

# Diogo Miguel Gaspar de Sousa

---

## CONTACT INFORMATION



*Email:* [diogogsousa@gmail.com](mailto:diogogsousa@gmail.com) PGP Fingerprint: F39A 0709 65D4 DA2E DE85  
0A7D DE93 F053 6CB1 A1D5  
*www:* <http://orium.pw> Github: <https://github.com/orium>

## PROFESSIONAL EXPERIENCE

 *Feedzai* (2014–2017)

Worked as a *Scala* Back-end Software Developer in a small team that developed *Feedzai Cloud*. Feedzai Cloud is a fraud prevention service that uses machine learning to score transaction according to their likelihood of being fraudulent.

## EDUCATION

  *Faculdade de Ciências e Tecnologia* (Faculty of Sciences and Technology) of *Universidade Nova de Lisboa* (New University of Lisbon), Portugal

*Mestrado em Engenharia Informática* (Computer Science Master's Degree) (2011–2013)

- Obtained Master's degree with a final grade of 17.
- Master's thesis grade of 19. In this thesis we developed a static analysis to detect potential atomicity violations when using libraries. The thesis is [publicly available](#) as well as the [prototype](#). The paper “Preventing Atomicity Violations with Contracts” covers the main contributions of the thesis (see “Publications”).
- 123 ECTS obtained (one extra course taken).

*Licenciatura em Engenharia Informática* (Computer Science Bachelor's Degree) (2008–2011)

- Obtained Bachelor's degree with a final grade of 17.
- 186 ECTS obtained (one extra course taken).
- Received an [excellence scholarship](#) for the third year average grade of 17.78.



*Universiteit Utrecht* (Utrecht University), Netherlands

*Applied Functional Programming* — A two-week summer course on functional programming with *Haskell* (2010)

- Developed [regex2dot](#), a regular expression to finite state automaton compiler. The program can output the image of the non-deterministic, deterministic or optimum finite state automaton.

 *BerkeleyX*, edX

CS191x: *Quantum Mechanics and Quantum Computation* — A ten-week online course on quantum computing (2013)

- Completed with a final grade of 94%.



## 2014

FCT Scholarship under the supervision of Prof. João Lourenço improving the atomicity violation detection tool developed in the master thesis. This tool is described in “Preventing Atomicity Violations with Contracts” (see “Publications”).

Participation as volunteer in 2014 Expo FCT with the *Haskell* programming activity.

## 2012–2013

FCT Scholarship under the supervision of Prof. João Lourenço improving an atomicity violation detection tool. This tool is described in “A Static Approach for Detecting Concurrency Anomalies in Transactional Memory” and the analysis implemented is presented in “Using program closures to make an API implementation thread safe” (see “Publications”).

Presentation of the paper “Prevenção de Violações de Atomicidade usando Contratos” (Preventing atomicity violation using contracts) in INForum2013.

Attended 2013 HTDC winter school on transactional memory.

Participation as volunteer in 2013 Expo FCT with the *Haskell* programming activity.

## 2011–2012

Teacher of the practical lessons of Introduction to Programming, a first semester course of the Computer Science Bachelor’s Degree.

ROPES scholarship (Research Opportunities Programme for Excelling Students) with one semester duration, under the supervision of Prof. João Lourenço in a unified system for databases and memory transactions.

Presentation of the paper “Aplicação do Fecho de Programas na Detecção de Anomalias de Concorrência” (Using program closures to detect concurrency anomalies) and “Paralelização de Código Puro numa Linguagem Imperativa” (Parallelization of Pure Code in an Imperative Programming Language) in INForum2012.

Participation as volunteer in 2012 Expo FCT with the *Haskell* programming activity.

## 2010-2011

ROPES scholarship (Research Opportunities Programme for Excelling Students) with one semester duration, under the supervision of Prof. João Lourenço. This project extended an existing tool that statically detect high-level data races by inferring correlations between variable. The approach used by this tool is covered in “Detecting Concurrency Anomalies in Transactional Memory Programs” (see “Publications”).

Presentation of the paper “A Static Approach for Detecting Concurrency Anomalies in Transactional Memory” in INForum2010.

Creation of a new activity of functional programming using *Haskell* for the Expo FCT, and participation as volunteer in the event.

## 2009–2010

*UROP* scholarship (Undergraduate Research Opportunities Program), with one year duration under the supervision of Prof. João Lourenço. Various programs with concurrency problems were analyzed and solutions using transactional memory was proposed.

Participation as volunteer in 2010 Expo FCT.

## 2008–2009

Integration with a group of students of Prof. João Lourenço, in the transactional memory subject, where two main contributions were made: porting *CTL* (*Consistent Transactional Layer*, a library for transactional memory) to the *GNU Autotools* build system, and adapted the file system benchmark *IOZone* to the *Transactional File System* (developed by another member of the research group).

Participation as volunteer in 2009 Expo FCT.

Participation as volunteer in 2009 Clube Math.

## PUBLICATIONS

Ricardo J. Dias, Carla Ferreira, Jan Fiedor, João M. Lourenço, Aleš Smrcka, Diogo G. Sousa, Tomáš Vojnar. Verifying Concurrent Programs Using Contracts. *IEEE International Conference on Software Testing, Verification and Validation 2017*. doi:10.1109/ICST.2017.25.

Diogo G. Sousa, Ricardo J. Dias, Carla Ferreira, João M. Lourenço. Preventing Atomicity Violations with Contracts. *arXiv*, 2015. arXiv:1505.02951.

Diogo G. Sousa, Carla Ferreira, João Lourenço. Prevenção de Violações de Atomicidade usando Contratos (*Preventing Atomicity Violations with Contracts*, english version above). *Proceedings of INForum 2013*.

Diogo G. Sousa, João C. Martins and João Costa Seco. Paralelização de Código Puro numa Linguagem Imperativa (*Parallelization of Pure Code in an Imperative Programming Language*). *Proceedings of INForum 2012*.

Eitan Farchi, Itai Segall, João Lourenço and Diogo Sousa. Using program closures to make an application programming interface (API) implementation thread safe. *PADTAD 2012*. doi:10.1145/2338967.2336810.

Vasco Pessanha, Ricardo J. Dias, João Lourenço, Eitan Farchi, Diogo Sousa. Practical verification of high-level dataraces in transactional memory programs. *PADTAD 2011*. doi:10.1145/2002962.2002968.

João Lourenço, Diogo Sousa, Bruno Teixeira and Ricardo Dias. Detecting Concurrency Anomalies in Transactional Memory Programs. *Computer Science and Information Systems*, Vol. 8, No. 2, 2011. doi:10.2298/CSIS110110007L.

Bruno C. Teixeira, João Lourenço, Eitan Farchi, Ricardo J. Dias, Diogo Sousa. Detection of Transactional Memory anomalies using static analysis. *PADTAD 2010*. doi:10.1145/1866210.1866213.

Bruno Teixeira, João Lourenço and Diogo Sousa. A Static Approach for Detecting Concurrency Anomalies in Transactional Memory. *Proceedings of INForum 2010*.

PROGRAMMING  
CONTESTS

[Southwestern Europe Regional Contest](#)

- Participation in the [2012 Southwestern Europe Regional Contest](#), ending in 39th place (of 44).
- Participation in the [2011 Southwestern Europe Regional Contest](#), ending in 23th place (of 42).

[Inter-University Programming Marathon \(Portugal — National finals\)](#)

- Participation in the [2013 Inter-University Programming Contest](#), ending in 8th place (of 13).
- Participation in the [2012 Inter-University Programming Contest](#), ending in 5th place (of 16). Selected for the [2012 Southwestern Europe Regional Contest](#).
- Participation in the [2011 Inter-University Programming Contest](#), ending in 3rd place (of 25). Selected for the [2011 Southwestern Europe Regional Contest](#).
- Participation in the [2010 Inter-University Programming Contest](#), ending in 14th place (of 18).
- Participation in the [2009 Inter-University Programming Contest](#), ending in 8th place (of 14).

[Inter-University Programming Contest \(Portugal\)](#)

- Participation in the [2013 Inter-University Programming Contest](#).
- Participation in the [2012 Inter-University Programming Contest](#).
- Participation in the [2011 Inter-University Programming Contest](#).
- Participation in the [2010 Inter-University Programming Contest](#).
- Participation in the [2009 Inter-University Programming Contest](#).

[International Olympiads in Informatics](#)

- Obtained 45 points (out of 600) in the [2008 International Olympiads in Informatics](#).

[Correspondence Iberoamerican Informatics Competition](#)

- Bronze medal in the [2008 Correspondence Iberoamerican Informatics Competition](#).

[National Olympiads in Informatics \(Portugal\)](#)

- [6th place](#) in [2008's](#) finals.  
Selected to the [2008 International Olympiads in Informatics](#), by two more contests in the University of Porto, which select the contestants with the four best scores.
- [3rd place](#) in [2007's](#) finals.

SOFTWARE  
PROJECTS

List of projects created or contributed to. Most of these projects are of a didactic nature, and are not maintained. All projects are under some version of the *GNU General Public License* or *GNU Lesser General Public License*.

[RPDS](#) (2017), a fully persistent data structure library in *Rust*. Currently implementing a classical functional list, a vector with fast indexing (this data structure does not really have a name, but if I had to name it it would be “compact index trie”; it is used in *Clojure* and *Scala*; implementation details [here](#)), a hash array mapped trie (also used in *Clojure* and *Scala*), a red-black tree, and a queue (implementation details [here](#)).

[Strava PR](#) (2017), a tool written in *Scala* that analyzes runs from your [Strava](#) account to give some insights on your personal records and how they evolved through time.

Several [contributions](#) to the [Hexchat](#) irc client (2013). In particular the support for the irc extension [server-time](#) and improvements in *url* recognition.

[PinguescoFS](#) (2011), a simple, persistent, file system. *PinguescoFS* is written in *C* and runs in userspace with *FUSE*.

*CDSC* (2011), a data structures library written in *C* that provides interfaces and implementation of several data structures in an object-oriented manner. Contains interfaces for map, ordered map, queue, stack, priority queue and disjoint sets.

*HTFS* (2008), a userspace volatile file system using *FUSE*. The file system was written in *C* and uses *hashtables* for all lookups.

*DBFI* (2008), a *brainfuck* interpreter and compiler written in *C*. The compiler generates *x86 assembly* with *GNU* syntax.

*Brute* (2008), a multithreaded brute force hash cracker written in *C*. The application is easily extensible to support new cryptographic hashing algorithms through a module system.

*Potatochess* (2008), a *P2P* chess game (with *IPv4* and *IPv6* support). The application was written in *C++*, and uses the *GTKmm* and *glademm* libraries for the graphical interface.

*Freeircd* (2007), an irc server written in *C*. The server supports *IPv6* and uses a *MySQL* database.

#### TECHNICAL SKILLS

Programming languages: *C*, *Scala*, *Rust*, *Java*, *Python*, *Shell scripting*, *C++*.

Some contact with programming languages (academic or small projects): *Haskell*, *x86 assembly*, *Ocaml*, *Idris*, *Spec#*, *Coq*, *Prolog*, *Perl*.

Other languages: *SQL*, *L<sup>A</sup>T<sub>E</sub>X*.

#### MISC

About 300 programming challenges solved in *UVa Online Judge*.

More about me on [my personal website](#).