

# LifLF – Théorie des langages formels

*Sylvain Brandel*

*2019 – 2020*

[sylvain.brandel@univ-lyon1.fr](mailto:sylvain.brandel@univ-lyon1.fr)

# Fonctionnement

- CM : 10 x 1h30
  - Mardi 14h
  - Silence dans l'amphi
- TD : 6 x 1h30
  - Généralement lundi 8h
  - Début des TD lundi 16 septembre
- TP : 4 x 1h30
  - Généralement mardi 9h45, après le TD de LF
  - Début des TP lundi 30 septembre
- Lien fort avec LC

# TP

- En Coq
  - <https://softwarefoundations.cis.upenn.edu>
  - Liens étroits avec LC
- En cours de restructuration

# Evaluation

- <http://liris.cnrs.fr/sylvain.brandel/wiki/doku.php?id=ens:lifLF>
- Le plan réussite en licence n'est plus
- UE en CCI (Contrôle Continu Intégral)
  - ECA (2 sessions)
    - 1<sup>ère</sup> session mardi 10 décembre 2019 14h, *a priori* QCM
    - 2<sup>de</sup> session en juin 2020 ...
  - TP noté
    - Lundi 9 décembre 2019
  - Interros « surprise »

## De votre côté

- Travail personnel conséquent
- Se préparer à l'avance
- Ne pas attendre que les réponses viennent toutes seules
- Lisez vos mails ...
- Contactez-moi, par mail, précisez LF

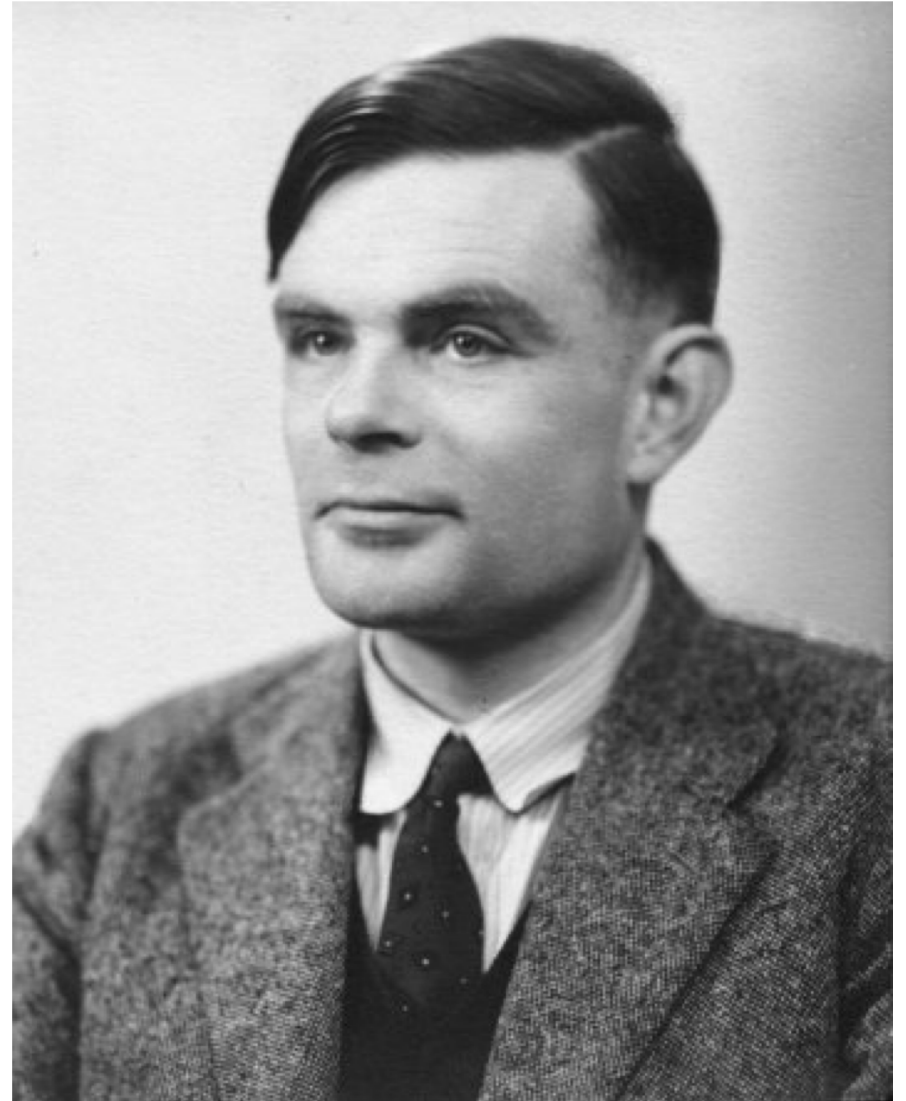
LifLF – Théorie des langages formels

*Sylvain Brandel*

2019 – 2020

[sylvain.brandel@univ-lyon1.fr](mailto:sylvain.brandel@univ-lyon1.fr)

# INTRODUCTION



# Motivations

- Informatique fondamentale
- Historiquement
  - Théorie de l'incomplétude
  - Que peut-on calculer avec un algorithme ?
- Lien avec les langages de programmation
  - Ce cours prépare à deux cours de master
    - Calculabilité et complexité
    - Compilation
- Vous intéresser ...
  - Si on sait qu'un problème est indécidable, inutile de chercher un algorithme pour le résoudre

# Comment

- Définition d'objets et d'ensembles
  - Par décision  $\rightarrow$  LifLF
  - Par construction  $\rightarrow$  LifLC, et aussi LifLF
- En LF : fonction de reconnaissance
- En LC : preuves de correction



# Programme

- Classifier des langages

Exemple d'école	Classe de langage	Reconnu par	Engendré par
$a^*b^*$	langages rationnels	automates à états finis	grammaire régulière
$\{a^n b^n \mid n \geq 0\}$	langages algébriques	automates à pile	grammaire algébrique
$\{a^n b^n c^n \mid n \geq 0\}$	langages rékursifs	machine de Turing	grammaire (générale)

- La décidabilité et la complexité en découlent

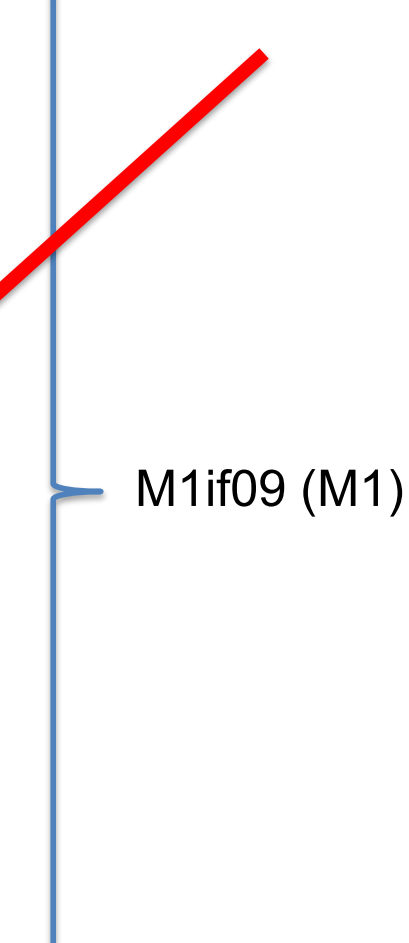
# Programme

## *Trame Lewis – Papadimitriou*

- Notions mathématiques de base
  - Ensembles
  - Alphabets, langages
- Langages rationnels
  - Grammaires régulières
  - Automates finis
  - Expressions régulières
- Langages hors contextes / algébriques
  - Grammaires algébriques
  - Automates à pile

# Programme

## *Trame Lewis – Papadimitriou*

- Machines de Turing
    - Formalisme de base
    - Langages rékursifs
    - Extensions
    - Machine de Turing Universelle
    - Grammaires
  - Indécidabilité
    - Thèse de Church – Turing
    - Problèmes indécidables
  - Complexité
    - Classes P, NP ...
    - NP-complétude
    - Théorème de Cook
- 
- M1if09 (M1)

# Programme

## *Prévisionnel*

- CM1 : Notions mathématiques de base
- CM2 : Alphabets et langages
- CM3 : Automates à états finis déterministes ou non
- CM4 : Élimination du non déterminisme
- CM5 : Caractérisation des langages rationnels
- CM6 : Minimisation des états
- CM6 : Langages rationnels, expressions régulières rationalité
- CM8 : Grammaires algébriques et langages algébriques
- CM9 : Automates à pile et algébricité
- CM10 : Analyse syntaxique

# Littérature

## **Elements of the Theory of Computation**

*Harry R. Lewis, Christos H. Papadimitriou*  
éd. Prentice-Hall

## **Introduction à la calculabilité**

*Pierre Wolper*  
éd. Dunod

## **Introduction to the Theory of Computation**

*Michael Sipser, MIT*  
éd. Thomson Course Technology

## **Introduction to Theory of Computation**

*Anil Maheshwari, Michiel Smid, School of Computer Science, Carleton University*  
free textbook

## **Gödel Escher Bach, les Brins d'une Guirlande Eternelle**

*Douglas Hofstadter*  
éd. Dunod

## **Logicomix**

*Apóstolos K. Doxiàdis, Christos Papadimitriou, Alecos Papadatos, Annie Di Donna*  
éd. Vuibert