

/SU/FSI/MASTER/INFO/MU4IN503 (APS)

# Analyse des Programmes et Sémantique

Février 2022

Pascal MANOURY – Romain DEMANGEON  
pascal.manoury@lip6.fr

Principes du cours

# Conception de langages

Reprise et poursuite des idées de MU4IN501 (DLP)  
*dans un cadre formel*

**Lietmotiv** : *syntaxe + typage + sémantique*

- ▶ **Syntaxe** Définition des chaînes de caractères admises comme *code source* des programmes
- ▶ **Typage** Définition du sous-ensemble des codes sources garantissant une cohérence des types
- ▶ **Sémantique** Définition du processus d'évaluation des codes sources

Règles

# Les langages *APS*

## *APS0* Noyau fonctionnel (pur)

Expressions, fonctions, expressions fonctionnelles

Fonctions récursives, expressions alternative

Types `int`, `bool`

Fermetures, fermetures récursives, liaison statique

## *APS1* Ajout d'un noyau impératif

Instructions, affectation, alternative, boucle

Type `void` Procédures

Mémoire, effet de bord

## *APS2* Structures de données

Listes et tableaux

## *APS3* Fusion fonctionnel/impératif

Rupture de contrôle, exceptions, continuations

# Mise en œuvre des langages *APS*

## **Le cours** édicte

1. les *règles* syntaxiques
2. les *règles* de typage
3. les *règles* sémantiques d'évaluation

## **Les TP** (TME) réalisent

1. les *analyseurs* lexicaux et syntaxiques
2. le *vérificateur* de types
3. *l'évaluateur* des programmes *APS*

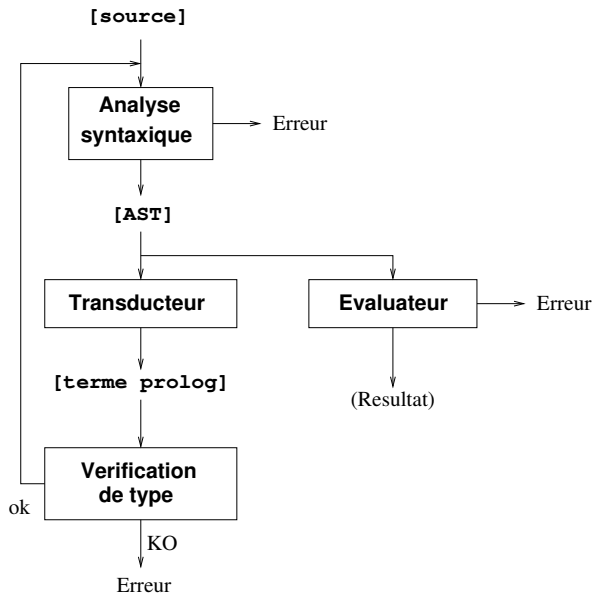
Pas de «feuilles de TP»

Un «projet»<sup>1</sup> tout au long du semestre

---

1. Réalisé sur le créneau TME du mercredi matin

# Schéma de principe



# Principes de réalisation

**Syntaxe** Utilisation de *générateurs* d'analyseurs lexicaux et syntaxiques

Outils `lex/yacc` (préférentiel)

Langage support libre

**Typage** Implantation des règles de typage

*Relations* de typage guidées par la syntaxe

Langage imposé : Prolog

**Sémantique** Implantation des règles sémantiques d'évaluation

Fonction d'évaluation par récurrence sur l'AST

Langage support libre