A Functional Programming Approach to Teaching in Portuguese Foundational Computing Course
Part 3: Formally Proved Algorithmic

Proposta de Mestrado

Orientador: Simão Melo de Sousa
Co-orientador António Ravara

1 Context

This MSc Dissertation proposal aims at promoting the use of OCaml in the Portuguese speaking academic community, namely by providing support to teaching approaches and tools. In particular it aims at extending and consolidating the OCaml software user base and teaching materials in the Portuguese language for Computational Logic and Foundation of Computing courses in undergraduate Computer Science degrees.

In short, the proposal aims at the development in OCaml of an automated marking platform offering exercises in Portuguese of First-order Logic and Language/Automata Theory. The idea is to make available OCaml implementations of classical algorithms like conversion of formulae to normal forms, satisfiability checking, inter-conversion of automata and grammars, etc. We will re-use existing implementations and develop new ones when no solution is available. We will build on well-known and mature platforms like TryOCaml, France-IOI and JFLAP. The algorithms will be in the back-end of an automated grading platform to assist our courses.

This proposal is subject to a one year grant and will be funded by the Tezos Foundation.

2 Objectives

The MSc student is expected to implement classical algorithms, but here within the use of a proof system and focus in the computational logic course. The idea here is to explore the use of proof tools like Coq or Why3 to specify the basic logical concepts exposed in the lecture, to allow the correct-by-construction execution of those which have an algorithmic counterpart (for instance, to verify the equivalence of two propositional formulas by verifying that both result in the same Boolean function or the same Boolean constant, etc.) and to prove them correct (within the logic supported by the proof tool). The supervision team has been experimenting this approach on only a very few exposed algorithms (namely, the Horn algorithm) but with promising results and impact in the classroom. One expected output is the ability to propose exercises where the students experiment with machine checked proofs of results exposed during the lecture and even to derive correct-by-construction OCaml implementation (which possible using tools like Why3 or COQ).

3 Plano

- Sept. - Nov. : Technological review and state of the art.
- Dec - Jan. : Architectural design fo the proposed solution.
- Fev. - Mar. : Implementation.
- Apr. Mid-May: Proof-of-concept, Validation and Verification.
- May - Jun. : Dissertation writing.

4 Contact

Simão Melo de Sousa (desousa@di.ubi.pt)