

IFT 1166
Programmation Orientée Objet en C++
<http://www.iro.umontreal.ca/~pift1166>

Présentation du cours

Ce cours est destiné aux étudiants ayant déjà suivi un premier cours de programmation dans un des langages C ou Java et qui désirent apprendre la programmation orientée objet en langage C++. Le cours présentera les concepts à la base de ce langage et l'application de ces concepts aux structures de données simples.

Modalités

Le cours, offert durant le trimestre d'hiver 2026, combine théorie et travaux pratiques sur une période de 12 semaines. Chaque semaine comprend 2 heures de cours théoriques et 2 heures de travaux pratiques.

Les séances se déroulent selon l'horaire suivant :

- **Vendredi 16h30 à 18h30** — Cours magistral (Salle Y-117), pavillon Roger Gaudry.
- **Vendredi 18h30 à 20h30** — Travaux pratiques / démonstration (Salle 1340), pavillon André-Aisenstadt.

Le premier cours théorique aura lieu le **9 janvier 2026**, et la première séance de travaux pratiques débutera le **16 janvier 2026**.

La dernière séance de cours et de travaux pratiques est prévue pour le **10 avril 2026**.

Aucune séance n'aura lieu à la date suivante :

- **6 mars 2026** — Semaine de relâche (activités libres).

Préalable

Préalables académiques : IFT1810 ou IFT1969.

Références

Le polycopié du cours est accessible sur la page web dédiée au cours.

Une référence principale : [Claude Delannoy, Programmer en C++ moderne](#), éditions Eyrolles.

D'autres ouvrages complémentaires seront présentés lors de la première séance.

Pour vérifier la disponibilité des livres en bibliothèque, veuillez consulter le comptoir de prêt :

<http://www.bib.umontreal.ca/nous-joindre/MI.htm>

Travaux pratiques

Les travaux pratiques comptent pour 40 % de la note finale. Ils seront réalisés sur les systèmes Windows ou Linux, dans l'environnement du réseau IRO, à l'aide du compilateur Gnu g++ (version 15.x).

Examens

Deux évaluations sont prévues :

- Examen intra : 20 % de la note finale — 27 février 2026, de 16h30 à 18h30, salle Y-117.
- Examen final : 40 % de la note finale — 24 avril 2026, de 16h30 à 19h30, salle Z-330.

Absences autorisées aux examens

- Pour l'examen **intra**, une reprise sera prévue durant la session, à une date qui sera précisée ultérieurement.
- Pour l'examen **final**, une reprise aura lieu à une date fixée par l'administration.

Dates importantes à retenir

- **22 janvier 2026** : Date limite pour annuler votre inscription sans frais.
- **13 mars 2026** : Date limite pour abandonner le cours avec frais.

Ces dates sont fournies par la **Faculté des arts et des sciences (FAS)**. Si vous êtes inscrit dans une autre faculté, veuillez vérifier que ces échéances s'appliquent également à votre programme.

Notation

Pour réussir ce cours, la moyenne pondérée des examens théoriques (intra + final) doit atteindre **au moins 50 %**.

- Si **Intra (sur 20) + Final (sur 40) ≥ 30**, alors :
Note finale = Intra + Final + TP
- Sinon :
Note finale = Intra + Final + TP / 2

Autrement dit, la réussite aux examens théoriques est indispensable pour valider le cours. La note sur 100 sera convertie en note littérale (**A+, A, A-, etc.**) à la fin du trimestre, selon un barème tenant compte de la moyenne du groupe et de la répartition des résultats. La note finale doit également satisfaire les exigences de la faculté dans laquelle l'étudiant est inscrit.

Rappel important : Le plagiat est strictement interdit et sanctionné par le *Règlement disciplinaire sur la fraude et le plagiat concernant les étudiants*. Pour plus d'informations, consultez : www.integrite.umontreal.ca

Sujets abordés durant le cours

- Anatomie d'un programme C++
- C++, C et Java
- Identificateurs, types de base et variables
- Les entrées et les sorties en C++
- Spécificités C++
- Opérateurs, expressions et conversions
- Les structures de contrôle
- Fonctions et Tableaux
- Pointeurs et arithmétique des pointeurs
- Allocation dynamique
- Énumérations et Structures
- Notions de Programmation Orientée Objet
- Classes et Objets
- Fonctions membres
- Construction et destruction des objets
- Héritage et fonctions virtuelles
- Listes chaînées, files, piles et arbres
- Algorithme de tri et de recherche