

---

Lu 09:00 - 10:59	S-144 Pav. Roger-Gaudry	Travaux pratiques : débutent 6 mai	
Me 14:00 - 14:59	S-144 Pav. Roger-Gaudry	Me 15:00 - 16:59	S-144 Pav. Roger-Gaudry

Cours : débutent 4 mai

Examens			
Jours et heures	Local	Description	Date
Lu 09:00 - 10:59	1360 Pav. André-Aisenstadt	Examen intra	2015-06-15
Me 14:00 - 16:59	1360 Pav. André-Aisenstadt	Examen final	2015-07-22

Congés : les lundi 18 mai, mercredi 24 juin et mercredi 1er juillet.

---

## 1 Contenu général et objectifs du cours :

Est-il possible d'analyser un programme afin

- de déterminer s'il bouclera à l'infini ?
- de borner le temps que prendra son exécution s'il ne boucle pas ?
- de suspecter la piraterie puisqu'il aura le même comportement qu'un programme connu ?

La théorie sous-jacente à l'informatique offre des balises en ce qui a trait à ce type de questions. La théorie permet également parfois de dire qu'un problème calculatoire donné est impossible à résoudre en un temps raisonnable, *quel que soit* le programme utilisé.

La théorie ne fournit pas que des réponses négatives. Elle rend aussi des choses possibles. Par exemple, ce sont des travaux théoriques sur les langages formels et la complexité qui ont permis des progrès répétés dans les domaines de la compilation des langages de programmation, de la vérification formelle de programmes et de la cryptographie.

Voici les principaux objectifs du cours IFT2105 :

- faire saisir la portée de l'étude des langages formels et de leur classification
- introduire la complexité du calcul et aborder une question centrale de l'informatique théorique actuelle, soit celle du lien entre les classes de complexité P et NP
- introduire la calculabilité, dont l'intérêt est à la fois philosophique, mathématique et informatique
- offrir un tremplin pour une étude plus approfondie.

En plus d'acquérir les connaissances factuelles ci-dessus, l'étudiant(e)

- adoptera un mode de pensée rigoureux,
- développera une plus grande aisance face au formalisme mathématique,
- saura prendre un certain recul face à l'outil qu'est l'ordinateur.

## 2 Bibliographie

Les livres ci-dessous seront placés en réserve à la bibliothèque de math-info (pav. Aisenstadt).

### 2.1 Livre fortement recommandé

Nous tirerons 90% de la matière du cours du livre de Michael Sipser, *Introduction to the Theory of Computation*, dont le prix est malheureusement outrageux. Une 3<sup>ième</sup> édition du livre est parue en 2013 mais la 2<sup>ième</sup> édition de 2006 fait très bien l'affaire.

## 2.2 Autres références

- John E. Hopcroft, Rajeev Motwani et Jeffrey D. Ullman, *Introduction to Automata Theory, Languages, and Computation*, 3<sup>ème</sup> édition, Addison-Wesley, 2007. ISBN : 0-321-46225-4.
- Pierre Wolper, *Introduction à la calculabilité*, 2<sup>ème</sup> édition, Dunod 2001, ISBN 2-10-004853-8.

## 3 Évaluation : devoirs (30%), intra (30%) et final (40%)

Les deux examens seront à livre fermé et l'examen final portera sur la matière de toute la session. Il y aura quatre devoirs à caractère théorique, sans programmation. Pour les devoirs, vous pouvez former des équipes de deux et ne remettre qu'une copie par équipe. Si la note moyenne aux examens n'atteint pas le seuil de 40%, le cours est échoué.

## 4 Plan approximatif de la session

Heures de cours	Matière	Chapitre de Sipser
3	Introduction, motivation, révision de techniques de preuve, terminologie	0, extra
7	Langages réguliers Automates finis Expressions régulières	1
3	Langages hors-contextes	2
5	Machines de Turing et hypothèse de Church-Turing	3, extra
7	Complexité du calcul P, NP, NP-complétude Sujets connexes	7,8
4	Indécidabilité Problème d'arrêt	4,5
TOTAL 29		

## 5 Et sur le plan méta

Vous devez avoir pris connaissance du [Code d'honneur de l'étudiant du DIRO](#). En particulier,

- citez scrupuleusement toute source d'information utilisée dans vos travaux
- remettre un devoir en équipe engage la responsabilité de toute l'équipe.

Tout présumé cas de plagiat sera traité selon les règles de l'université. Pour plus d'information, consultez [Intégrité à l'Université de Montréal](#).