Luca **Negrini**

Software Engineer · Static Analysis Specialist

Email | Homepage

Experience

Corvallis Srl (previously JuliaSoft Srl) *Verona, Italy*

Software Engineer & Research Scientist *Apr 2018 - present*

- Development of the Julia Static Analyzer (now part of CodeSonar by GrammaTech, Inc.): software development and engeneering in Java and C#
- Research topics: static analysis of object oriented software, tools for static analysis, abstract interpretation

Miscellaneous experience

2016-2017	**Software Developer**, Click Realtà Virtuale, Cerea (VR), Italy
Jun-Aug 2011	**Employee**, ABM Sistemi di Bellani Marco, Nogara (VR), Italy
Jun-Aug 2010	**Farmhand**, Farmacia delle Piante, Gazzo Veronese (VR), Italy
Jun-Jul 2009	**Internship**, ABM Sistemi di Bellani Marco, Nogara (VR), Italy

Education

Università Cà Foscari Venezia *Venice, Italy*

PhD in Computer Science *Sept 2019 - present*

Università degli Studi di Verona *Verona, Italy*

Master Degree in Computer Science, 110/110 cum laude *Sept 2015 - Mar 2018*

Università degli Studi di Verona *Verona, Italy*

Bachelor Degree in Computer Science, 90/110 *Sept 2012 - Mar 2016*

ITIS Guglielmo Marconi *Verona, Italy*

High School Diploma in IT, 93/100 *Sept 2007 - Jun 2012*

Publications

Conferences and Workshops

FNAC21 P. Ferrara, L. Negrini, V. Arceri, A. Cortesi. 2021. *"Static analysis for dummies: experiencing LiSA"*. In Proceedings of the 10th ACM SIGPLAN International Workshop on the State Of the Art in Program Analysis (SOAP 2021). Association for Computing Machinery, New York, NY, USA, 1–6.

NAFC21 L.Negrini, V. Arceri, P. Ferrara, A. Cortesi, *"Twinning Automata and Regular Expressions for String Static Analysis"*, in Verification, Model Checking, and Abstract Interpretation. VMCAI 2021. Lecture Notes in Computer Science, vol 12597. Springer, Cham.

FN20 P. Ferrara and L. Negrini, *"SARL: OO Framework Specification for Static Analysis"*, in Software Verification. Springer, Cham, 2020. pp. 3-20.

NF18 L. Negrini and P. Ferrara, *"SARL: Framework Modeling for Static Analysis"*, in Proceedings of the 9^{th} Workshop on Tools for Automatic Program Analysis (TAPAS 2018), Freiburg im Breisgau, Germany, August 28, 2018

Talks

Conferences and Workshops

18/01/2021	Twinning Automata and Regular Expressions for String Static Analysis, VMCAI 2021, Virtual conference
20/07/2020	SARL: OO Framework Specification for Static Analysis, VSTTE 2020, Virtual conference
28/08/2018	SARL: Framework Modeling for Static Analysis, TAPAS 2018, Freiburg im Breisgau, Germany

Seminars

30/06/2021	Modular Multi-language analysis in LiSA, Università Cà Foscari Venezia, Online seminar

Teaching

Mar-May 2021	**Lectures (4 hrs), Online tutor**, Università Cà Foscari Venezia, "Software Correctness, Security and Reliability"
Mar-May 2020	**Online tutor**, Università Cà Foscari Venezia, "Software Correctness, Security and Reliability"

Community Service

Reviewer activities

VALID 2021	The Thirteenth International Conference on Advances in System Testing and Validation Lifecycle

Projects

LiSA

As part of my PhD research project, I started the development of LiSA, together with the Software and System Verification group @ Ca' Foscari University of Venice, Italy. LiSA (Library for Static Analysis) eases the creation and implementation of static analyzers based on the Abstract Interpretation theory. LiSA provides an analysis engine that works on a generic and extensible control flow graph representation of the program to analyze. Abstract interpreters in LiSA are built for analyzing such representation, providing a unique analysis infrastructure for all the analyzers that will rely on it.

Building an analyzer upon LiSA boils down to writing a parser for the language that one aims to analyze, translating the source code or the compiled code towards the control flow graph representation of LiSA. Then, simple checks iterating over the results provided by the semantic analyses of LiSA can be easily defined to translate semantic information into warnings that can be of value for the final user.

LiSA is distributed under the MIT license, and is available on GitHub.

Tarsis

Tarsis is a new abstract domain for string values based on finite state automata. Standard finite state automata abstract domain has been shown to provide precise abstractions of string values when all the components of such strings are known, but with high computational cost. Instead of considering standard finite automata built over an alphabet of single characters, Tarsis considers automata that are built over an alphabet of strings, comprising a special value to represent statically unknown strings. Tarsis is maintained by the Software and System Verification group @ Ca' Foscari University of Venice, Italy,and it is available on GitHub.

Master Thesis

Title	Automatic Application Splitting
Supervisor	Prof. Fausto Spoto
Co-supervisor	Pietro Ferrara, PhD
Description	Design and implementation of advanced algorithms and application of machine learning to obtain automatic application splitting for scaling up interprocedural static analyses to industrial software

Languages

Italian	Mother language
English	Intermediate spoken and written

Interests

Professional	Software Engineering, Software Development
Scientific	Program Verification, Static Analysis, Abstract Interpretation, Cybersecurity