Seminário de Teoria da Computação

Strong call-by-need

Prof. Thiabaut Balabonski
Univ. Paris-Saclay, LRI, VALS team

07/05/21
10:00 Horas

Zoom Meeting

Abstract. Call-by-need is an evaluation strategy for the lambda-calculus, which models computation in lazy functional programming languages. Its defining feature is ensuring that each argument of a function is evaluated only once, and only if its value is actually needed.

As any other evaluation strategy, call-by-need is mainly concerned with producing values. As such it only performs weak reduction, that is it never reduces inside a lambda-abstraction. On the other hand, other applications based on the lambda-calculus like partial evaluators or proof assistants require computing normal forms, rather than simply values. For this they need strong reduction.

This talk introduces a calculus that enforces the principles of call-by-need evaluation while allowing strong reduction. The calculus is sound and complete with respect to usual beta-reduction: it computes the same normal forms as the lambda-calculus, in a by-need fashion. I will also relate how formalizing the calculus in a proof assistant (here: Abella) led to significant improvements in the pen-and-paper definitions.