

# Introduction

---

IFT503/711 – Théorie du calcul

Hiver 2024



- Plan de cours
- Introduction à la théorie du calcul

# Plan de cours

## Michael Blondin

@ michael.blondin@usherbrooke.ca

🌐 mblondin.espaceweb.usherbrooke.ca/

📍 bureau D4-1024-1 au 1<sup>er</sup> étage

## Manuel Lafond

@ manuel.lafond@usherbrooke.ca

🌐 mlafond.recherche.usherbrooke.ca/

📍 bureau D4-2010 au 2<sup>ème</sup> étage

## Dave Touchette

@ dave.touchette@usherbrooke.ca

📍 bureau D4-1018-2 au 1<sup>er</sup> étage

### Discipline:

- 15 × informatique
- 13 × information quantique
- 2 × physique

### Cycle:

- 23 × baccalauréat
- 6 × maîtrise (4 × recherche, 2 × cours)
- 1 × doctorat

<b>Cours</b>	Mercredi	10h30 – 12h20
	Vendredi	10h30 – 12h20
<b>Exercices</b>	Selon la semaine	

## **Préalable (au 1<sup>er</sup> cycle):**

- IFT313 – Introduction aux langages formels

## **Atouts:**

- IFT436 – Algorithmes et structures de données
- IFT339 – Structures de données

## **Utile pour:**

- tout cours avec des problèmes algorithmiques

- Qu'est-ce qu'un ordinateur et un algorithme?
- L'ordinateur peut-il répondre à toutes nos questions?
- Peut-on toujours rendre un algo. efficace?
- Peut-on surmonter les problèmes difficiles?
- Plusieurs processeurs = toujours plus rapide?
- Qu'est-ce que l'ordinateur quantique?

## Grandes questions du cours

- Qu'est-ce qu'un ordinateur et un algorithme?
- L'ordinateur peut-il répondre à toutes nos questions?
- Peut-on toujours rendre un algo. efficace?
- Peut-on surmonter les problèmes difficiles?
- Plusieurs processeurs = toujours plus rapide?
- Qu'est-ce que l'ordinateur quantique?

## Grandes questions du cours

- Qu'est-ce qu'un ordinateur et un algorithme?
- L'ordinateur peut-il répondre à toutes nos questions?
- Peut-on toujours rendre un algo. efficace?
- Peut-on surmonter les problèmes difficiles?
- Plusieurs processeurs = toujours plus rapide?
- Qu'est-ce que l'ordinateur quantique?

## Grandes questions du cours

- Qu'est-ce qu'un ordinateur et un algorithme?
- L'ordinateur peut-il répondre à toutes nos questions?
- Peut-on toujours rendre un algo. efficace?
- **Peut-on surmonter les problèmes difficiles?**
- Plusieurs processeurs = toujours plus rapide?
- Qu'est-ce que l'ordinateur quantique?

## Grandes questions du cours

- Qu'est-ce qu'un ordinateur et un algorithme?
- L'ordinateur peut-il répondre à toutes nos questions?
- Peut-on toujours rendre un algo. efficace?
- Peut-on surmonter les problèmes difficiles?
- Plusieurs processeurs = toujours plus rapide?
- Qu'est-ce que l'ordinateur quantique?

## Grandes questions du cours

- Qu'est-ce qu'un ordinateur et un algorithme?
- L'ordinateur peut-il répondre à toutes nos questions?
- Peut-on toujours rendre un algo. efficace?
- Peut-on surmonter les problèmes difficiles?
- Plusieurs processeurs = toujours plus rapide?
- Qu'est-ce que l'ordinateur quantique?

## Fondements

1. Calculabilité
2. Décidabilité
3. P vs. NP

## Sujets avancés

4. Logique
5. Calcul parallèle (sous P)
6. Complexité (au-delà de NP)
7. Informatique quantique

## 1. Calculabilité

8h (Blondin)

- Ordinateur et algorithmes: formalisation
- Machine de Turing
- Hypothèse de Church-Turing
- Temps et mémoire

## 2. Décidabilité

**8h** (Touchette)

- Problèmes décidables et non décidables
- Réductions entre problèmes
- Problème de l'arrêt
- Théorème de récursion

## 3. P vs. NP

8h (Lafond)

- Temps polynomial déterministe et non déterministe
- Réductions polynomiales
- NP-complétude

## 4. Logique

4h (Blondin)

- Logique du premier ordre
- Arithmétique: décidabilité et indécidabilité

## 5. Calcul parallèle

4h (Blondin)

- Classes à l'intérieur de P
- Circuits booléens
- P-complétude

## 6. Complexité du calcul (au-delà de NP)

6h (Lafond)

- Complexité en espace
- Complexité exponentielle

## 7. Informatique quantique

**6h** (Touchette)

- Circuits quantiques
- Algorithmes quantiques

## **Cours théorique** (pas de programmation)

- Enseignement au tableau
- Certaines notes de cours en ligne
- Références:
  - M. SIPSER: *Introduction to the Theory of Computation*, 2<sup>ème</sup> édition
  - S. ARORA, B. BARAK: *Computational Complexity: A Modern Approach*
  - I. CHUANG, M. NIELSEN: *Quantum Computation and  
Quantum Information*

- 6 devoirs
- 1<sup>er</sup> cycle: à deux ou individuel
- 2/3<sup>ème</sup> cycles: individuel + questions avancées

Devoirs	60%
Examen périodique	—
Examen final	40%

Sujets	Devoirs
1: Calculabilité	Devoir 1 ~ 2 semaines
2: Calculabilité	
3: Décidabilité	Devoir 2 ~ 3 semaines
4: Décidabilité	
5: Décidabilité, P vs. NP	
6: P vs. NP	Devoir 3 ~ 1½ semaine
7: P vs. NP, logique	
8: Pas de cours (examens autres cours)	
9: Relâche	

Sujets	Devoirs
10: Logique, calcul parallèle	Devoir 4 ~ 2 semaines
11: Calcul parallèle, complexité	
12: Complexité	Devoir 5 ~ 2 semaines
13: Complexité, info. quantique	
14: Info. quantique	Devoir 6 ~ 1½ semaine
15: Examens	—
16: Examens	—

Sur **rendez-vous**

en priorité avec le prof.  
qui a enseigné la matière

 [ift503-mb.espaceweb.usherbrooke.ca](http://ift503-mb.espaceweb.usherbrooke.ca)

# Calculabilité