San José State University
Department of Computer Science
CS 153 Concepts of Compiler Design, Section 1
Fall 2015

Course and Contact Information

Instructor:	Ronald Mak
Office Location:	MH 413
Telephone:	(408) 533-2726
Email:	ron.mak@sjsu.edu
Office Hours:	MW 3:00-4:00 PM
Class Days/Time:	MW 9:00-10:15 AM
Classroom:	DH 450
Prerequisites:	CS 47 or CMPE 102 CS 146 CS 154

Faculty Web Page and Piazza 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/~mak/CS153/index.html. You are responsible for regularly checking with the messaging system through Piazza to learn of any updates.

Course Description

Theoretical aspects of compiler design, including parsing context free languages, lexical analysis, translation specification and machine-independent code generation. Programming projects to demonstrate design topics.

Learning Outcomes

1. LO 1: Develop a scanner and a parser for a procedure-oriented programming language such as Pascal.
2. LO 2: Generate a symbol table and intermediate code (parse trees).
3. LO 3: Perform semantic analysis such as type checking.
4. LO 4: Develop an interpreter that creates a suitable runtime environment from the intermediate code and the symbol table and executes the source program.
5. LO 5: Use the JavaCC compiler-compiler.
6. LO 6: Develop a compiler that generates object code in the form of assembly language for the Java Virtual Machine (JVM).
7. LO 7: The source language can be an existing programming language or one that the programming team invents.
8. LO 8: Assemble the assembly code into executable code that will run on the JVM.
9. LO 9: Learn critical job skills that employers look for in new college hires:

a. LO 9a: Work together in a small programming team. All students must form teams of around four students each. *This is a major programming course. There will be too much work for a student to work alone.*
b. LO 9b: Understand and modify a Big Hairy Legacy Application (a Pascal interpreter written in Java).
c. LO 9c: Practice good object-oriented design and apply modern software engineering tools and practices to successfully develop a large complex application (your team's compiler project).

Course Learning Outcomes (CLO)

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

1. CLO 1: Develop working compilers and interpreters for a major subset of Pascal, a simple but powerful procedure-oriented programming language.
2. CLO 2: Generate executable code for the Java Virtual Machine (JVM)
3. CLO 3: Use the JavaCC compiler-compiler to automatically generate scanners and parsers.
4. CLO 4: Employ good object-oriented design and team-based software engineering practices.

This is a major programming class and a Computer Science Department **deep course**.

Required Texts/Readings

Textbook

Title:	**Writing Compilers and Interpreters**, *3rd edition*
Author:	Ronald Mak
Publisher:	Wiley Publishers, Inc.
ISBN-13:	978-0-470-17707-5
Source files:	http://www.apropos-logic.com/wci/
Title:	**Generating Parsers with JavaCC**, *2nd edition (PDF)*
Author:	Tom Copeland
Publisher:	Centennial Books http://generatingparserswithjavacc.com/
ISBN-10:	0-9762214-3-8

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.

You will form **project teams** of around four students each. The teams will last throughout the semester. Once the teams are formed, you will not be allowed to move from one team to another, so form your teams wisely! Each student must be on a team.

#	Date	Due	Assignment
1	Aug 24	Sept 2	Write a Pascal program http://www.cs.sjsu.edu/~mak/CS153/assignments/1/Assignment1.pdf **Input file:** widgets.csv http://www.cs.sjsu.edu/~mak/CS153/assignments/1/widgets.csv
2	Sept 9	Sept 18	Java scanner http://www.cs.sjsu.edu/~mak/CS153/assignments/2/Assignment2.pdf **Input file:** javatest.in http://www.cs.sjsu.edu/~mak/CS153/assignments/2/javatest.in
3	Sept 16	Sept 30	Pascal set expressions and assignments http://www.cs.sjsu.edu/~mak/CS153/assignments/3/Assignment3.pdf **Input files:** sets.txt seterrors.txt http://www.cs.sjsu.edu/~mak/CS153/assignments/3/sets.txt http://www.cs.sjsu.edu/~mak/CS153/assignments/3/seterrors.txt **Sample output:** sets.out seterrors.out http://www.cs.sjsu.edu/~mak/CS153/assignments/3/sets.out http://www.cs.sjsu.edu/~mak/CS153/assignments/3/seterrors.out
4	Sept 30	Oct 16	Pascal set type definitions and set variable declarations http://www.cs.sjsu.edu/~mak/CS153/assignments/4/Assignment4.pdf **Input files:** input.txt errors.txt http://www.cs.sjsu.edu/~mak/CS153/assignments/4/input.txt http://www.cs.sjsu.edu/~mak/CS153/assignments/4/errors.txt
5	Oct 21	Oct 30	Generate a Java scanner with JavaCC http://www.cs.sjsu.edu/~mak/CS153/assignments/5/Assignment5.pdf **Input files:** input.txt http://www.cs.sjsu.edu/~mak/CS153/assignments/5/input.txt
6	Oct 28	Nov 6	Generate a parser with JavaCC http://www.cs.sjsu.edu/~mak/CS153/assignments/6/Assignment6.pdf
7	Nov 9	Nov 20	Generate parse trees with JJTree http://www.cs.sjsu.edu/~mak/CS153/assignments/7/Assignment7.pdf

During the last half of the semester, each programming team will develop a compiler using the JavaCC compiler-compiler. This project will involve:

- Creating the **grammar** for the source language. This can be an existing procedure-oriented language (or subset thereof), or you can invent your own source language. *No Lisp or Lisp-like languages.*
- Generating the **scanner** and the **parser** for the source language using **JavaCC**.
- Developing the **code generator** that will emit Jasmin assembly code for the JVM. A Jasmin assembler will be provided.
- Executing compiled example programs written in your chosen source language.

Deliverables (what each team turns in) for the compiler project include:

- Your **.jjt** file and a zip file of your **src** directory.
- A **written report** (5-10 pp.) that includes:
    - A high-level description of the design of the compiler. UML diagrams of the major classes are acceptable.
    - The grammar for your source language, either in BNF or as syntax diagrams. (Use JJDoc.)
    - Code diagrams that show the Jasmin code your compiler generates for some key constructs of the source language.
- Instructions on **how to build** your compiler. Unix shell, DOS .bat, or Ant scripts are acceptable. (Only if necessary.)
- Instructions on **how to run** your compiler. Once again, shell, .bat, or Ant scripts are acceptable. (Only if necessary.)
- Sample **source programs** to compile and execute on the JVM.
- Text files of the **output** from executing your source programs.

As part of the deliverables, each student must also turn in a short (1 or 2 pp.) **individual postmortem report** that includes:

- A brief description of what you learned in the course.
- An assessment of your accomplishments for your project team on the assignments and the compiler project.
- An assessment of each of your other project team members.

You should start thinking about and planning for your project early in the semester.

NOTE that University policy F69-24 at http://www.sjsu.edu/senate/docs/F69-24.pdf states that "Students should attend all meetings of their classes, not only because they are responsible for material discussed therein, but because active participation is frequently essential to insure maximum benefit for all members of the class. Attendance per se shall not be used as a criterion for grading."

Grading Policy

Your individual class grade will be weighted as follows:

30%	Assignments*
35%	Project*
15%	Midterm exam**
20%	Final exam**

** project team scores*
*** individual scores*

Each assignment and exam will be scored (given points) but not assigned a letter grade. The mean score and standard deviation will be announced after each assignment and exam. Final individual class grades will be assigned based on the class curve. Your final class grade can be adjusted up or down depending on your level and quality of participation on your project team as determined by the project tracking tools and your team members' assessments of your performance.

Note that "All students have the right, within a reasonable time, to know their academic scores, to review their grade-dependent work, and to be provided with explanations for the determination of their course grades." See University Policy F13-1 at http://www.sjsu.edu/senate/docs/F13-1.pdf for more details.

Classroom Protocol

It is very important for each student to attend classes and to participate. Cell phones in silent mode, please.

University Policies

General Expectations, Rights and Responsibilities of the Student

As members of the academic community, students accept both the rights and responsibilities incumbent upon all members of the institution. Students are encouraged to familiarize themselves with SJSU's policies and practices pertaining to the procedures to follow if and when questions or concerns about a class arises. See University Policy S90–5 at http://www.sjsu.edu/senate/docs/S90-5.pdf. More detailed information on a variety of related topics is available in the SJSU catalog, at http://info.sjsu.edu/web-dbgen/narr/catalog/rec-12234.12506.html. In general, it is recommended that students begin by seeking clarification or discussing concerns with their instructor. If such conversation is not possible, or if it does not serve to address the issue, it is recommended that the student contact the Department Chair as a next step.

Dropping and Adding

Students are responsible for understanding the policies and procedures about add/drop, grade forgiveness, etc. Refer to the current semester's Catalog Policies section at http://info.sjsu.edu/static/catalog/policies.html. Add/drop deadlines can be found on the current academic year calendars document on the Academic Calendars webpage at http://www.sjsu.edu/provost/services/academic_calendars/. The Late Drop Policy is available at http://www.sjsu.edu/aars/policies/latedrops/policy/**.** Students should be aware of the current deadlines and penalties for dropping classes.

Information about the latest changes and news is available at the Advising Hub at http://www.sjsu.edu/advising/.

Consent for Recording of Class and Public Sharing of Instructor Material

University Policy S12-7, http://www.sjsu.edu/senate/docs/S12-7.pdf, requires students to obtain instructor's permission to record the course and the following items to be included in the syllabus:

- "Common courtesy and professional behavior dictate that you notify someone when you are recording him/her. You must obtain the instructor's permission to make audio or video recordings in this class. Such permission allows the recordings to be used for your private, study purposes only. The recordings are the intellectual property of the instructor; you have not been given any rights to reproduce or distribute the material."
    - It is suggested that the greensheet include the instructor's process for granting permission, whether in writing or orally and whether for the whole semester or on a class by class basis.
    - In classes where active participation of students or guests may be on the recording, permission of those students or guests should be obtained as well.
- "Course material developed by the instructor is the intellectual property of the instructor and cannot be shared publicly without his/her approval. You may not publicly share or upload instructor generated material for this course such as exam questions, lecture notes, or homework solutions without instructor consent."

Academic integrity

Your commitment, as a student, to learning is evidenced by your enrollment at San Jose State University. The University Academic Integrity Policy S07-2 at http://www.sjsu.edu/senate/docs/S07-2.pdf requires you to be honest in all your academic course work. Faculty members are required to report all infractions to the office of

Student Conduct and Ethical Development. The Student Conduct and Ethical Development website is available at http://www.sjsu.edu/studentconduct/.

Campus Policy in Compliance with the American Disabilities Act

If you need course adaptations or accommodations because of a disability, or if you need to make special arrangements in case the building must be evacuated, please make an appointment with me as soon as possible, or see me during office hours. Presidential Directive 97-03 at http://www.sjsu.edu/president/docs/directives/PD_1997-03.pdf requires that students with disabilities requesting accommodations must register with the Accessible Education Center (AEC) at http://www.sjsu.edu/aec to establish a record of their disability.

CS 153 Concepts of Compiler Design, Fall 2015

Course Schedule

Subject to change with fair notice. Change announcements will be made via Piazza.

Readings: WCI = **Writing Compilers and Interpreters**
JavaCC = **Generating Parsers with JavaCC**

Week	Dates	Topics and activities	Readings
1	Aug 24, 26	Overview of the course What are compilers and interpreters? A software framework for compilers and interpreters Syntax diagrams Scanning (lexical analysis) *Form programming teams*	WCI 1, 2, 3
2	Aug 31 Sep 2	Symbol table management Top-down recursive-descent parsing Parsing assignment statements and expressions	WCI 4 WCI 5
3	Sep 9	Intermediate code (parse trees)	WCI 5
4	Sep 14, 16	Interpreting assignment statements and expressions Parsing control statements Parser error handling	WCI 6, 7
5	Sep 21, 23	Interpreting control statements Runtime error handling Parsing declarations	WCI 8, 9
6	Sep 28, 30	Parsing declarations*, cont'd* Semantic actions and type checking	WCI 9, 10
7	Oct 5, 7	Scope and the symbol table stack Parsing programs, procedures, and functions Parsing procedure and function calls Runtime memory management The runtime stack and activation frames	WCI 11, 12
8	Oct 12, 14	Passing parameters by value and by reference Interpreting Pascal programs *Midcourse review* ***Midterm exam Wednesday, October 14***	WCI 12
9	Oct 19, 21	A simple DFA scanner BNF grammars for programming languages The JavaCC compiler-compiler	JavaCC 1
10	Oct 26, 28	Generating a scanner with JavaCC Generating a parser with JavaCC JJTrees	JavaCC 2, 3, 4
11	Nov 2, 4	JavaCC error handling The Java Virtual Machine (JVM) architecture Jasmin assembly language Code templates and code generation	JavaCC 7 WCI 15

12	Nov 9	Code for expressions Code for assignment statements	WCI 16
13	Nov 16, 18	Code for procedure and function calls Code to pass parameters by value and by reference Code for string operations	WCI 17
14	Nov 23, 25	Code for control statements Code for arrays and records Code for records JavaCC error handling with synchronization	WCI 18
15	Nov 30 Dec 2	Project presentations and demos	
16	Dec 7	Static and dynamic scoping Heap, stack, and garbage collection *Course review*	WCI 19
	Dec 9	***Final projects due Wednesday, December 9***	
	Dec 16	***Final exam Wednesday, December 16*** DH 450 7:15 - 9:30 **AM**	