

# 2EL1540 - Informatique théorique

Responsables : **Pascale LE GALL** , **Marc AIGUIER**

Département de rattachement : **DÉPARTEMENT INFORMATIQUE**

Langues d'enseignement : **FRANCAIS**

Type de cours : **Electif 2A**

Campus où le cours est proposé : **CAMPUS DE PARIS - SACLAY**

Nombre d'heures d'études élèves (HEE) : **60**

Nombre d'heures présentielles d'enseignement (HPE) : **30**

Année académique : **2024-2025**

Catégorie d'électif : **Sciences fondamentales**

Niveau avancé : **oui**

---

## Présentation, objectifs généraux du cours :

Cet enseignement donne une partie des fondements de la science informatique sur ses deux paradigmes de calcul que sont la réduction (calcul pas à pas) et la résolution (inférence logique/démonstration automatique).

## Période(s) du cours (n° de séquence ou hors séquence) :

SG8

## Prérequis :

Cours Algorithmes et Complexité (ST Modélisation)

Goût pour l'abstraction mathématique et pour le raisonnement

## Plan détaillé du cours (contenu) :

Il sera alors abordé dans ce cours les notions fondamentales sous-jacentes aux thèmes

- de l'induction et la récurrence (ensemble bien-fondé et équivalence avec l'induction mathématique). L'objectif est de formaliser les notions fondamentales d'induction et récursivité sous-jacentes à toutes les mathématiques discrètes.
- de l'algorithmique (fonctions récursives de Gödel/Herbrand, machine de Turing et tous les résultats d'indécidabilité associés). L'idée est de définir formellement (i.e. mathématiquement) ce qu'est un problème de décision et donner une dénotation formelle à la notion d'algorithme (thèse de Church).
- de la logique (syntaxe, sémantique et systèmes de preuves). L'intérêt des logiques est d'exprimer de façon formelle les propriétés des systèmes et de raisonner, de façon automatique ou assistée, à propos de ces propriétés. Les logiques propositionnelles et du premier ordre

seront détaillées.

Le cours est composé des parties suivantes :

- Fondements de l'induction et la récursivité.

On y abordera plus particulièrement les notions de : Théorie des ensembles : Ordre et préordre, Majorants et minorants, Ensembles bien fondés et induction; Systèmes formels, preuves, correction et complétude.

- Logique propositionnelle.

La logique propositionnelle sera présentée selon les composantes classiques : syntaxe, sémantique et preuve. En particulier, les points suivants seront abordés : arbres binaires de décision, méthode des tableaux, algorithme DPLL, satisfiabilité des formules propositionnelles (problèmes SAT et 3-SAT), SAT-solveurs, les systèmes de preuve de la résolution, déduction naturelle, séquents.

- Théorie de la calculabilité et de la complexité.

On verra plus précisément : les fonctions primitives récursives, les fonctions récursives, les problèmes décidables, indécidabilité du problème de l'arrêt, fonction récursive universelle (interpréteur) ; les machines de Turing , théorèmes d'équivalence, thèse de Church ; la théorie de la complexité en lien avec les modèles de calcul (complexité en temps, classes P et NP, structuration de la classe NP - problèmes NP-complets et NP-durs, un premier problème NP-complet) ; la décidabilité et NP-complétude du problème 3-SAT.

- Logique des prédicats

La logique des prédicats est une extension de la logique propositionnelle et est la logique privilégiée pour décrire les structures de données informatiques. On verra que la logique des prédicats est indécidable et la programmation logique sera introduite.

## Déroulement, organisation du cours :

Le cours sera divisé en 13,5h de cours magistraux et 15h de tds.

Un ou plusieurs travaux personnels (projet avec réalisation informatique ou problème à résoudre) seront proposés et constitueront la note de l'évaluation obligatoire.

## Organisation de l'évaluation :

L'évaluation se fera au moyen d'un projet et d'un examen écrit d'une durée de 2 heures.

La note finale sera partagée en 50% pour l'évaluation obligatoire et 50% pour l'examen écrit.

Pour cet examen, ne sont autorisés que le polycopié ainsi que les notes personnelles. Les dispositifs électroniques (ordinateurs portables, téléphones portables et tablettes) ne sont pas autorisés.

## Moyens :

- Equipe enseignante (noms des enseignants des cours magistraux) : Marc Aiguier et Pascale Le Gall
- Taille des TD : 35 élèves au maximum
- Outils logiciels et nombre de licence nécessaire : les seuls logiciels utilisés (prolog, solveurs, assistants à la preuve) seront des logiciels libres d'accès, que les élèves installeront sur leur machine personnelle

## Acquis d'apprentissage visés dans le cours :

Comprendre les principes fondamentaux et les outils formels (i.e. mathématiquement fondés) à la base de toutes les méthodes de conception, de vérification et d'implantation des systèmes informatiques.

Savoir formaliser un problème de nature informatique et maîtriser les outils fondamentaux, de nature théorique, méthodologique et logicielle, nécessaires pour raisonner à propos de ces formalisations.

Donner les outils théoriques fondés sur les modèles de calcul et utilisés pour l'analyse de complexité des algorithmes abordés dans le cours Algorithmes et Complexité, ainsi que les méthodes de raisonnement fondées sur la logique mathématique.

## Description des compétences acquises à l'issue du cours

:

Formalizing computing problems and mastering the fundamental theoretical tools necessary to reason about these formalizations. These theoretical tools are based on computation models classically used for the complexity analysis of the algorithms (see the course "Algorithms and Complexity"), as well as on reasoning methods based on mathematical logic.

## Bibliographie :

Il sera fourni aux élèves un polycopié ainsi que le sujet des tds et leur correction.