

CpS 450: Language Translation Systems Spring Semester 2023-2024

Instructor: Stephen Schaub Email: sschaub@bju.edu

Office: Mack Building 2nd Floor Telephone: Ext. 2264
Office Hours: By appointment Credit/Load: 3/3

Course Information

Course Description

A study of compilers and interpreters, including scanning, parsing, and code generation. A compiler will be implemented with the aid of compiler generation tools. Second semester, three hours. *Prerequisite: CpS 350*.

Course Overview

This course provides an introduction to the issues involved in translating high-level computer languages to low-level ones. Compiler-generation tools are the result of one of the most successful examples of the application of theory to practice in the history of computer science, and their discussion will receive special emphasis.

As a requirement for passing this course, the student must successfully implement a compiler that meets certain minimum specifications. The compiler project is divided into several deliverables and checkpoints. A successful project will be the result of timely submission of deliverables as well as a working final product.

Students who fail to meet the minimum project specifications will not pass the class.

Textbook Information

Compiler Design: Theory, Tools, and Examples, by Seth Bergmann. Available Online.

Grading

Grading:						
Qty	Item	Points	Total			
3	Exercises	Varies	100			
4	Deliverables	Varies	500			
2	Written Tests	Varies	280			
1	Final Exam	120	120			
Total Points:		100	0			

Scale:	approximate
A	90-100%
В	80-90%
C	70-80%
D	60-70%
F	<60%

Course Policies

Homework problem sets are due at the beginning of class and may not be submitted late. Neatness counts. The instructor may require sloppy work to be redone and resubmitted for a 10% penalty.

Project deliverables can receive full credit only if submitted electronically in full at 11:59 p.m. on the day due. A 25% penalty will be applied if the assignment is not turned in on time. No credit is possible after one week. The instructor may waive one late penalty for students who are punctual in their class attendance.

I award +3% bonus to the first three students who upload their solutions electronically. In order to receive this bonus, your electronic submission must be in order, and your printed submission must be on time and include all required elements.

Deliverables may be submitted late by approval of the instructor. The instructor will allow this only for students who formally request permission to submit the program late. The request must be made by email, and should report the number of hours invested, summarize the work completed to date, include a description of problems encountered, and attach your current program listing. The request must be submitted by the original assignment deadline. Requests indicating little effort invested will be denied.

Electronic devices may not be used during class except by special arrangement with the instructor.

Gum chewing in professional settings is inappropriate and therefore not permitted in class. Compliance with student handbook policies is expected during class.

You may not use **generative AI tools** (i.e. Chat GPT, Bing Chat, Google Bard, etc.) in this course for any assignment without the professor's express permission. Should an AI tool be used with permission, its use must be documented.

Compliance with student handbook policies is expected during class.

University Policies

Attendance Policy

You are expected to attend class and be on time per the standard University attendance policy: https://home.bju.edu/bju-policies/. If you come late or leave early, I will record a partial attendance mark if you missed at most 15 minutes of class. If you miss more than 15 minutes of class, you will be marked absent. If you exceed the 3 allowed absences for this class, you may be withdrawn from class.

For planned absences, notify me a week ahead of time by e-mail. Written assignments and scheduled tests should be completed before your planned absence; please contact me to make arrangements for doing so. It is your responsibility to check in advance of a planned absence to verify what is due.

For absences due to illness or emergency, contact me by the end of the day of your absence to indicate the reason for your absence and to arrange for making up any graded work without penalty. In these situations, you will be able to make arrangements for making up tests without penalty for the first occurrence. Each subsequent time a test is missed because of incapacitating illness or emergency, an additional 10 percent grade penalty for that test will be incurred.

Accommodations for Students with Disabilities

If you have a documented learning disability or if you are impaired in some way (auditory, visual, cognitive, neurological, or physical), please let the instructor know this within the first week of the course so that any necessary adjustments can be made before you get behind.

Academic Honesty and Integrity Policy

See the Computer Science Department's Academic Integrity Policy:

https://cs.bju.edu/academics/policies/academic-integrity-policy/

Copyright 2024 Stephen Schaub as to this syllabus and all lectures. Students are prohibited from selling (or being paid for taking) notes during the course to, or by any person, or commercial firm without the express written permission of the professor teaching the course. This course uses copyrighted material from other sources. This material is being used by permission.

Schedule				
Date	Day	Class	Assignment	
Jan. 11	Th	Introduction		
Jan. 16	T	Scanning		
Jan. 18	Th	Scanning		
Jan. 23	T	Syntactic Analysis	Phase 1	
Jan. 25	Th	Recursive Descent Parsing ANTLR Parser Generation	Ch. 3 Homework	
Jan. 30	T	Grammar Analysis	Ch. 2 Homework	
Feb. 1	Th	LL Parsing	Phase 2 Checkpoint	
Feb. 6	T	ALL Parsing	Ch. 4 Homework	
Feb. 8	Th	Test 1		
Feb. 13	T	ANTLR Tree Processing	Phase 2	
Feb. 13 – 16	T-F	Bible Conference		
Feb. 20	T	Symbol Table		
Feb. 22	Th	Semantic Processing		
Feb. 27	T	Code Generation		
Feb. 29	Th	Code Gen – Variables	Phase 3	
Mar. 5	T	Code Gen – Control Structures		
Mar. 7	Th	Code Gen – Method Calls		
Mar. 12	T	Code Gen – Debugging		
Mar. 14	Th	Test 2		
Mar. 18-22	M-F	Spring Break		
Mar. 26	T	Code Gen – Runtime Storage	Phase 4	
Mar. 28	Th	Semantics – Classes		
Apr. 2	T	Strings Semantics – Inheritance		
Apr. 4	Th	Code Gen - Inheritance		
Apr. 9	T	Garbage Collection		
Apr. 11	Th	Optimization		
Apr. 16	T	Optimization		
Apr. 18	Th	TBA		
Apr. 23	T	Peer Reviews		
Apr. 25	Th	Beyond Compilers		
Apr. 29-May 2	M- Th	Final Exam Week		

Curriculum Information

Context

This course supports the following objectives of the Computer Science and Information Technologies programs:

- CS 2. Use appropriate technology as a tool to solve problems in various domains
- CS 5. Demonstrate an ability to communicate technological information effectively both in written and oral forms
- CS 6. Demonstrate an ability to acquire new knowledge in the computing discipline

Learning Objectives

Objective	Content	Assessment
Describe the steps and algorithms used by language	(Most	Test 1, 2, Final
translators (CS 8)	lectures)	
Use compiler generation tools to assist in the construction of a compiler (CS 2)	Lecture 1, 3, 8	Project
Recognize the connection of formal models such as finite state automata to language definition through regular expressions (CS 8)	Lecture 2	Test 1
Discuss the types and effectiveness of optimization (CS 8)	Optimization Lecture	Final