Programming in Scala

Scala is a JVM-based programming language that nicely blends object oriented and functional programming constructs. Scala is used today extensively on the server side as well as on many analytics frameworks such as Spark and Flink. This course will introduce you to Scala programming putting emphasis on its construct that support functional programming. As in my experience, to really learn a programming language you need to program a sufficiently complex application, this class will be divided into a series of lectures and a personal project through which you will develop a non-trivial application.

The course will be organised as follows:

Lecture 1 (22 Nov 2017 / 15.00 — 19.00)

- Scala Overview
- Programming Paradigms
    - Object Oriented Programming
    - Functional Programming

Lecture 2 (6 Dec 2017 / 15.00 — 19.00)

- The Scala Programming Language (Part I)
    - Expressions
    - Conditions
    - Iterations
    - Objects
    - Functions
    - Pattern Matching
    - Type Parametrisation
    - Implicit Conversions

Lecture 3 (13 Dec 2017 / 15.00 — 19.00)

- The Scala Programming Language (Part II)
    - Expressions
    - Conditions
    - Iterations
    - Classes and Objects
    - Functions and Closures
    - Pattern Matching
    - Type Parametrisation
    - Implicit Conversions

Lecture 4 (15 Dec 2017 / 08.00 — 12.00)

- Functional Programming in Scala
    - List comprehension
    - For-Constructs
    - Folding, mapping, etc.
    - Monads

- Concurrent and Parallel Programming in Scala *(if time allows)*
    - Parallel Collections
    - Futures

Programming Project

The programming project will be agreed with the students. A list of projects will be distributed before the first lecture. Students can also propose submit a proposal at **angelo@icorsaro.net** using as subject **“ESIEE — Scala Programming Project Proposal”**. Keep in mind that the project has a credit of 16 hours, thus the complexity should be sufficient to justify the credit. Projects are individual, in other terms not groups are allowed.

Some example project are:

- Sudoku Generator and Solver (with GUI)
- Scalaman (A Scala version of the vintage Pacman Game)
- First and second order Statistics Library with support for parallel computation
- Web Crawler
- Simple QUIC Server
- Simple HTTP Server
- …

Grading

- Graded Test
- Individual Project