# 1 Mastère: "Méthodes formelles et fiabilité du logiciel" (Resp. Joëlle Despeyroux) *Draft - 30 avril 1999*

**Intervenants:** Robert De Simone, Joelle Despeyroux.

**Objectifs:** Former des chercheurs et des ingénieurs de haut niveau à la spécification et aux preuves sur machine. Les laboratoires de recherche des industries un peu partout dans le monde sont actuellement très demandeurs de personnel qualifié dans ce secteur. Et cette demande devrait encore grandir, vu que l'informatique est de plus en plus utilisée dans des domaines critiques pour la vie humaine (comme le nucléaire, l'industrie avionique ou ferrovière), ou bien dans des domaines où les enjeux financiers sont importants (comme le spacial, ou le domaine bancaire).

Les méthodes d'analyse et de preuves de propriétés de programmes se divisent en deux grandes familles complémentaires: les méthodes interactives (preuves par induction) et les méthodes automatiques (vérification exhaustive par modèles finis, réécriture). Les preuves de propriétés sémantiques et les preuves de correction de programmes utilisent des méthodes et des outils différents. On distingue aussi les preuves sur le logiciel des preuves sur le hardware; les preuves sur des langages impératifs, applicatifs, langages à objets, langages temps réels, etc...

L'objectif de cette filière est de couvrir le domaine des spécifications et des preuves sur machine le plus complètement possible, par la présentation (méthodes et outils) des approches les plus modernes et les plus utilisées et par la pratique de certaines d'entre elles.

**Plan du cours:**

1. Spécifications: sémantique, langages et méthodes:
   - sémantique des langages de programmation
   - mobilité et migration: pi-calcul, join, ...
   - langages réactifs (a)synchrones: Esterel, Statecharts, Lotos, ...
   - langages concurrents répartis et à objets: Java, ...
   - formalismes logiques: logique temporelle et Unity
   - méthodes assertionnelles: méthode B, VDM, ...
2. Preuves:
   - mécanisation du raisonnement
   - la théorie des types, le système Coq
   - autres systèmes de preuves: Isabelle, HOL, PVS
3. Analyse de programmes et algorithmiques de décision:
   - interpretation abstraite
   - model-checking
   - réécriture
4. Applications:
   - preuves de propriétés de langages
   - preuves de correction de programmes
   - développement d'algorithmes certifiés
   - protocoles de sécurité (Gilles Barthe)