

# IF04M050 : Théorie des langages

## Introduction

---

Jérôme Delobelle

`jerome.delobelle@u-paris.fr`

LIPADE - Université de Paris

- Cours : à distance tout le semestre
- TDs :

Groupes	Enseignant	Horaire	Salle
GP1	P. de Perthuis	Vendredi 14h-17h	(à distance)
GP2	M. Lhoumeau	Vendredi 15h15-18h15	(à distance)

**Les horaires de cours ou de TD sont susceptibles de changer au cours du semestre, vérifiez régulièrement votre emploi du temps sur l'ENT.**

- Contrôle continu  $CC$  :
    - Contrôle continu  $CC$  : **épreuve écrite** de 1h30 (*semaine du 15/03 ?*)
    - Examen  $E$  : **épreuve écrite** de 1h30
  - Note session 1 =  $\max(E; (E + CC) / 2)$
  - Note session 2 =
    - Si  $CC \succeq 10$  alors  $\max(E; (E + CC) / 2)$
    - Sinon  $E$
- Avec  $E$  = note de l'examen de la session concernée



# Motivations

- Concept de langage formel
  - Créé par les linguistes pour décrire les langages naturels
    - Effectuer des traductions automatiques
  - Noam CHOMSKY : modèle mathématique pour les grammaires
    - 1958 : langage ALGOL décrit à l'aide d'une grammaire "Hors-Contexte"

# Motivations

- Concept de langage formel
  - Créé par les linguistes pour décrire les langages naturels
    - Effectuer des traductions automatiques
  - Noam CHOMSKY : modèle mathématique pour les grammaires
    - 1958 : langage ALGOL décrit à l'aide d'une grammaire "Hors-Contexte"
- Pourquoi étudier les langages formels ?
  - Conception de langages : syntaxe, grammaire
  - Analyse
    - lexicale (déterminer les entités syntaxiques)
    - syntaxique (vérifier la syntaxe)
  - Compilation : traduction
  - Générateurs d'analyseurs lexicaux
  - Méta-compilateurs : compilateurs de compilateurs, ou générateur de compilateurs

# Motivations

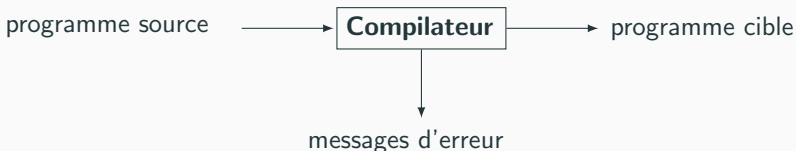
- Concept de langage formel
  - Créé par les linguistes pour décrire les langages naturels
    - Effectuer des traductions automatiques
  - Noam CHOMSKY : modèle mathématique pour les grammaires
    - 1958 : langage ALGOL décrit à l'aide d'une grammaire "Hors-Contexte"
- Pourquoi étudier les langages formels ?
  - Conception de langages : syntaxe, grammaire
  - Analyse
    - lexicale (déterminer les entités syntaxiques)
    - syntaxique (vérifier la syntaxe)
  - Compilation : traduction
  - Générateurs d'analyseurs lexicaux
  - Méta-compilateurs : compilateurs de compilateurs, ou générateur de compilateurs

⇒ Nécessaire pour l'étude des langages de programmation

# Un compilateur, c'est quoi ?

Un compilateur est un **programme** qui

- prend en entrée une **donnée textuelle** source (programme, donnée xml, fichier de configuration, etc)
- la **reconnâit** (l'analyse) pour vérifier sa correction
- émet éventuellement un **message d'erreur**
- le traduit dans un **langage cible**





- **Correction :**
  - Entrée invalide détectée ?
  - Sortie conforme aux attentes ?
  - ⇒ Outil de prédilection = **théorie des langage**
- **Efficacité :**
  - Faut-il *attennnnnnndre* le résultat de la compilation ?
  - ⇒ Outil de prédilection = **algorithmique**
- **Bonne conception du logiciel :**
  - Logiciel facile à modifier/étendre ?
  - ⇒ Outil de prédilection = **génie logiciel**

# A quoi sert la théorie des langages en compilation ?

Permet de **définir** rigoureusement et **reconnaître algorithmiquement** (pour les langages source et cible) :

- leur **vocabulaire** ou lexique : les mots autorisés
  - ⇒ analyse lexicale
  - ⇒ *automates à nombre fini d'états, expressions régulières*

# A quoi sert la théorie des langages en compilation ?

Permet de **définir** rigoureusement et **reconnaître algorithmiquement** (pour les langages source et cible) :

- leur **vocabulaire** ou lexique : les mots autorisés
  - ⇒ analyse lexicale
  - ⇒ *automates à nombre fini d'états, expressions régulières*
- leur **syntaxe** : la structure des phrases autorisées
  - ⇒ analyse syntaxique
  - ⇒ *automates à pile, grammaires algébriques*

# A quoi sert la théorie des langages en compilation ?

Permet de **définir** rigoureusement et **reconnaître algorithmiquement** (pour les langages source et cible) :

- leur **vocabulaire** ou lexique : les mots autorisés
  - ⇒ analyse lexicale
  - ⇒ *automates à nombre fini d'états, expressions régulières*
- leur **syntaxe** : la structure des phrases autorisées
  - ⇒ analyse syntaxique
  - ⇒ *automates à pile, grammaires algébriques*
- leur **sémantique** : la signification des phrases autorisées
  - ⇒ analyse sémantique
  - ⇒ *grammaires attribuées*

1. Alphabets et langages
2. Grammaires
3. Automates finis
4. Expressions régulières
5. Grammaires Hors-Contexte
6. Automates à pile
7. Analyse lexicale
8. Analyse syntaxique