

Automates et grammaire (A&G) / Automata theory and grammars (A&G)



Composante
Polytech
Grenoble - INP,
UGA

- > **Langue(s) d'enseignement:** Français
- > **Ouvert aux étudiants en échange:** Oui
- > **Code d'export Apogée:** KARI5M06

Présentation

Description

L'enseignement "AUTOMATES et GRAMMAIRES" comporte 4 volets :

- La présentation de résultats fondamentaux de l'informatique : une technique de preuve de correction de programmes, différents modèles de calculs (séquentiel, parallèle, non-déterministe), la réalisation en un temps fini d'opérations sur des données infinies (les langages représentés sous forme d'automates), la récursivité
- L'illustration des ces notions dans le cas des automates et des grammaires
- Un exemple concret d'utilisation récente des automates : vérification de drivers, protocoles médicaux, personnage autonome dans les jeux vidéo, ...
- Un exemple concret d'utilisation des grammaires attribuées
- La présentation d'un résultat surprenant : les automates d'arbres, l'ordinateur quantique, les limites de l'informatique, ...

1. Preuve de correction partielle de programmes par la technique de Floyd-Hoare-Dijkstra : comment être sûr que un programme fait bien ce qu'on attend. Autrement dit, monterez-vous dans l'avion dont vous avez programmé le pilote automatique ?

2. Automates (à nombres d'états finis/à pile, déterministes/non-déterministes) : quel est le modèle de calcul d'un processeur ? y'a t'il des langages (des modèles) plus puissants que d'autres ?
3. Représentations équivalentes (des grammaires régulières aux équations d'Arden aux expressions régulières et finalement jusqu'aux automates à états finis) : comment passent-on d'une description lisible à une version utilisable par un processeur ?
4. Application et implantation des automates : les automates sont utilisés pour programmer (des analyseurs lexicaux, des micro-contrôleurs, des interfaces, des protocoles, des jeux,...), pour piloter (des chaînes de production, des systèmes cyber-physiques,...) et pour vérifier (des drivers, des politiques de sécurité ...) en fait on en trouve un peu partout.
5. Grammaires attribuées et génératives : les grammaires sont le quotidien des informaticiens qui ne cessent de traduire un langage ou des données d'un format vers un autre.
6. Application : implantation en ANTLR ou JavaCC d'un interpréteur d'expressions booléennes

The course "AUTOMATES and GRAMMARIES" has 4 components:

- À 1. The presentation of fundamental results of computer science: a technique of correctness proof of programs, different computation models (sequential, parallel, non-deterministic), the computation in finite time of operations on infinite data (the languages represented as automata), recursion
- À 2. The illustration of these notions in the case of automata and grammars
- À 3. A concrete example of recent use of automata: verification of drivers, medical protocols, autonomous character in video games, ...
4. A concrete example of the use of attributed grammars
5. The presentation of a surprising result: tree automata, the quantum computer, the limits of computing, ...

TARGETED SKILLS:

- know how to prove that a program is correct
- know how to program using an automaton
- know how to write a simple analyzer / translator

1. Proof of partial program correctness by the Floyd-Hoare-Dijkstra technique: how to be sure that a program is doing what is expected. In other words, will you get on the plane which runs an autopilot you programmed?
2. Automata (with finite state / stack numbers, deterministic / non-deterministic): what is the computation model of a processor? are there languages ?more powerful than others?
3. Equivalent representations (from Regular Grammars to Arden's Equations then to Regular Expressions and finally to Finite State Automata): how do we go from a readable description to a version usable by a processor?
4. Application and implementation of Automata (ie. Programmable Logic Controller): PLC are used to program (lexical analyzers, micro-controllers, interfaces, protocols, games, ...), to control (production chains, physical, ...) and to check (drivers, security policies ...) in fact we find them everywhere.
5. Attributed and Generative Grammars: Grammars are the everyday tool of developers who translate language or data from one format to another.

6. Application: implementation in ANTLR or JavaCC of an interpreter of Boolean expressions

Objectifs

Heures d'enseignement

Automates et grammaire (A&G) / Automata theory and grammars (A&G) - CMTD

Cours magistral - Travaux dirigés

42h

Pré-requis recommandés

Connaître les rudiments d'un langage impératif tel que C

Basic knowledge of an imperative language such as C

Période : Semestre 5

Évaluation initiale / Session principale - Épreuves

Libellé	Nature de l'enseignement	Type d'évaluation	Nature de l'épreuve	Durée (en minutes)	Nombre d'épreuves	Coefficient de l'épreuve	Remarques
						50/100	

Bibliographie

- Introduction à la calculabilité, Pierre Wolper, Éditions Dunod (3eme édition, 2006)

- Cours en ligne sur le site web A&G : <http://www-verimag.imag.fr/~perin/enseignement/RICM3/infaeg/>

Infos pratiques

Lieu(x) ville

› Grenoble

Campus

› Grenoble - Saint-Martin d'Hères