Course Information

- **ECE 573:** Compilers and Translator Writing Systems
- **CRN:** 16843
- **Meeting time:** MWF 1:30-2:20 (FRNY G124)
- **Instructional Modality:** Face-to-Face
- **Course credit hours:** 3
- **Prerequisites:** A solid grounding in Java or some other high level programming language. A working knowledge of data structures and assembly language. A grasp of the basic fundamentals of computer organization and architecture.

Instructor(s) Contact Information

Xiaokang Qiu (Instructor)

- **BHEE 329**
- **765-494-9987**
- **xkqiu@purdue.edu**
- **Office hours:** MW 12:25-1:25 (in-person), Th 2-3 (online); Zoom link: https://purdue-edu.zoom.us/my/xkqiu

Pranab Dash (Teaching Assistant)

- **BHEE 317**
- **dashp@purdue.edu**
- **Office hours:** TuTh 10-12 (in-person), WF 9:30-10:30 (online); Zoom link: https://purdue-edu.zoom.us/j/93399641371?pwd=NHIxWWowanI3SXRlNVVLakhUVnJaZz09

Jason Jones (Teaching Assistant)

- **TBD**
- **jone2078@purdue.edu**
- **Office hours:** M 2-3, Tu 3-5 (all hybrid); Zoom link: https://purdue-edu.zoom.us/j/7018517363

Course Description

The design and construction of compilers and other translators. Topics include compilation goals, organization of a translator, grammars and languages, symbol tables, lexical analysis, syntax analysis (parsing), error handling, intermediate and final code generation, assemblers, interpreters, and an introduction to optimization. Emphasis is on engineering a compiler or interpreter for a small programming language–typically a C or Pascal subset. Projects involve the stepwise implementation (and documentation) of such a system.

Learning Resources, Technology & Texts

Websites

- **Website (one-stop shop, reference implementation):** https://cap.ecn.purdue.edu/compilers/
- **Brightspace (lecture videos, quizzes, grades):** https://purdue.brightspace.com/d2l/home/880332
- **Gradescope (problem sets):** https://www.gradescope.com/courses/569379
- **Piazza:** https://piazza.com/purdue/fall2023/ece468573

Notes and Books

The primary course material will be in the form of lecture notes and (digital) handouts. Optionally, students might find the following textbook useful:

Crafting a Compiler, C. Fischer & R. LeBlanc, Addison Wesley, 1991, ISBN No. 978-0805321661.

Software

- **Java or Python:** While students have the option of building their course project in other languages, we will provide substantial starter code in Java and Python.
- **Antlr:** The starter software uses ANTLR, a toolkit for building compilers. ANTLR can be downloaded here: https://www.antlr.org
- **IDEs:** We recommend the use of an IDE for development. You can try:
  - Visual Studio Code: https://code.visualstudio.com
  - Eclipse: https://www.eclipse.org/downloads/packages/
- **Git and GitHub:** Programming assignments will be distributed via GitHub classroom. So you will need an installation of git and a GitHub account (see below under "assignment submission").

Learning Outcomes

At the end of the course, a student who has successfully met the course objectives will be able to:

1. Describe and explain the terminology, representation and use of formal languages and grammars [1];
2. Describe and explain the terminology and techniques of lexical analysis, parsing, semantic actions and code generation [1];
3. Design and implement a compiler for a small language based on their knowledge of the previous two points [1, 2, 5, 6].

More specifically, at the end of the course, you will be able to:

- Explain the various passes of a compiler (scanners, parsers, semantic actions and code generation, register allocation and basic optimizations) and how they relate to the overall compilation process.
- Explain and implement the algorithms for each of these processes.
- Be able to implement each of these passes and integrate them into a full compiler.
- Explain program analysis techniques that are used for code optimization, such as dataflow analysis, reaching definitions analysis, liveness analysis, etc.
- Describe basic code transformations and their application to program optimization.

Assignments and Exams

Exams

We will have evening midterms. The midterms and final are open book and open notes. The tentative times and topics for the exams are below:

Exam	Time	Location	Topics
Midterm 1	Wednesday, September 27th, 8:00-9:15pm	BHEE 129	Introduction, scanning and parsing, code generation
Midterm 2	Thursday, November 2nd, 8:00-9:15pm	ME 1130	Type checking, register allocation, instruction scheduling, peephole optimizations

Final	Friday, December 15th, 8:00-10:00am	Hiler Thtr	Loop optimizations, dataflow analysis, pointer analysis and dependence analysis

Due to the extra hours for midterms, the meetings on Monday, November 20th, Monday, November 27th and Friday, December 8th are to be omitted (tentatively).

Problem Sets

There will be 6-7 problem sets, approximately one every two weeks. Each set will be posted online on Gradescope and normally will be due by 11:59pm on Sundays. **Late submissions will not be accepted.** The problem sets will be graded on a 0-1 system: 1 point for turning in a serious attempt, 0 points for not turning in a set or not making an honest attempt at the problems. While the problem sets factor into your grade, the primary benefit is for your own study; midterm and exam questions will often be in the same format as the questions on the problem sets.

Project

This project involves implementing a full-fledged optimizing compiler for a simple language. The project consists of multiple steps, each of which will be graded separately. However, each step builds on the results of previous steps, so it behooves you to ensure that each step works properly. The bulk of your project grade is based on the correctness of your compiler on several predetermined test programs; 30% of the grade will be based on your final compiler's correctness and performance on many undisclosed test programs. The project steps (and suggested dates of completion) are as follows:

Step	Task	Suggested date of completion
Step 1	*Scanning and parsing*	*Fri, September 8*
Step 2	*Code generation for assignments and expressions*	*Fri, September 15*
Step 3	*Code generation for control statements*	*Fri, September 22*
Step 4	*Code generation for functions*	*Fri, October 20*
Step 5	*Register allocation*	*Fri, November 3*
Step 6	*Pointers and arrays*	*Fri, November 17*
Step 7	*Choose your own adventure*	*Tue, November 28*
Step 8	*Continuous improvement and optimization*	*N/A*

* Due dates subject to change.

You can (optionally) work on the project in teams of 2. You must decide if you want to work on a group prior to turning in Step 1. If you choose to work in a group, you must indicate it in your submission. Once you choose a partner, you must continue to work with this partner for the remainder of the project. You and your partner will get the same grade, i.e., we do not distinguish your individual contribution from your partner.

We reserve the right to make appropriate adjustments to ensure that grades are appropriate in rare and unusual circumstances, e.g., performance that gets dramatically better or worse over the course of the term, extreme mismatches between exam and homework averages, sickness affecting an exam, etc. In cases of doubt, we will try to be generous, within the limits of common sense.

The deadline for all steps is Friday, December 1st, 5PM. Late penalty will be 10% per day, up to 5 days. Weekends do count as late days. Due dates are subject to change at the instructor's discretion.

Grading Scale

Grades will be assigned as follows:

36% — Tests (2 midterms and 1 final, 12% each)
49% — Project
10% — Problem sets (6-7 total)
5% — Class participation
1% — Bonus for submitting course evaluation

The participation credit is determined by many factors, some of which are measurable (e.g., every endorsed answer on Piazza gets you a point, and every reported and confirmed bug of the project/assignments gets you two points) and some of which are at the instructional staff's discretion (e.g., how active you were in classroom/office hours).

There may be a constant curve (i.e., all grades will be increased by a fixed amount) for individual exams at the instructor's discretion. Your course grade will be determined using an absolute scale: 97–100: A+; 91–97: A; 88–91: A-; and continuing down.

How to Succeed in This Course

1) If you think an assignment is buggy or unclear, post a query to Piazza.
2) Start early! You are highly encouraged to start working on the steps as soon as they are posted and not wait until the last day.
3) You are encouraged to write your own test cases to make sure that your compiler fully supports the provided grammar. Doing this throughout the semester will help you ensure that your compiler will pass the hidden test cases.
4) In any case, if you feel lost, please come and speak with the instructor/TA during their office hours, and/or/xor send them email.

Attendance Policy

This course follows Purdue's academic regulations regarding attendance, which states that students are expected to be present for every meeting of the classes in which they are enrolled. However, class participation points *will not be tied to attendance*. Instead, they will be evaluated based on engagement with the course. This may be through asking questions during recitation sections, but may also include engagement during office hours or on Piazza.

When conflicts or absences can be anticipated, such as for many University-sponsored activities and religious observations, the student should inform the instructor of the situation as far in advance as possible. For unanticipated or emergency absences when advance notification to the instructor is not possible, the student should contact the instructor as soon as possible by email or phone. When the student is unable to make direct contact with the instructor and is unable to leave word with the instructor's department because of circumstances beyond the student's control, and in cases falling under excused absence regulations, the student or the student's representative should contact or go to the Office of the Dean of Students website to complete appropriate forms for instructor notification. Under academic regulations, excused absences may be granted for cases of grief/bereavement, military service, jury duty, and parenting leave. For details, see the Academic Regulations & Student Conduct section of the University Catalog website. Guidance on class attendance related to COVID-19 are outlined in the Protect Purdue Pledge for Fall 2021 on the Protect Purdue website.

Course Schedule

Week	Topic
Week 1	*What is a compiler*
Week 2	*Regular expressions, finite automata and scanners*
Week 3	*Context free grammars and parsers*
Week 4	*Symbol tables and assignments and expressions*
Week 5	*Control structures*
Week 6	*Functions*
Week 7	*Type checking*
Week 8	*Optimizations*
Week 9	*Optimizations – cont'd*
Week 10	*Register allocation*
Week 11	*Pointers and arrays*
Week 12	*Control flow graphs and dataflow analysis*
Week 13	*Pointer analysis*
Week 14	*Loop optimization*
Week 15	*Loop optimization – cont'd*
Week 16	*Slack*

* Schedule and assignments subject to change. Any changes will be posted in Brightspace and/or Piazza.

Academic Integrity

We hope (and assume) that you are all honest and have no intentions of cheating. Cheating ruins the experience for everyone and we will pursue appropriate penalties if we catch someone cheating. Please don't -- it's not worth it, both for you and for us. You can find Purdue's student guide for academic integrity at https://www.purdue.edu/odos/academic-integrity/ .

If we do catch someone cheating for the first time, the following penalties will be applied:

- On an exam: zero for the exam.
- On an assignment: zero on the assignment. Moreover, we will subtract 10% from your final average in the course, or reduce your letter grade by one category (e.g., B to B-), whichever results in the lower grade.

For second cheating offense of any kind, you will fail this course.

Nondiscrimination Statement

Purdue University is committed to maintaining a community that recognizes and values the inherent worth and dignity of every person; fosters tolerance, sensitivity, understanding, and mutual respect among its members; and encourages each individual to strive to reach his or her potential. In pursuit of its goal of academic excellence, the University seeks to develop and nurture diversity. The University believes that diversity among its many members strengthens the institution, stimulates creativity, promotes the exchange of ideas, and enriches campus life. A hyperlink to Purdue's full Nondiscrimination Policy Statement is included in our course Brightspace under University Policies and Statements.

Accessibility

Purdue University strives to make learning experiences accessible to all participants. If you anticipate or experience physical or academic barriers based on disability, you are welcome to let me know so that we can discuss options. You are also encouraged to contact the Disability Resource Center at: drc@purdue.edu or by phone at 765-494-1247.

Mental Health/Wellness Statement

If you find yourself beginning to feel some stress, anxiety and/or feeling slightly overwhelmed, try WellTrack**.** Sign in and find information and tools at your fingertips, available to you at any time.

If you need support and information about options and resources, please contact or see the Office of the Dean of Students. Call 765-494-1747. Hours of operation are M-F, 8 am- 5 pm.

If you find yourself struggling to find a healthy balance between academics, social life, stress, etc., sign up for free one-on-one virtual or in-person sessions with a Purdue Wellness Coach at RecWell. Student coaches can help you navigate through barriers and challenges toward your goals throughout the semester. Sign up is free and can be done on BoilerConnect.

If you're struggling and need mental health services: Purdue University is committed to advancing the mental health and well-being of its students. If you or someone you know is feeling overwhelmed, depressed, and/or in need of mental health support, services are available. For help, such individuals should contact Counseling and Psychological Services (CAPS) at 765-494-6995 during and after hours, on weekends and holidays, or by going to the CAPS office on the second floor of the Purdue University Student Health Center (PUSH) during business hours. The CAPS website also offers resources specific to situations such as COVID-19.

Basic Needs Security

Any student who faces challenges securing their food or housing and believes this may affect their performance in the course is urged to contact the Dean of Students for support. There is no appointment needed and Student Support Services is available to serve students 8 a.m.-5 p.m. Monday through Friday.

Emergency Preparation

In the event of a major campus emergency, course requirements, deadlines and grading percentages are subject to changes that may be necessitated by a revised semester calendar or other circumstances beyond the instructor's control. Relevant changes to this course will be posted onto the course website or can be obtained by contacting the instructors or TAs via email or phone. You are expected to read your @purdue.edu email on a frequent basis.