San José State University
College of Science/Department of Computer Science
CS152, Programming Paradigms, Sections 1 & 2
Fall Semester, 2018

Course and Contact Information

Instructor:	Jon Pearce
Office Location:	416 MacQuarrie Hall
Telephone:	(408) 924-5065
Email:	jon.pearce@sjsu.edu
Office Hours:	TR 13:30 – 15:00
Class Days/Time:	section 1 : TR 10:30 – 11:45 section 2 : TR 12:00 – 13:15
Classroom:	222 MacQuarrie Hall
Prerequisites:	C- or better in CS 46B and CS 151

Course Description:

Programming language syntax and semantics. Data types and type checking. Scope, bindings, and environments. Functional and logic programming paradigms, and comparison to other paradigms. Extensive coverage of a functional language.

Section Description:

After an overview of the major concepts of programming language syntax and semantics, the Scala language will be covered in depth. Scala is growing in popularity; it provides a nice introduction to the Functional Programming Paradigm as well as a 21st Century look at the Object-Oriented Paradigm. The second half of the course introduces Jedi, a simple but powerful experimental language. Using Scala as a meta-language, students will write parsers, compilers, and reference interpreters for various subsets of Jedi. Prolog will also be introduced as an example of the logic programming paradigm. Students will write an interpreter for the Proplog subset of Prolog.

Course Learning Outcomes:

Upon successful completion of this course, students will be able to:

1. Have a basic knowledge of the history of programming languages
2. Have a basic knowledge of the procedural, object-oriented, functional, and logic programming paradigms
3. Understand the roles of interpreters, compilers, and virtual machines
4. Critique the design of a programming language

5. Read and produce context-free grammars
6. Write recursive-descent parsers for simple languages, by hand or with a parser generator
7. Understand variable scoping and lifetimes
8. Write interpreters for simple languages that involve arithmetic expressions, bindings of values to names, and function calls
9. Understand type systems
10. Understand the implementation of procedure calls and stack frames
11. Produce programs in a functional programming language in excess of 200 LOC

Required Texts/Readings

Textbook

Lecture note and other materials will be posted at CS152 Course Website:

http://www.cs.sjsu.edu/faculty/pearce/modules/courses/Sp18/CS152/index.htm

Other Readings

David Watt, Programming Language Concepts and Paradigms, Prentice Hall, 1990

Friedman, Wand and Haynes, *Essentials of Programming Languages*, 2nd ed., MIT Press 2001

Lohr, *Go To: The Story of the Math Majors, Bridge Players, Engineers, Chess Wizards, Maverick Scientists and Iconoclasts--The Programmers Who Created the Software Revolution*.

Horstmann, *Scala for the Impatient*, 2 ed; Addisson-Wesley, 2016.

Other equipment / material requirements

Students should bring laptops to class. The following software should be installed:

- Scala Eclipse
- SWI Prolog

Course Requirements and Assignments

SJSU classes are designed such that in order to be successful, it is expected that students will spend a minimum of forty-five hours for each unit of credit (normally three hours per unit per week), including preparing for class, participating in course activities, completing assignments, and so on. More details about student workload can be found in University Policy S12-3 at http://www.sjsu.edu/senate/docs/S12-3.pdf.

Many of the assignments require students to implement parsers and reference interpreters in Java or Scala for various experimental languages. No prior knowledge of Scala will be assumed. Several weeks will be spent introducing students to Scala, a language which supports multiple paradigms. Thus, students will gain experience with different programming paradigms while learning programming language concepts through the reference interpreters they will write. These assignments enable the following CLOs: 2, 3, 5, 6, 7, 8, 9, 10, and 11. The actual assignments and their tentative due dates are posted below.

Grading Scheme

Course grades will be determined by computing a weighted average of all submitted work using the following weights:

Assignments	60%
Midterm	15%

Final	25%
TOTAL	100%

Weights of individual assignments appear on the course web page.

Assuming a normal distribution of weighted averages, I will use the following scheme for computing letter grades:

A	94% - 100%
A-	90% - 93%
B+	87% - 89%
B	84% - 86%
B-	80% - 83%
C+	77% - 79%
C	74% - 76%
C-	70% - 73%
D+	67% - 69%
D	64% -66%
D-	60% - 63%
F	0% - 59%

Classroom Protocol

Students should bring laptops to class and be prepared to work together on in-class labs.

University Policies

Per University Policy S16-9, university-wide policy information relevant to all courses, such as academic integrity, accommodations, etc. will be available on Office of Graduate and Undergraduate Programs' Syllabus Information web page at http://www.sjsu.edu/gup/syllabusinfo/

Course Schedule

Below is a tentative schedule of topics and activities. The instructor reserves the right to make changes to the schedule with fair warning. Exact due dates of assignments are given on the Assignments Page (see below).

Week	Dates	Topics/Activities	Assignment
1	8/21, 8/23	Bit of history. Programming paradigms.	
2	8/28, 8/30	Language processors & concepts. Scala basics.	
3	9/4, 9/6	Language concepts. Scala basics.	1
4	9/11, 9/13	Recursion in Scala.	2
5	9/18, 9/20	Functional programming in Scala.	3
6	9/25, 9/27	Scala collections. OOP in Scala.	4

7	10/2, 10/4	OOP in Scala.	
8	10/9, 10/11	Midterm review. Midterm.	
9	10/16, 10/18	Language processing in Scala.	5
10	10/23, 10/25	Language processing in Scala. Jedi 1.0.	
11	10/30, 11/1	Jedi 1.0.	6
12	11/6, 11/8	Jedi 2.0, Jedi 3.0	
13	11/13, 11/15	Jedi 4.0	7
14	11/20, T-Day	Prolog.	8
15	11/27, 11/28	Prolog.	
16	12/4, 12/6	Prolog. Final review.	9
17	12/12, 12/14	CS152 sec1 final: 12/12, 9:45 - 12; sec 2 final: 12/14, 9:45 - 12	

Assignment details can be found at:

- Assignments