Titre / Title	**Advanced topics in software analysis and verification**

Enseignant(s) / Instructor(s)	**Kuncak Viktor: IN**	Langue / Language	**EN**

Programme(s) Période(s)	Nombre d'heures / Number of hours	Spéc / filière /orient	Type
Informatique (2008-2009, Master semestre 2)	**C: 4 H hebdo, TP: 2 H hebdo**	***B F***	**opt**
Systèmes de communication - master EPFL (2008-2009, Master semestre 2)	**C: 4 H hebdo, TP: 2 H hebdo**	***F***	**opt**
Systèmes de communication - master EPFL (2008-2009, Master semestre 4)	**C: 4 H hebdo, TP: 2 H hebdo**	***F***	**opt**

Objectifs:

Introduction à la vérification de logiciel: bases théoriques, algorithmes, outils.

Objectives:

The class will introduce foundations, algorithms, and tools for automated analysis and verification of complex properties of software systems.

Contenu:

Matières:
- Logic
- Sémantique de programme
- Génération d'état de vérification
- S'avérer automatisé de théorème
- Procédures de décision
- Interprétation abstraite
- Abstraction d'attribut
- Analyse d'indicateur
- Analyse de forme
- Analyse d'Interprocedural
- Construction de graphique d'appel
- Analyse des programmes concourants

Content:

Motivation:
Tools for automated analysis and verification of software can improve reliability of software that we use every day. The underlying techniques are also used for compiler optimizations and program understanding. In recent years, new algorithms and combinations of existing techniques have made such tools more effective than in the past. This course will give an overview of basic techniques, as well as the recent advances that made this progress possible.

Topics covered include:
- Logic and program semantics
- Verification condition generation
- Theorem proving and decision procedures
- Syntactic loop invariant inference
- Abstract interpretation and data flow analysis
- Predicate abstraction; shape analysis
- Modular verification
- Interprocedural analysis
- Analysis of object-oriented and concurrent programs
- Dynamic analysis; bug finding; loop unrolling

Prérequis:

Theory of Computation, Compiler Construction, and basics of Formal Logic are helpful but not required. If you are not familiar with these topics, please see the instructor.

Required prior knowledge:

Theory of Computation, Compiler Construction, and basics of Formal Logic are helpful but not required. If you are not familiar with these topics, please see the instructor.

Préparation pour:

Research and application of program analysis, verification, software reliability, and compilers.

Prerequisite for:

Research and application of program analysis, verification, software reliability, and compilers.

Forme d'enseignement:

The course will include lectures, exercises, paper discussions, mini project presentations, and possibly an invited lectures.

Type of teaching:

The course will include lectures, exercises, paper discussions, mini project presentations, and possibly an invited lectures.

Forme du contrôle:

Grading will be based on one mini project, weekly homeworks, class participation, and taking lecture notes. Students will participate in homework grading.

Form of examination:

Grading will be based on one mini project, weekly homeworks, class participation, and taking lecture notes. Students will participate in homework grading.

URLs	1) http://lara.epfl.ch		
Matière examinée / subjects examined	Session	Coefficient / Crédits ECTS	Forme de l'examen / Type of examination
Advanced topics in software analysis and verification	**ETE**	**6**	**Pendant le semestre**