Combinaison des approches algébrique et "à la Hoare" :
Preuve de concept

Proposition

Sujet de stage 3A école d'ingénieurs / M2 - 5 à 6 mois

Encadrement : Catherine Dubois, professeur ENSIIE, Samovar, Guillaume Burel, maître de conférences ENSIIE, INRIA Deducteam

Lieu du stage : Cachan ou Evry

Sujet

FoCaLiZe [2] est un environnement de développement de programmes certifiés : il permet de spécifier (dans un style algébrique à l'aide d'un langage du premier ordre), implanter (dans un style purement fonctionnel, avec un langage à la ML) et prouver (en utilisant le prouveur automatique Zenon). La compilation d'un programme FoCaLiZe produit un code exécutable en OCaml. Il produit également un code Dedukti [1] ou Coq à des fins de vérification des preuves fournies par Zenon.

Par ailleurs il existe de nombreux outils permettant de vérifier du code impératif ou orienté-objets basés sur la logique de Hoare, ou plus précisément la génération de conditions de vérification (Why3, FramaC/WP, etc). Dans ce cadre, la preuve des obligations de preuve générées assure la correction du programme par rapport à sa spécification écrite sous la forme d'un contrat (pré-condition/post-condition).

L'objectif du stage est d'étudier la faisabilité d'une approche permettant l'utilisation conjointe de FoCaLiZe et d'un prouveur "à la Hoare" (Why3 par exemple). Dans un premier temps, on adoptera une approche où l'on fait confiance au prouveur "à la Hoare", dans un deuxième temps on cherchera à expérimenter une approche *sceptique* où l'on cherchera à vérifier la correction de l'ensemble.

Pré-requis minimaux : programmation fonctionnelle, notions de base en logique et preuve de programmes. La pratique d'un outil de vérification déductive est un plus qui serait apprécié.

Ce sujet peut donner lieu à une poursuite en thèse.

References

[1] Raphaël Cauderlier, Catherine Dubois, ML pattern-matching, recursion, and rewriting: from FoCaLiZe to Dedukti, ICTAC 2016

[2] FoCaLiZe. voir le site `http://focalize.inria.fr`