San José State University
College of Science/Department of Computer Science
CS 152, Programming Language Paradigms, Section 05, Fall 2021

Course and Contact Information

Instructor:	Thomas Austin
Office Location:	MacQuarrie Hall 216 Zoom: https://sjsu.zoom.us/j/3796767168?pwd=SzNVOE4zSTNyNHNqR1RhNlJ6cDAwUT09
Telephone:	408-924-7227
Email:	thomas.austin@sjsu.edu
Office Hours:	Mondays, 11am-noon (Zoom and in-person) Thursdays, 10-11am (Zoom only) Other times by appointment SEE http://www.cs.sjsu.edu/~austin/office-hours-updates.txt FOR LAST MINUTE CHANGES
Class Days/Time:	Monday/Wednesday 12:30-1:45
Classroom:	Science 311
Prerequisites:	CS 151 or CMPE 135 (with a grade of "C-" or better)

Course Format

Faculty Web Page and MYSJSU Messaging

Course materials such as syllabus, handouts, notes, assignment instructions, etc. can be found on my faculty web page at http://www.cs.sjsu.edu/~austin/cs152-fall19 and Canvas (http://sjsu.instructure.com/). You are responsible for regularly checking with the messaging system through Canvas to learn of any updates.

Course Description

(Copied from http://info.sjsu.edu/web-dbgen/catalog/courses/CS152.html) 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.

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 (CLO)

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

"Teach Yourself Scheme in Fixnum Days", Dorai Sitaram, http://www.ccs.neu.edu/home/dorai/t-y-scheme/t-y-scheme.html.

Additional readings TBD.

Course Requirements and Assignments (Required)

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.

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.

"Success in this course is based on the expectation that students will spend, for each unit of credit, a minimum of 45 hours over the length of the course (normally three hours per unit per week) for instruction, preparation/studying, or course related activities, including but not limited to internships, labs, and clinical practica. Other course structures will have equivalent workload expectations as described in the syllabus."

Final Examination or Evaluation

The final exam is worth 20% of the total grade for the class. It is a written exam. Paper will be provided. Bring something to write with.

Grading Information

Determination of Grades

1. 30% -- Homework assignments (individual)
2. 20% -- Class project (team)
3. 20% -- Midterm
4. 20% -- Final
5. 10% -- Participation (labs and drills)

Assignments are due by 11:59 PM Pacific Time on the specified day.
Late homework assignments will not be accepted.
Nominal grading scale:

Percentage	Grade
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

Classroom Protocol

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.

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/"

CS 152, Programming Language Paradigms, Fall 2021

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

TENTATIVE Course Schedule

Week	Date	Topics, Readings, Assignments, Deadlines
1	8/23	Course introduction
1	8/25	Scheme
2	8/30	Syntax & Semantics, and Language Design Criteria
2	9/1	Higher order functions – map and filter
3	9/6	****LABOR DAY – NO CLASS****
3	9/8	Higher order functions – fold
4	9/13	Operational semantics
4	9/15	Closures and scoping
5	9/20	Programming contracts
5	9/22	Macros
6	9/27	Scheme review
6	9/29	JavaScript
7	10/4	Event-based programming
7	10/6	Scoping in JavaScript, JSLint, and TypeScript
8	10/11	ECMAScript 6 and metaobject protocols
8	10/13	ANTLR & Syntax
9	10/18	Review for midterm
9	10/20	****MIDTERM EXAM****
10	10/25	Prolog
10	10/27	Resolution and unification, the cut operator
11	11/1	Lists in Prolog and debugging Prolog
11	11/3	Ruby
12	11/8	Ruby blocks
12	11/10	Virtual Machines
13	11/15	Dynamic code evaluation, taint analysis, and information flow analysis

Week	Date	Topics, Readings, Assignments, Deadlines
13	11/17	LaTeX and domain specific languages
14	11/22	Inform 7
14	11/24	****HAPPY THANKSGIVING!!!****
15	11/29	Python and IDE plugins
15	12/1	Advanced Java features, wrap-up
16	12/6	Review for final
Final Exam	12/14	**12:15pm-2:15pm**