

Journal de bord du module Complexité

Les renvois de la table de matières et les hyperliens en couleur magenta sont cliquables.

Table des matières

Documents et détails organisationnels

[Recueil d'exercices.](#)

[Notes du cours \(dernière mise à jour le 22 mars 2020\).](#)

[Devoir à rendre.](#)

1 Introduction

- Motivation : répondre aux questions
 - qu'est-ce un ordinateur ?
 - que peut-on (en principe) calculer sur un ordinateur ?
 - que peut-on (en pratique) calculer sur un ordinateur ?
- Développement historique de la théorie des ensembles et de la logique.
 - Dixième problème de Hilbert et *non ignorabimus*.
 - Théorie naïve des ensembles : Cantor, Frege, Dedekind, Peano.
 - Paradoxe de Russell.
 - *Principia mathematica* de Russell et Whitehead.
 - Axiomatisation de Zermelo-Fraenkel.
 - Incomplétude de Gödel.
- Calculabilité effective.
 - Thèse de Church-Turing.
 - Équivalence de Turing-calculabilité, Church λ -calculabilité, Gödel-Herbrand-calculabilité.
- Complexité algorithmique, classes de complexité.

2 Relations

- Rappels sur le produit cartésien.
- Représentations cartésienne, matricielle, sagittale.
- Relations réciproque, complémentaire. ← **Fin du cours du 08 janvier 2020.**
- Réunion ou intersection de deux relations.
- Composition des relations.
- Relations réflexives, anti-réflexives, symétriques, antisymétriques.

- R -chemins.
- Graphes dirigés $G = (G^0, G^1, s, t)$. Bijection entre graphes dirigés sans arêtes multiples et relations.
- Espace des trajectoires G^* d'un graphe dirigé.
- Clôtures réflexive, symétrique, transitive. ← Fin du cours du 15 janvier 2020.
- Relations acycliques, arbres.
- Si T est un arbre avec racine r , alors
 - T est une relation acyclique,
 - la racine r est unique,
 - l'ensemble de prédécesseurs de r est vide,
 - tout sommet différent de la racine a exactement un prédécesseur. ← Fin du cours du 22 janvier 2020.
 - TD du 22 janvier 2020: Exercices 1–6.
- Arbres ordonnés; représentation par ensembles de listes de couples.
- Arbres positionnels et leur arbres binaires.
- Arbres couvrants.
- Amalgame. ← Fin du cours du 29 janvier 2020.
- TD du 29 janvier 2020: Exercices 7–10.

3 Langages formels et grammaires génératrices

- Grammaires : symboles terminaux et non-terminaux, productions.
- Les productions définissent une relation sur les mots.
- La dérivation directe engendre un digraphe sur les mots (éventuellement avec des arêtes multiples).
- Clôture monoïdale de la dérivation.
- Langage sur un alphabet fini A engendré par une grammaire.
- Hiérarchie de Chomsky de grammaires (linéaires, non-contextuelles, contextuelles, non-contraintes) et automates de reconnaissance correspondants.
- Arbres de dérivation pour les grammaires algébriques (et régulières).
- Grammaires non-contextuelles ambiguës. Ambiguïté intrinsèque d'un langage. ← Fin du cours du 05 février 2020.

4 La logique comme langage

- Algèbres de Boole.
- Fonctions booléennes; circuits.
- Logique propositionnelle (bivaluée classique) vue comme langage.
- Sémantique des formules propositionnelles.
 - Valuations, modèles, formules satisfaisables, tautologies.
 - Conséquence logique $\Phi \models \psi$. ← Fin du cours du 12 février 2020.
 - Équivalences logiques.
 - Axiomes, théorèmes, règles d'inférence.
- Un exemple élémentaire d'une logique modale (non-classique).

- Logique du premier ordre (logique des prédicats).
- Termes, formules atomiques, formules.
- Règles de déduction. Preuves.
- Idées essentielles de la preuve du premier théorème d'incomplétude. ← Fin du cours du 19 février 2020.
- TD du 19 février 2020: Exercices : 12–15.

5 Automates finis

5.1 Automates déterministes finis (ADF) et langages réguliers

- Quelques premières notions de base.
- Définition d'un ADF ; l'opération d'un ADF vue comme l'évolution d'un système dynamique.
- Langage reconnu par un ADF.
- Langages réguliers.
- Opérations sur les langages : réunion, concaténation, clôture monoïdale.
- La classe de langages réguliers est fermée par réunion finie (démonstration directe). ← Fin du cours du 04 mars 2020.
- TD du 04 mars 2020: Exercices 16–21, 23, 22a–22b.

5.2 Automates non-déterministes finis (ANF)

- Quelques notions de base.
- Définition formelle d'un automate non-déterministe fini.
- Pour tout automate fini non-déterministe, il existe un automate fini déterministe qui lui est équivalent.
- La classe des langages réguliers est fermée par réunion finie (re-démonstration de ce résultat à l'aide d'automates finis non-déterministes).
- La classe des langages réguliers est fermée par concaténation et par clôture monoïdale. ← Fin du cours du 06 mars 2020.
- TD du 06 mars 2020: Exercices 22c–22d, 24–25.

5.3 Automates non-déterministes généralisés finis (ANGF) et expressions régulières

- Expressions régulières.
- Théorème : Un langage est régulier si et seulement si une expression régulière le décrit.
- Transformation d'un automate déterministe fini en automate généralisé.
- Algorithme récursif de transformation d'un automate fini généralisé en expression régulière. Quelques exemples exhaustivement traités.
- Théorème de l'itération (« pumping lemma ») permettant d'établir la non-régularité d'un langage. ← Fin du cours du 11 mars 2020.
- TD du 11 mars 2020: Exercices 26–27.

6 Machines de Turing

- Introduction très informelle.
- Descriptions qualitative et formelle.
- Configuration instantanée d'une machine de Turing.
- Système dynamique associé à toute trajectoire calculatrice.
- Temps d'arrêt.
- Langages reconnaissables ; fonctions partielles réalisées par une machine de Turing.
- Le dixième problème de Hilbert est Turing-reconnaissable mais il n'est pas Turing-décidable.
- Exemple explicite de fonctionnement.
- Variantes : machines non-déterministes, à plusieurs rubans, avec mouvements enrichis de la tête.
- Toute machine non-déterministe a une machine équivalente déterministe.

7 Décidabilité

- Problème calculatoire comme problème d'appartenance dans un langage.
- Les langages des automates finis déterministes, non-déterministes ou des expressions régulières sont décidables.
- Le problème d'acceptation d'automates dont le langage est vide est décidable.
- Un exemple de codage d'un automate dans l'alphabet de l'entrée.
- Démonstration du fait que l'acceptation d'automates dont le langage est vide est décidable.
- Le langage qui consiste à accepter un couple de mots s'ils sont les descriptions d'automates acceptant le même langage est décidable.
- Le langage des machines de Turing est reconnaissable mais non décidable.
- Un langage L est décidable si, et seulement si, L et \bar{L} sont reconnaissables.
- Il existe de langages non Turing-reconnaissables.
- Exemple d'un langage non Turing-reconnaissable.

8 Calculabilité

- Représentation des entiers en base b par des mots finis ; fonction $\text{rep}_b : \mathbb{N} \rightarrow \mathbb{B}_b^*$. Valeur numérique entière d'un mot ; fonction $\text{num}_b : \mathbb{B}_b^* \rightarrow \mathbb{N}$.
- Toute fonction $f : \mathbb{N} \rightarrow \mathbb{N}$ est représentable en base b par la fonction $\hat{f}_b : \mathbb{B}_b^* \rightarrow \mathbb{B}_b^*$ définie par

$$\hat{f}_b(\beta) = \text{rep}_b(f(\text{num}_b(\beta))).$$

- Une fonction $F : \mathbb{A}^* \rightarrow \mathbb{A}^*$ est Turing-calculable, s'il existe une machine de Turing M telle que $F = \text{Tur}_M$ (mêmes domaines, mêmes valeurs sur le domaine). ← Fin du cours du 13 mars 2020.

Épidémie Covid-19

Continuité pédagogique : Une version actualisée et corrigée de notes de cours sera mise à jour sur le serveur dans le courant de la semaine prochaine. Le chapitre sur la Complexité ne sera pas examiné. Cependant, Il est vivement recommander de l'étudier attentivement et me poser vos questions éventuelles par mail. Les solutions par visioconférence sont à l'étude (sans trop d'espoir car le matériel audiovisuel du laboratoire risque d'être lourdement sollicité par plusieurs autres activités).

Si les cours reprennent : il y aura au moins une séance pour expliquer les notions de complexité, répondre à des questions et faire des exercices.

Évaluation du module : elle se fera par un devoir personnel qui sera mis en ligne en fin de la semaine prochaine.