ne vous semble pas clair, cela l'est sûrement aussi pour la majeure partie de vos collègues. N'hésitez donc pas à poser la question sur le forum : ainsi tout le monde pourra profiter de la réponse.

Quelques remarques importantes au sujet du forum :

- des consignes d'utilisation du forum sont postées sur ce dernier en début de semestre. Lisez-les attentivement.
- le forum s'est vu souvent dénaturé de sa fonction prévue par les messages des "plus forts" qui, très tôt, mettent en avant leurs connaissances techniques en programmation, systèmes d'exploitation, langages de scripts etc. Du coup, les vrais débutants, qui sont ceux pour lequel le forum doit fonctionner en priorité, n'osent plus parler des bugs dans leur `HelloWorld.java` et restent silencieux comme la tombe jusqu'à la fin du semestre. Il est évidemment souhaitable que cela ne soit pas le cas! Donc, une consigne : n'hésitez pas à **poser vos questions, même si elles vous semblent stupides**. Vous vous rendrez compte que souvent votre problème a été rencontré avant vous et que vous pouvez aussi trouver de l'aide parmi vos camarades. Le but est de comprendre la matière... et il n'y a que le premier pas qui coûte!

Moodle héberge aussi un « forum administratif » nommé *Annonces du cours*. L'équipe enseignante utilisera ce forum pour poster des annonces relatives au déroulement du cours (modalités et salles d'examen par exemple). Vous **recevrez une copie de ces annonces** par email. En principe, vous n'aurez donc pas à intervenir souvent sur ce forum, sauf éventuellement au cas où vous souhaiteriez avoir des compléments d'informations par rapport à une de nos annonces.

Les MOOCs mettent aussi à disposition de ses utilisateurs des fora auquel vous êtes aussi bien sûr libre de participer et qu'il vous est vivement recommandé de consulter. Il est notamment recommandé de les utiliser pour les problèmes liés aux devoirs du MOOC ainsi que pour les problèmes d'installation.

9 Ordinateurs

Nous disposons pour les TP des salles informatiques INF1, INF2 et INF3 ainsi que de 3 salles d'appoint (INF 019, INF 213 et INM 203). Seule la INF3 dispose de machines (64 machines avec Ubuntu Desktop 22.04). Les salles INF1 à INF3 disposent de postes flottants auquel vous pouvez brancher vos propres ordinateurs portables et les connecter au réseau. Il est également possible d'utiliser vos ordinateurs pour vous connecter à distance à des machines virtuelles offertes par l'EPFL (Ubuntu Desktop 22.04 toujours).

Comptes informatiques et accès aux salles de TP Les détails des comptes (ordinateurs et email) vous ont normalement été distribués lors de la journée d'accueil ou envoyés à la maison si vous n'avez pas pu y participer. Les accès aux salles sont attribués d'office aux cartes des étudiants en SIN et SSC. Les autres étudiants devront en faire la demande individuellement. En cas de problèmes d'accès aux salles, par exemple si votre carte ne marche plus, il faut contacter la réception de la faculté I&C.

Chaque étudiant recevra une adresse email valable pour toute la durée de ses études à l'EPFL. L'adresse aura la forme `prénom.nom@epfl.ch`. Le cours de programmation vous apprendra seulement les bases d'UNIX/Linux nécessaires pour suivre le cours.

10 Documentation

Transparents, séries et divers supports

La documentation du cours comporte les transparents, les séries d'exercices, quelques livres de référence ainsi que des fiches résumé et des mini-références. Cette documentation est complétée par les ressources en relation avec les MOOCs (vidéos, quizzes en ligne et livres interactifs, les BOOCs édités par les PPUR). Nous utiliserons également de la documentation en ligne. L'intégralité de ces ressources est référencée depuis la page Moodle du cours.

Les documents et les fichiers seront disponibles quelques jours avant le cours en question. Ils sont entièrement révisés chaque année. N'utilisez pas les documents des années précédentes.

Important : **les énoncés des exercices sont découplés du matériel MOOC**. Ils constituent un sur-ensemble de ce dernier (prenant parfois une forme légèrement différente). Il vous est donc demandé de **ne pas utiliser les exercices du MOOC directement sur le site de Coursera** (les devoirs du MOOC seront référencés dans les séries d'exercices).

Fiches résumé et mini-références

Le but des fiches résumé est double : présenter de façon condensée ce qu'il faut connaître, puis (plus tard pour les programmeurs) avoir un accès très rapide à tel ou tel détail de syntaxe (une fois les concepts connus).

Les mini-références visent par contre à donner une information plus complète sur les aspects techniques du langage Java et offrent une présentation différente de la matière technique donnée dans les transparents du cours.

Livre sur Java

Vous pouvez compléter le matériel fourni en cours par un livre de référence sur Java. Des conseils d'acquisition et une petite bibliographie sont mis à votre disposition sur la page web du cours (lien "Ressources et Références", puis rubrique « Livres et tutoriaux divers » dans les ressources Java).

11 Le mot de la fin.. ou plutôt du début

Terminons cet aperçu du cours par un petit conseil un peu à contre-courant : ne soyez pas victimes de vos passions. Il a été souvent constaté avec regret qu'un nombre non négligeable des meilleurs étudiants des cours de programmation se retrouvaient à la MAN (sans doute pour n'avoir fait que ce qu'ils aimaient le plus). Il est important pour vous d'être bon en programmation. Il est tout aussi important de savoir équilibrer son travail entre les différentes matières. Être bon ingénieur en informatique ou en systèmes de communication c'est aussi avoir un solide bagage en mathématiques et en sciences de base.

Nous souhaitons à tous une année riche en apprentissage et couronnée de succès!