



Diplôme en technologies de l'information

INF 601 – Algorithmique et programmation procédurale

Plan de cours

Automne 2020

Mise en contexte

Bien que le monde de la programmation ait, dans son ensemble, pris le virage du modèle objet, la programmation procédurale (structurée) y demeure omniprésente, quand bien même ce ne serait que parce que le modèle objet lui doit en partie son succès. Encore aujourd'hui, il est souvent naturel d'aborder, partiellement ou intégralement, certains problèmes sous un angle structuré, quitte à inclure leur solution dans un cadre orienté objet plus englobant.

Les principes fondamentaux de la programmation structurée sont souvent approchés comme les principes fondamentaux de la programmation en général. Pour aller plus loin, il faut encore aujourd'hui passer d'abord par une compréhension solide de ceux-ci. Par la suite, nous verrons comment appliquer ces principes dans le contexte de la programmation orientée objet.

Place du cours dans le programme

Ce cours est une propédeutique permettant d'acquérir les compétences préalables nécessaires pour aborder le diplôme en technologies de l'information.

Cours du programme préalable à celui-ci :

- *Aucun*

Cours du programme en lien avec ce cours :

- INF 731 – Programmation orientée objet;
- INF 732 – Bases de données.

Descripteur du cours

Objectifs : Savoir analyser un problème; avoir un haut degré d'exigence quant à la qualité des programmes; pouvoir développer systématiquement des programmes de bonne qualité, dans le cadre de la programmation procédurale séquentielle.

Contenu : Introduction aux ordinateurs. Analyse et conception de solutions informatiques : simplification, décomposition, modularisation et encapsulation. Critères de qualité : la conformité, la fiabilité et l'adaptabilité. Concepts de base de la programmation structurée : séquence, itération, sélection. Modélisation du traitement et modularité : concept de fonctions et d'abstraction procédurale. Introduction aux structures de données. Introduction à la programmation orientée objet.

Objectifs généraux

Ce cours porte sur la conception logicielle. Il contribue de façon majeure à l'atteinte des objectifs généraux d'un cours de base de programmation qui consistent à développer :

- une connaissance générale des concepts relatifs au développement d'applications;
- une connaissance appliquée des concepts algorithmiques élémentaires;
- une connaissance appliquée des concepts relatifs à la modularisation;
- une connaissance appliquée des concepts relatifs à l'organisation des données en mémoire;
- une connaissance appliquée de la programmation procédurale (structurée);
et,
- une connaissance théorique et appliquée des concepts préalables à l'introduction de la programmation orientée objet (POO).

Objectifs spécifiques

Après avoir complété ce cours, les étudiantes et les étudiants seront capables :

- {1} de comprendre le fonctionnement d'un ordinateur dans le contexte de l'utilisation d'un outil d'élaboration d'une solution informatique;
- {2} d'analyser un problème dans le but d'y trouver une solution informatique;
- {3} d'élaborer une solution formelle (algorithmique) à un problème;
- {4} de modéliser un algorithme en respectant une convention textuelle ou graphique;
- {5} d'implémenter un algorithme à l'aide d'un langage de programmation;
- {6} de planifier une séance de test pour un programme;
- {7} de vérifier le bon fonctionnement d'un programme;
- {8} de respecter des normes et des standards de programmation;

Planification

Introduction au développement logiciel – méthodologie et outils

Concept de logiciel. Problèmes et solutions. Processus et cycle de développement. Phase d'élaboration de la solution. Notions d'intrants, d'extrants, de traitement et de jeux d'essai. Conception et représentation de l'algorithme. Phase de conception logicielle. Traduction et validation. Exécution d'un programme par un ordinateur et raison d'être des langages de programmation. Environnement de développement intégré. Compilation et édition de liens.

Arrimage aux objectifs : objectifs {1} et {2}.

Outil de programmation : Microsoft Visual C# .NET

Concepts algorithmiques fondamentaux

Structure et représentation d'un algorithme. Les composants d'un algorithme. Les données, les opérations, les expressions et les structures. Variables et constantes. Concepts d'identificateur et de type de données. Les opérations primitives : lecture, écriture et affectation. Les expressions arithmétiques, relationnelles et logiques. Les structures algorithmiques : la séquence, l'alternative et la répétitive. Traduction d'un algorithme en programme. Vérification du bon fonctionnement du programme et déverminage.

Arrimage aux objectifs : objectifs {2} à {8}.

Outil de programmation : Microsoft Visual C# .NET

La modularisation : fonctions et bibliothèques

Analyse et modélisation procédurale. Notions de sous-programmes. Les échanges d'informations entre sous-programmes : valeur de retour et paramètres, par valeur et par référence. Particularités des paramètres : paramètres constants, tableau. Les bibliothèques : conception et usage.

Arrimage aux objectifs : objectifs {2} à {8}.

Outil de programmation : Microsoft Visual C# .NET

Allocation dynamique et organisation de la mémoire

Organisation des données en mémoire. Introduction à la notion de tableau. Définition d'un tableau. Usages possibles. Débordement. Adressage. Passage d'un tableau en paramètre. Notions relatives aux adresses et aux références.

Arrimage aux objectifs : objectifs {2} à {8}.

Note : la notion de tableau aura été introduite dans la deuxième section du cours.

Outil de programmation : Microsoft Visual C# .NET

Types définis par l'utilisateur

Introduction aux classes et aux énumérations.

Arrimage aux objectifs : objectifs {2} à {8}.

Outil de programmation : Microsoft Visual C# .NET

Approche pédagogique préconisée

Étant donné qu'il s'agit d'un mode d'enseignement à distance, l'atteinte des objectifs d'apprentissage se fera par le biais :

- de la lecture de documents exposant les aspects théoriques et pratiques;
- de la lecture de document présentant des études de cas;
- la participation au forum de discussion du cours, où les étudiantes et les étudiants sont fortement encouragés à contribuer par leurs questions et commentaires;
- de la réalisation d'exercices formatifs; et
- par des contrôles d'autoévaluation liés aux différentes parties du cours.

L'évaluation des objectifs d'apprentissage se fera par le biais :

- de contrôles et de travaux vérifiant les aspects pratiques et théoriques, qui seront évalués par l'enseignant de façon formelle et dont le résultat contribuera à votre note finale; et
- d'un contrôle théorique final sous supervision vérifiant formellement l'atteinte des objectifs.

Règles régissant l'évaluation des apprentissages

- Étant donné la nature à distance de ce cours et pour éviter que les étudiantes et les étudiants accumulent un retard insurmontable dans leur progression, aucun retard ne sera toléré dans la remise des travaux pratiques.
- Les règles de qualité des programmes qui seront mentionnées en cours de session seront applicables aux travaux pratiques et au code rédigé dans le cadre des contrôles.
- Tout travail devra être produit dans un français jugé de bonne qualité. Une pénalité allant jusqu'à 5 % des points pourra être appliquée à un travail produit dans un français ne rencontrant pas les standards de qualité de la faculté des sciences.
- Le non-respect des modalités de remise et des consignes de réalisation des travaux peut entraîner la note 0 pour le travail concerné.
- L'absence à un examen donne droit à la note 0.
- L'échec à l'examen final entraîne automatiquement l'échec au cours.

Évaluation de l'apprentissage

Évaluation	Pondération
<p>Travaux</p> <p>Quatre travaux seront à réaliser tout au long de la session. Ces travaux comporteront une section théorique et une section pratique. Dans les sections pratiques, vous devrez, à partir d'un problème concret, appliquer les notions liées à l'élaboration et à la réalisation de sa solution. Chacun de ces travaux demandera une part importante de programmation.</p> <p>Les travaux viseront à évaluer la progression générale des étudiantes et des étudiants. Ils couvriront donc l'ensemble de la matière présentée entre le début du cours et le moment du contrôle.</p> <p>Ces travaux se feront à distance, mais devront être réalisés de façon individuelle en respectant certaines règles.</p> <p>Ces travaux se feront sur base individuelle.</p>	<p>70 %</p>
<p>Examen final</p> <p>Un examen final récapitulatif valant 30 % de la note finale aura lieu lors de la dernière séance de la session. Cet examen portera sur l'ensemble de la matière de la session.</p> <p>L'examen final est une évaluation strictement individuelle pour laquelle vous n'aurez pas droit à vos notes de cours, ni à un ordinateur ou tout autre équipement électronique.</p> <p>Étant donné que l'examen final sera la seule évaluation supervisée, un échec à celle-ci engendrera automatiquement un échec au cours.</p>	<p>30 %</p>

Note sur le plagiat

Conformément à l'article 9 du Règlement des études de l'Université de Sherbrooke, le plagiat, soit le fait dans une activité évaluée de faire passer pour sien les idées et le travail d'autrui, est un délit académique qui peut être sanctionné par les autorités disciplinaires compétentes. Peuvent être imposées à titre de sanctions, l'une ou plusieurs des mesures suivantes :

- a) la réprimande simple ou sévère consignée au dossier étudiant pour la période fixée par l'autorité disciplinaire ou, à défaut, définitivement. En cas de réprimande fixée pour une période déterminée, la décision rendue demeure au dossier de la personne aux seules fins d'attester de l'existence du délit en cas de récidive;
- b) l'obligation de reprendre une production ou une activité pédagogique, dont la note pourra être établie en tenant compte du délit survenu antérieurement;
- c) la diminution de la note ou l'attribution de la note E ou 0;
- d) le renvoi du dossier à la personne responsable de l'évaluation d'une production ou d'une activité pédagogique pour qu'elle attribue une nouvelle note en tenant compte du délit.

Par plagiat, on entend notamment :

- copier intégralement une phrase ou un passage d'un livre, d'un article de journal ou de revue, d'une page Web ou de tout autre document en omettant d'en mentionner la source ou de le mettre entre guillemets;
- reproduire des présentations, des dessins, des photographies, des graphiques, des données sans en préciser la provenance et, dans certains cas, sans en avoir obtenu la permission de reproduire;
- utiliser, en tout ou en partie, du matériel sonore, graphique ou visuel, des pages Internet, du code de programme informatique ou des éléments de logiciel, des données ou résultats d'expérimentation ou toute autre information en provenance d'autrui en le faisant passer pour sien ou sans en citer les sources;
- résumer ou paraphraser l'idée d'un auteur sans en indiquer la source;
- traduire en partie ou en totalité un texte en omettant d'en mentionner la source ou de le mettre entre guillemets;
- utiliser le travail d'un autre et le présenter comme sien (et ce, même si cette personne a donné son accord);
- acheter un travail sur le Web ou ailleurs et le faire passer pour sien;
- utiliser sans autorisation le même travail pour deux activités différentes (autoplégat).

Adresses électroniques pour contacter l'enseignant

- Enseignant : vincent.echelard@usherbrooke.ca

Identifiant Skype de l'enseignant

- Enseignant : **vechelard**

Site du Moodle du cours INF 601

- <https://www.usherbrooke.ca/moodle2-cours/course/view.php?id=10371>

Bibliographie (non obligatoire)

Les livres énumérés ci-dessous ne seront pas utilisés dans le cours, mais pourrait servir de support à un étudiant qui chercherait une référence complémentaire. Les règles et standards de programmation présentés dans les notes de cours ont préséances sur celles qui pourraient être utilisées dans ces manuels.

C#: C# for Beginners Crash Course: Master C# Programming Fast and Easy Today

Raj Ali, 2015, CreateSpace Independent Publishing Platform; 1^{re} édition, 178 pp.

ISBN-10: 1514873893

ISBN-13: 978-1514873892

14,34 CAD chez amazon.ca (4,14 CAD Kindle Edition)

Essential C# 7.0 (6th edition)¹

MICHAELIS, Mark with LIPPERT, Eric;

Addison-Wesley, 2018.

ISBN-10: 1509303588

ISBN-13: 978-1509303588

68,01 CAD chez amazon.ca (47,99 CAD Kindle Edition)

¹ Il s'agit actuellement du manuel obligatoire pour le cours INF731 – Programmation orientée objet