SJSU SAN JOSÉ STATE UNIVERSITY

College of Science · Computer Science

Programming Paradigms Section 01

CS 152

Fall 2024 3 Unit(s) 08/21/2024 to 12/09/2024 Modified 08/20/2024

Contact Information

Prof. Thomas H. Austin

Email: thomas.austin@sjsu.edu
Office: MacQuarrie Hall 216

Office Hours

Mondays noon-1pm (in office).

Fridays 8-9am (Zoom only. https://sjsu.zoom.us/j/3796767168?pwd=SzNVOE4zSTNyNHNqR1RhNlJ6cDAwUT09 (https://sjsu.zoom.us/j/3796767168?pwd=SzNVOE4zSTNyNHNqR1RhNlJ6cDAwUT09)).

Other times by appointment.

See http://www.cs.sjsu.edu/~austin/office-hours-updates.txt (http://www.cs.sjsu.edu/~austin/office-hours-updates.txt) for notes on cancelled/rescheduled office hours.

Course Information

Monday, Wednesday, 10:30 AM to 11:45 AM, MacQuarrie Hall 222

Course Description and Requisites

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.

Prerequisite: CS 151 or CMPE 135 (with a grade of "C-" or better); Allowed Majors: Computer Science or Software Engineering; or instructor consent.

Letter Graded

Classroom Protocols

Attendance is recommended, but it is not mandatory, except for exam dates.

Cell phone use is prohibited.

Punctuality is appreciated.

Bring your laptop to class.

Program Information

Diversity Statement - At SJSU, it is important to create a safe learning environment where we can explore, learn, and grow together. We strive to build a diverse, equitable, inclusive culture that values, encourages, and supports students from all backgrounds and experiences.

Course Goals

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

1. Understand programming language design.
2. Achieve competence in a functional programming language.

Course Learning Outcomes (CLOs)

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.

Course Materials

Teach Yourself Scheme in Fixnum Days

Author: Dorai Sitaram
Availability: Online resource

https://ds26gte.github.io/tyscheme/ (https://ds26gte.github.io/tyscheme/)

Other resources TBD

Course Requirements and Assignments

Homework assignments consist of 2 programming assignments in Scheme, one in JavaScript, and one in Prolog. There is also a group project building an interpreter using ANTLR and Java.

There is a final and a midterm.

In-class labs are used as the basis for your participation grade. Any question in the lab is fair game for the exams.

See Canvas at http://sjsu.instructure.com/ for more details.

Grading Information

25% -- Homework assignments (individual)
15% -- Class project (team)
25% -- Midterm
25% -- Final
10% -- Participation (labs and drills)

Assignments are generally due by 11:59 PM Pacific Time on the specified day.

Late homework assignments will not be accepted.

Nominal grading scale:

92 and above	A
90 - 91	A-
88 - 89	B+
82 - 87	B

80 - 81	B-
78 - 79	C+
72 - 77	C
70 - 71	C-
68 - 69	D+
62 - 67	D
60 - 61	D-
59 and below	F

University Policies

Per University Policy S16-9 (PDF) (http://www.sjsu.edu/senate/docs/S16-9.pdf), relevant university policy concerning all courses, such as student responsibilities, academic integrity, accommodations, dropping and adding, consent for recording of class, etc. and available student services (e.g. learning assistance, counseling, and other resources) are listed on the Syllabus Information (https://www.sjsu.edu/curriculum/courses/syllabus-info.php) web page. Make sure to visit this page to review and be aware of these university policies and resources.

Course Schedule

The final exam is December 12, 9:45-11:45am.

Please note that the schedule is subject to change with fair notice, which will be posted through Canvas at https://sjsu.instructure.com.

Tentative planned topics by class session:

1. Course introduction
2. Scheme
3. Syntax & Semantics, and Language Design Criteria
4. Higher order functions – map and filter
5. Higher order functions – fold
6. Operational semantics

7. Closures and scoping
8. Macros
9. Programming contracts
10. Java lambdas
11. JavaScript
12. Event-based programming
13. Scoping in JavaScript, JSLint, and TypeScript
14. ECMAScript 6 and metaobject protocols
15. Review for midterm
16. Midterm exam
17. ANTLR & Syntax
18. Prolog
19. Resolution and unification, the cut operator
20. Lists in Prolog and debugging Prolog
21. Ruby
22. Ruby blocks
23. Eval in Ruby, Intro to LaTeX, Domain Specific Languages (DSLs)
24. Virtual Machines
25. Python and IDE plugins
26. Inform 7
27. TBD
28. TBD
29. Review for final