

## Logique de séparation et vérification de programmes

---

**Encadrement & Contact :** Xavier RIVAL

*e-mail* : rival@di.ens.fr

tél : 01 44 32 21 50

**Description de l'objet du mémoire :**

La vérification de programme a pour but de formaliser des preuves de correction de fragments de code, à l'aide d'annotations qui peuvent être soit écrites à la main, soit calculée par un analyseur statique. La logique de Hoare [1] formalise de telles annotations sous la forme  $\text{pre}C\text{post}$  où  $C$  est un fragment de programme,  $\text{pre}$  une pré-condition et  $\text{post}$  une post-condition. Dans le cas de programmes manipulant des structures de données avec pointeurs telles que des listes ou des graphes, il est particulièrement difficile de traiter le cas des *alias*, au cours des preuves. La logique de séparation [2] propose une description très fine de la structure de la mémoire, et apporte une solution à ce problème. Ce sujet d'exposé propose d'étudier ce problème et le principe de la logique de séparation.

**Références bibliographiques :**

- 1 C. A. R. Hoare.  
An axiomatic basis for computer programming.  
Communications of the ACM, October 1969.
- 2 J. Reynolds.  
Separation logic : A logic for shared mutable data structures.  
In *LICS*, pages 55–74, 2002.