

<i>Sigle</i>	<i>Groupe</i>	<i>Trimestre</i>		
ECO3200	20	Automne		
<i>Titre</i>	Modélisation et analyse quantitative en sciences économiques			
<i>Horaire</i>	Mardi, 14h-17h			
<i>Enseignant</i>	Kristian Behrens			
<i>Courriel</i>	behrens.kristian@uqam.ca		<i>Local</i>	DS-M940
<i>Heures de disponibilité</i>	Lundi (10h-12h), sur rendez-vous			

1. Sommaire et objectifs du cours

L'outil informatique joue un rôle de plus en plus important dans les sciences économiques et est devenu de fait indispensable. L'analyse empirique et la création et extraction de données (e.g., web-scraping, reconnaissance optique, apprentissage machine, etc.) y ont massivement recours, ainsi que la simulation de modèles d'équilibre économique et la résolution de problèmes numériques complexes. Malheureusement, les étudiants en sciences économiques sont trop peu exposés à l'apprentissage des outils de programmation, alors même qu'une maîtrise de base de ces derniers constitue un atout considérable sur le marché du travail et dans le monde académique. Ceci est particulièrement pénalisant puisqu'il est rare de disposer exactement de l'outil nécessaire pour accomplir une tâche spécifique. Par conséquent, les étudiants seront amenés à devoir adapter les outils existants à leur besoin en écrivant du code.

Ce cours a donc pour objectif de familiariser les étudiants avec les éléments de base de la programmation informatique et de les appliquer à de simples problèmes pour lesquelles les outils 'tout prêts' ne sont pas toujours bien adaptés. L'accent est mis sur la compréhension et les éléments de base. Même si des langages de programmation particuliers sont priorisés dans ce cours (un langage bas niveau compilé, C++; et un langage de type 'script' haut niveau interprété, Stata), les connaissances acquises sont largement portables et donneront aux étudiants des bases pour progresser dans la maîtrise des outils et langages qu'ils seront amenés à utiliser plus tard.

2. Présentation du cours

Ce cours s'adresse aux étudiants du programme des sciences économiques. Il se veut être une introduction et ne nécessite pas de connaissances préalables en programmation. Il est important de souligner qu'il ne s'agit pas d'un cours de programmation tel qu'il serait dispensé traditionnellement dans un département de sciences informatiques. Nous passerons en effet sous silence un grand nombre de concepts plus avancés qui seraient vus dans le cadre d'un tel cours. Néanmoins, les bases que nous allons développer fournissent aux étudiants les outils nécessaires au cas où ils auraient besoin d'approfondir ces connaissances.

Ce cours consiste en **deux parties distinctes**. Dans la **première partie du cours**, nous allons présenter des éléments de base concernant : la syntaxe; les types de données et les variables; les

opérateurs, les instructions conditionnelles et les boucles; les fonctions et les membres d'objets; la gestion des erreurs; la logique des algorithmes; et des éléments de bonne pratique de travail en codage (structuration du code, conventions de notations, principe DnRY etc.). Pour ce faire, nous familiariserons les étudiants avec les aspects de base des langages C++ et Stata lesquels nous avons choisi pour ce cours. Le premier puisqu'il est de bas niveau (et force donc à se poser les questions de base et de les comprendre) ainsi que très performant et flexible. Le second puisqu'il est 'plus simple' et largement utilisé pour l'analyse empirique en sciences économiques à l'aide du logiciel Stata. Nous verrons aussi que ces deux langages sont conceptuellement très différents mais ont néanmoins beaucoup de similarités.

Le contenu priorisé est celui que nous jugeons important de comprendre pour développer des applications simples. L'acquisition de connaissances de base est priorisée par rapport à l'apprentissage de 'recettes de cuisine'. Trop de cours d'informatique pour économiste sont du type 'recette de cuisine', ce qui ne favorise ni la compréhension ni la capacité à résoudre des problèmes plus compliqués. Notre expérience montre que les recettes de cuisine ne sont jamais bien adaptées au problème concret que les étudiants seront amenés à résoudre.

Dans la **seconde partie du cours**, nous utiliserons les éléments vus dans la première partie pour créer une application concrète. L'objectif sera de développer un petit modèle de simulation qui sera codé en C++ et d'utiliser Stata afin d'analyser les résultats du modèle de simulation de manière automatisée. Même si cet exemple se veut simple, il permettra aux étudiants de mieux comprendre le processus de la recherche scientifique. Nous procéderons de manière modulaire, c'est-à-dire nous développerons progressivement les parties du code et les interfacieront afin d'arriver à un produit final. Cette partie du cours nécessitera que les étudiants refassent le codage chez eux afin de bien comprendre les différentes étapes et la manière de faire. L'application en C++ simulera un modèle de ségrégation de Schelling. L'accent sera mis sur une version opérationnelle du code, c'est-à-dire d'une version qui pourrait être étendu pour être appliquée aux données afin de simuler le modèle sur des problèmes de grande taille. Nous montrerons comment développer du code efficace permettant de simuler le modèle des centaines de milliers de fois afin de générer des ensembles de données qui pourront ensuite être analysés avec des outils statistiques tels que Stata. Nous analyserons aussi des données empiriques afin de vous illustrer l'intérêt concret de ce type de travail.

3. Matériel pédagogique

Aucun manuel n'est requis pour ce cours. Les manuels existants couvrent beaucoup trop de matière compliquée par rapport à ce que nous allons faire. Comme expliqué auparavant, nous allons présenter une introduction sélective adaptée aux problèmes d'applications que nous allons développer. Une fois la matière du cours assimilée, les étudiants désireux de poursuivre leur apprentissage pourront consulter n'importe quel ouvrage d'introduction plus détaillée à la programmation. De nombreuses ressources sont aussi disponibles en ligne, même si ces ressources sont souvent compliquées (voir, par exemple, le site très complet <https://en.cppreference.com/w/>).

Des acétates seront mis à la disposition des étudiants, ainsi que certaines capsules vidéo qui fourniront suffisamment d'information permettant d'assimiler la matière présentée lors du développement des applications. Il y aura aussi un grand nombre d'exemples à faire afin d'assimiler la matière. Tout le matériel du cours sera disponible en ligne sur le site Moodle du cours.

Il va sans dire que le 'matériel pédagogique' le plus important est la pratique. Ainsi, les étudiants devront pratiquer les notions développées en les explorant eux-mêmes et en effectuant systématiquement les exercices qui seront données. **Ceci implique de taper du code.** Nul ne peut apprendre de langue sans pratique : le même précepte s'applique à la programmation. Vous n'allez jamais progresser et maîtriser la programmation si vous ne passez pas un grand nombre d'heures à expérimenter avec du code et à essayer de l'explorer de vous-même.

4. Modalités d'enseignement

Ce cours se donnera en présentiel dans le local informatique.¹ Vous trouverez le calendrier du cours ci-dessous sous le point 5. Les séances de la première partie se donneront en deux blocs : un premier bloc (environ 1h15) pour corriger les exercices, répondre aux questions, et approfondir les points vus la semaine précédente; et un second bloc (environ 1h30) de nouvelle matière à apprendre et étudier pour la semaine suivante. Le second bloc du cours est de type 'cours magistral' : nous présenterons le matériel, donnerons des exemples et répondrons à vos questions de compréhension. Il y aura donc peu de travail préparatoire à faire de votre côté pour ce bloc, mais nous vous encourageons fortement d'expérimenter par vous-même avec le code présenté d'une semaine sur l'autre. Cependant, le premier bloc est de type travaux pratiques, c'est-à-dire que c'est vous qui allez devoir faire la majeure partie du travail et des explications.

Dans la seconde partie du cours, vous allez voir comment nous développons les différentes parties des applications sur les ordinateurs. Ensuite, ce sera à vous de reproduire la même chose en suivant les mêmes étapes et en essayant de réfléchir par vous-même à comment nous procédons et pourquoi. Dans cette partie du cours, **il est important de lire le matériel fourni et de le travailler chez vous avant le cours de la semaine afin de faciliter la compréhension**. Parfois nous allons vous demander d'essayer de proposer un code pour implémenter une fonctionnalité de l'application. Ensuite, il y aura un période durant lequel nous allons répondre à vos questions, revenir sur des notions mal comprises, corriger votre code etc. Donc, dans la deuxième partie du cours les 3h d'enseignement vont plus ressembler à des 'travaux pratiques' qu'à un cours magistral conventionnel.

5. Contenu détaillé et organisation de la session

Le cours se donnera en 15 séances de 3h chacune, incluant deux séances d'examen (un intra et un final). Comme expliqué ci-dessus, le cours se divise en deux grandes parties. Dans la première partie (les 8 premières séances), nous allons présenter des concepts et notions clés de la programmation dont nous allons avoir besoin pour développer des applications. Cette partie se donnera en cours magistral aux horaires prévus à cet effet (mardi de 14h-17h). L'examen intra portera sur cette partie et vérifiera l'acquisition des connaissances. La seconde partie (les 5 séances suivantes) sera consacrée au développement de l'application. Comme expliqué, cette partie prendra une forme de séances de travaux pratiques (style de cours interactif ou nous reviendrons sur vos questions et donnerons des explications et des exemples plus détaillés).

Calendrier prévisionnel de la session.

Date	Thème	Notions vues
5 septembre	Types de données, variables, syntaxe (1)	Introduction, généralités, variables
12 septembre	Types de données, variables, syntaxe (2)	Variables, types, conversions
19 septembre	Logique et instructions	Opérateurs, conditionnelles, boucles
26 septembre	Fonctions, méthodes, objets (1)	Fonctions, méthodes, objets
3 octobre	Fonctions, méthodes, objets (2)	Fonctions, méthodes, objets (2); bonnes habitudes à prendre
10 octobre	Introduction à l'algorithmique, notions plus avancées	Introduction à l'algorithmique, utilisation d'objets de la STL (e.g., <code>std::vector</code>)

¹ Il y aura aussi une ou deux séances basées sur des capsules vidéo permettant de présenter le matériel d'une autre manière et de faire du travail préparatoire.

17 octobre	Examen intra	
24 octobre	Semaine de préparation (capsules vidéos + lectures)	
31 octobre	Application, séance 1	Faits empiriques; préparation; cahier des charges; préparations des types;
7 novembre	Application, séance 2	Codage (1)
14 novembre	Application, séance 3 : Notions plus avancées pour l'application	Gestion de la mémoire; input-output; gestion des erreurs; codage (2)
21 novembre	Application, séance 4	Codage (3)
28 novembre	Application, séance 5	Codage (4)
5 décembre	Application, séance 6	Simulations; analyse; dé-briefing
12 décembre	Examen final	

Il n'y a pas de séances de monitorat dans ce cours. Les parties de séance qui sont des travaux pratiques avec les enseignants suffiront à cet effet.

6. Évaluation et barème d'évaluation

L'évaluation consiste en deux examens (un examen intra et un examen final). Dans ces examens nous allons vous demander de construire des mini-codes fonctionnels en utilisant les notions et outils vus en classe et/ou de trouver l'erreur et corriger un exemple de code erroné. Les modalités de ces examens sont les suivantes :

- Pour l'examen intra : l'énoncé vous sera envoyé vendredi 15 octobre 2021. Vous devez rendre deux fichiers avec votre code (un pour C++ et un autre pour Python) le vendredi suivant 22 octobre 2021. Pour rendre vos fichiers, vous devez les téléverser sur le site Moodle du cours.
- Pour l'examen final : l'énoncé vous sera envoyé vendredi 17 décembre 2021. Vous devez rendre deux fichiers avec votre code (un pour C++ et un autre pour Python) le vendredi suivant 24 décembre 2021. Pour rendre vos fichiers, vous devez les téléverser sur le site Moodle du cours.
- Chaque étudiant prépare son propre code. Dans votre code vous devez inclure des commentaires expliquant la fonctionnalité de chaque partie du code. Les commentaires servent à démontrer que vous comprenez votre code et leur contenu fera partie intégrante de l'évaluation de votre travail. Rendre des fichiers code identiques (copier-coller) pour plusieurs étudiants sera considéré comme plagiat.

Concernant les notes, le département des sciences économiques a adopté le barème de notation suivant :

Note	Lettre	Intervalle
4,3	A+	≥ 90
4	A	[86, 90]
3,7	A-	[82, 86]
3,3	B+	[78, 82]
3	B	[74, 78]
2,7	B-	[70, 74]
2,3	C+	[66, 70]
2	C	[62, 66]

1,7	C-	[58,62]
1,3	D+	[54,58]
1	D	[50,54]
0	E	< 50

Nous appliquerons ce barème au cours ECO3200.

Page d'information

Personnes étudiantes en situation de handicap bénéficiant d'aménagements

Les personnes étudiantes qui bénéficient d'aménagements offerts par le Bureau de l'inclusion et de la réussite étudiante (BIRÉ) doivent transmettre leur lettre (attestation des mesures d'aménagements) à leurs enseignants en début de session afin de les informer des aménagements qu'ils devront mettre en place.

Les personnes étudiantes en situation de handicap qui ne bénéficient pas d'aménagements peuvent contacter les services du BIRÉ afin de procéder à l'ouverture de leurs dossiers et à l'analyse de leurs besoins.

Accueil du service aux personnes étudiantes en situation de handicap situation.handicap@uqam.ca 514-987-3148 :AB-2300

Pour obtenir d'autres informations sur les services offerts par le BIRÉ : bire.uqam.ca

Les personnes étudiantes qui souhaitent se prévaloir de leurs aménagements dans le cadre d'examens devront faire leurs demandes de passation sur l'application dédiée étudiant.e.s en situation de handicap : sasesh.uqam.ca afin d'informer le Service des examens avec aménagements (SEA) de vos besoins en examen.

PLAGIAT - Règlement no 18 sur les infractions de nature académique

Tout acte de plagiat, fraude, copiage, tricherie ou falsification de document commis par une étudiante, un étudiant, de même que toute participation à ces actes ou tentative de les commettre, à l'occasion d'un examen ou d'un travail faisant l'objet d'une évaluation ou dans toute autre circonstance, constituent une infraction au sens de ce règlement.

La liste non limitative des infractions est définie comme suit :

- la substitution de personnes ;
- l'utilisation totale ou partielle du texte d'autrui en le faisant passer pour sien ou sans indication de référence ;
- la transmission d'un travail pour fins d'évaluation alors qu'il constitue essentiellement un travail qui a déjà été transmis pour fins d'évaluation académique à l'Université ou dans une autre institution d'enseignement, sauf avec l'accord préalable de l'enseignante, l'enseignant ;
- l'obtention par vol, manœuvre ou corruption de questions ou de réponses d'examen ou de tout autre document ou matériel non autorisés, ou encore d'une évaluation non méritée ;
- la possession ou l'utilisation, avant ou pendant un examen, de tout document non autorisé ;
- l'utilisation pendant un examen de la copie d'examen d'une autre personne ;
- l'obtention de toute aide non autorisée, qu'elle soit collective ou individuelle ;
- la falsification d'un document, notamment d'un document transmis par l'Université ou d'un document de l'Université transmis ou non à une tierce personne, quelles que soient les circonstances ;
- la falsification de données de recherche dans un travail, notamment une thèse, un mémoire, un mémoire-crédation, un rapport de stage ou un rapport de recherche.

Les sanctions reliées à ces infractions sont précisées à l'article 3 du Règlement no 18

Pour plus d'information sur les infractions académiques et comment les prévenir :

www.integrite.uqam.ca

Politique 16 sur le harcèlement sexuel

Le harcèlement sexuel se définit comme étant un comportement à connotation sexuelle unilatéral et non désiré ayant pour effet de compromettre le droit à des conditions de travail et d'études justes et raisonnables ou le droit à la dignité.

La Politique 16 identifie les comportements suivants comme du harcèlement sexuel :

1. Manifestations persistantes ou abusives d'un intérêt sexuel non désirées.
2. Remarques, commentaires, allusions, plaisanteries ou insultes persistants à caractère sexuel portant atteinte à un environnement propice au travail ou à l'étude.
3. Avances verbales ou propositions insistantes à caractère sexuel non désirées.
4. Avances physiques, attouchements, frôlements, pincements, baisers non désirés.
5. Promesses de récompense ou menaces de représailles, implicites ou explicites, représailles liées à l'acceptation ou au refus d'une demande d'ordre sexuel.
6. Actes de voyeurisme ou d'exhibitionnisme.
7. Manifestations de violence physique à caractère sexuel ou imposition d'une intimité sexuelle non voulue.
8. Toute autre manifestation à caractère sexuel offensante ou non désirée.

Pour plus d'information :

http://www.instances.uqam.ca/ReglementsPolitiquesDocuments/Documents/Politique_no_16.pdf

Pour rencontrer une personne ou faire un signalement : Bureau d'intervention et de prévention en matière de harcèlement : 514-987-3000, poste 0886

<http://www.harcelement.uqam.ca>

Informations sur les Services à la vie étudiante

Les Services à la vie étudiante accompagnent les étudiantes et étudiants dans la réussite de leur parcours universitaire.

Bureau des services-conseils (**soutien psychologique, bien-être aux études, information scolaire et insertion professionnelle, orientation, emploi**) : pour prendre rendez-vous, communiquez au 514 987-3185 ou par courriel à services-conseils@uqam.ca.

Bureau de l'**aide financière** : pour prendre rendez-vous, écrivez à aidefinanciere@uqam.ca

Concernant les **bourses**, pensez à consulter Le Répertoire institutionnel des bourses d'études (RIBÉ) et écrivez à bourses@uqam.ca pour toute question.

Consultez les informations et l'ensemble des coordonnées et services offerts par les Services à la vie étudiante à l'adresse suivante : vie-etudiante.uqam.ca