

IFT 1169
Programmation avancée en C++
<http://www.iro.umontreal.ca/~dift1169>

Présentation du cours

Ce cours est destiné aux étudiants ayant une maîtrise intermédiaire de la programmation orientée objet en C++, et souhaitant approfondir leurs compétences. Il abordera des concepts avancés de la programmation orientée objet, la bibliothèque de modèles standard (STL), ainsi que leur application aux structures de données complexes et aux interfaces graphiques. Les nouveautés du langage C++, introduites dans les dernières normes, seront intégrées progressivement tout au long du trimestre.

Modalités

Le cours, offert durant le trimestre d'automne 2025, 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 au pavillon Roger-Gaudry selon l'horaire suivant :

- **Mardi 18h30 à 20h30** — Cours magistral (Salle S-144).
- **Mardi 20h30 à 22h30** — Travaux pratiques / démonstration (Salle X-115).

Le premier cours théorique aura lieu le **2 septembre 2025**, et la première séance de travaux pratiques débutera le **9 septembre 2025**.

La dernière séance de cours et de travaux pratiques est prévue pour le **9 décembre 2025**.

Aucune séance n'aura lieu aux dates suivantes :

- **30 septembre 2025** — Journée nationale de la vérité et de la réconciliation.
- **21 octobre 2025** — Semaine de relâche (activités libres).

Préalable

Préalables académiques : IFT1166.

Références

Le polycopié du cours ainsi que les enregistrements vidéo (version 2.0) sont accessibles 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 — 28 octobre 2025, de 18h30 à 20h30, salle Y-117.
- Examen final : 40 % de la note finale — 23 décembre 2025, de 18h30 à 21h30, salle Z-310.

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 au début de l'année 2026.

Dates importantes à retenir

- **17 septembre 2025** : Date limite pour annuler votre inscription sans frais.
- **7 novembre 2025** : 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) \geq 30**, alors :

$$\text{Note finale} = \text{Intra} + \text{Final} + \text{TP}$$

- Sinon :

$$\text{Note finale} = \text{Intra} + \text{Final} + \text{TP} / 2$$

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

- Identification dynamique de type (RTTI)
- Espaces de noms
- Fonctions amies
- Surcharge des opérateurs
- Gestion des exceptions
- Flots d'entrée/sortie
- Patrons de fonctions et de classes
- Bibliothèque de modèles standard (STL)
- Interfaces graphiques
- Gestion avancée de la mémoire
- Programmation concurrente (threads)
- Nouveautés du langage C++ introduites dans les dernières normes