

CS 320: Principles of Programming Languages

Fall 2018 Syllabus

Instructor

| Name | Email | Office Hours |
|-----------------|--|-----------------------------------|
| Katie Casamento | cas28@pdx.edu | Wednesday 2-4pm or by appointment |

Teaching Assistant

| Name | Email | Office Hours |
|-------------|--|--------------|
| Phuong Pham | phuong@pdx.edu | Monday 2-4pm |

Meeting times and location

Lectures are on Mondays and Wednesdays in EB 103, 6:40-8:30pm.

Class webpage

We will be using D2L (<http://d2l.pdx.edu>; Odin account required for login) as our primary website. Announcements, class materials (including lecture slides and assignments), discussion forums, and homework submission will be hosted at that site.

Course description

Syntax and semantics. Compilers and interpreters. Programs as data. Regular expressions and context free grammars. Programming paradigms, including procedural, functional, and object-oriented programming. Type systems, including dynamic and static typing disciplines. Binding, scope, data abstraction, and modularity. Denotational, operational, and axiomatic semantics. Introduction to program correctness. (The official description is available at <http://www.pdx.edu/computer-science/cs320>.)

Prerequisites

This class requires CS 202 and 251, passed with grades of C or better. These courses provide you with the background that you will need to understand the course material and develop solutions to the assignments that you will be given. It is assumed that you have experience programming in a high-level, object-oriented language (specifically C++ or Java, from CS 202 and earlier classes). If you have not completed the prerequisites, then you may not have the background that you need to pass this class. In this case, by default, you will not be able to continue in the class and you should contact the instructor at the earliest opportunity to avoid an administrative drop.

Acknowledgements

Most of the material for this course (including much of this document) is borrowed or derived, with permission, from Prof. Mark Jones' materials for his Winter 2017 iteration.

Course objectives

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

1. Explain the distinction between language syntax and semantics.
2. Describe the similarities and differences between interpreters and compilers.
3. Explain the phase structure of a typical compiler and the role of each phase.
4. Use regular expressions and context-free grammars to describe the syntax of simple programming languages.
5. Illustrate the features and characteristics of different programming paradigms, including procedural, functional, and object-oriented programming.
6. Explain the concepts of binding, scope, block structure, and lifetime, and apply them to resolving variable uses to their binding sites in a variety of languages.
7. Describe and apply the basic concepts of type systems, including primitive types, compound, and recursive types, abstract data types, and type equivalence models.
8. Describe the strengths and limitations of static and dynamic typing disciplines.
9. Describe and apply the basic concepts of data abstraction, encapsulation, object-oriented classes, and modules.
10. Explain basic approaches and applications for the formalization of programming language semantics.

Outline of course content (SUBJECT TO CHANGE)

| Week | Lecture Dates | Lecture Topic | |
|------|---------------|---|--|
| 1 | 9/24, 9/26 | Course overview. Syntax and Semantics. Program equivalence, and the distinction between syntactic and semantic equivalences. Representation of programs as abstract syntax trees. Distinctions between concrete and abstract syntax and between static and dynamic semantics. | Functional programming. Programming without side effects. Recursion. Processing structured data. First-class functions. Binding and scope. |
| 2 | 10/1, 10/3 | Programs that manipulate programs as data. Basic principles in the function and construction of interpreters, compilers, program generators, and program analysis tools. | |
| 3 | 10/8, 10/10 | Techniques for describing language syntax. Regular expressions, context-free grammars, and common usage patterns. | |
| 4 | 10/15, 10/17 | | |
| 5 | 10/22, | Object-oriented programming. Stateful objects. Classes, fields, | |

| | | | |
|----|--|---|--|
| | 10/24 | methods, and constructors. Subclasses, inheritance, method overriding, and dynamic dispatch. Object lifetimes. | |
| 6 | 10/29, 10/31 | Midterm exam. Post-midterm review. | |
| 7 | 11/5, 11/7 | Type systems. A type as a set of values together with operations. Primitive types. Compound types. Product and sum types. Recursive types. Parameterized types. Abstract types. Modularity and encapsulation. | |
| 8 | 11/14 (no class 11/12 for Veteran's Day) | Type checking. Association of types to variables and values. Dynamic and static typing disciplines. Type safety. Decidability issues. Interpreting type error messages. | |
| 9 | 11/19, 11/21 | Techniques for formalization of programming language semantics, including denotational, operational, and axiomatic approaches. | |
| 10 | 11/26, 11/28 | Introduction to program correctness. Program specification using predicate logic. Application of formal semantics, and the role of tools for mechanized reasoning. | |

Exams

The midterm exam will be held on Monday of week 6 (10/29); the final exam will be Monday 12/3 at 7:30-9:20pm in the regular classroom, in accordance with [university guidelines](#).

Required reading

The slides and other materials that are presented in lectures and on the course web site are required reading.

There is no *required* textbook for this class, but all students are strongly encouraged to do background reading in parallel with the coverage of topics in class. There are many textbooks on the subjects of this course, including a good range that are available in the PSU library. However, we do not expect you to buy or have access to any particular text: for example, we will not set exercises that come from a text book, or require you to have read specific sections of any particular book.

Of course, there are plenty of good resources on the Internet of direct relevance to this course that can be found on sites like Wikipedia or simply by searching on Google.

Course requirements and method of evaluation

The course will be assessed by a combination of exams and homework.

There will be one midterm exam (in Week 6) and one final exam (in finals week), both of which will be *closed-book*. Exams are scheduled in advance and, unless a prior arrangement is made, a grade of zero will be recorded for missed exams.

The midterm will contribute 25% of your overall grade for the class, while the final will contribute 35%. The remaining 40% will be determined by homework assignments. The number and frequency of homework assignments will be determined by balancing the need for timely feedback and grading with consideration of the class size and the available resources. Final course grades may be curved at the instructor's discretion to account for potential class-wide low numerical scores.

A general expectation is that answers should include appropriate explanation, justification, or commentary to support your answers and show that you understand why your answer is correct. The grading schemes for individual assignments will typically include points specifically for these items, so writing only the final answer to a question, even if it is correct, may not earn full marks.

For questions that require programming, you will often be asked to include evidence of "thoughtful testing". The intent here is that you aim for quality of tests rather than quantity, focusing on which aspects of your solution are exercised by your tests, the handling of edge cases, and the steps that you took to determine what the correct behavior or result should be. If you just show the output from running a program without including further commentary to justify or explain the results, then all you are really doing is showing that your program runs, and not that it is correct.

There may be some assignments that are not formally assessed. This might include written exercises, additional reading, watching prepared videos, etc. These will be designed to help you master the subject, to keep pace with the lectures, and to prepare for the exams, so it will be very important for you to keep up to date with those exercises.

Getting assistance

A fundamental assumption and expectation of this course is that students will ask for help or clarification if they have any problems understanding the course materials, or the work that is required for an assignment. As general guidelines:

- Questions may be asked in person (during lectures, or office hours, for example) or online using the D2L forums (preferred) or email (for personal matters, or lower-priority questions).
- In general, we will prioritize conversations on D2L forums over individual email because D2L postings have the potential to benefit the largest possible group of people. If you send a message by email, we will still try to respond as quickly as possible, but may need to respond to D2L postings first or to other individual email messages that arrived before yours.
- Anonymous posting on D2L is permitted for those who are uncomfortable about posting questions under their own name.
- To avoid a breach of academic integrity (see details below), do not include specifics of your solution in postings on D2L; instead, please try phrase your questions, comments, or answers in general terms.
- If you do send a question by email, please use a regular email system. The "email" facility that is built in to D2L generates invalid sender addresses, that are not recognized by the PSU mail system.
- You should ensure that you are subscribed to receive notifications when news items or new forum messages are posted on D2L. If you do not subscribe for these updates, then you may miss important announcements, clarifications, or updates.
- If you are not comfortable or do not have time to ask for help, then you may find that the course is (perhaps considerably) more difficult than intended. In this situation, you may want to consider whether it is appropriate for you to be taking the course at this time: please contact the instructor if you have any concerns about this.

Computing facilities

The assignments for this course will require the use of a variety of programming language implementations. Students may develop their solutions on any machine and operating system on which these implementations are available. At a minimum, all of the systems that we use will be installed on the departmental Linux systems that can be accessed in person in the lab area, or remotely via ssh/putty.

Academic integrity

We follow the standard guidelines for academic integrity. This is important because a breach of academic integrity by one student undermines the efforts and achievements of the other students in the class who have made honest and legitimate attempts to study for and complete assignments and exams. It is permissible to *discuss* assignments with other students, but you must develop the solution yourself. *Do not, under any circumstances, copy any part of another person's solution and submit it as your own.* Unless you are given explicit written instructions to the contrary, sharing of code or test cases is not permitted. In particular, posting or soliciting all or part of your solution on D2L or any other public forum, media, or site may be considered a breach of academic integrity. Writing code for use by another person, or using another person's code in any form (even with their permission) will be considered cheating. Cheating on an assignment or exam will result in an automatic zero grade for that piece of work, and the initiation of disciplinary action at the University level. Please refer to <http://www.pdx.edu/dos/codeofconduct> for details of the general PSU Student Code of Conduct. Any student with questions about academic integrity issues, either relating to their own behavior, or with concerns about the behavior of other students, should contact the instructor. All such matters will be treated in confidence.

Always remember that the instructor and the TA are here to help you to succeed in this class; if you find yourself in a difficult position, do not be tempted to cheat, and instead get in touch to ask for some help or guidance.

Disabilities and accommodations

We will do everything possible to provide accommodations, with full confidentiality, to any student who needs them. All accommodations must be approved by the Disability Resource Center, which establishes fair and consistent standards across campus. Students with a documented disability who are registered with the Disability Resource Center are responsible for ensuring that their specific requirements are clearly communicated to the instructor at the earliest possible opportunity, either directly or via the DRC. Students should take steps to notify the instructor as soon as possible if they feel that their needs are not being met. If you have accommodations that include taking tests at the university test center, then you are required to schedule those tests so that they coincide with the time that the other students are taking the test in the main classroom. It is important that you take steps to make the necessary reservations at the earliest possible opportunities; the times and dates for both the midterm and final exams are already fixed for 2/12 and 3/19, respectively, as described above.

Subjective matters

To ensure consistency, all questions about matters that are potentially subjective - such as possible grading errors or determining whether an exceptional circumstance warrants an extension of a deadline - should be addressed to the primary instructor for this class.

Other policies and guidelines

In addition to the specific items described above, we adopt the following policies and guidance:

1. By default, all deadlines are firm. At the discretion of the instructor, penalties may be assessed for late submissions.
2. No submissions can be accepted once a sample solution has been distributed and/or discussed in class or online.
3. In general, writing only the correct answer may not earn you full marks for an assignment, and you should also be prepared to include appropriate explanation, justification, commentary etc. The intent here is that you should be able, not only to say what the correct answer is but also to explain why it is the correct answer.
4. We are unable to provide detailed feedback on draft versions of assignment solutions prior to submission; feedback of that kind might be considered as providing an unfair advantage over other students who are not able to get their work "reviewed" prior to submission. We are happy, however, to address any questions that might help to clarify the expectations and requirements for an assignment.
5. Students are solely responsible for ensuring that their attempts to upload homework solutions to D2L are successful. A "miscellaneous uploads" dropbox is provided for students to submit material after the deadline for a given assignment has passed, but there is no guarantee that items submitted there will be accepted or considered for grading. Any student who uses the miscellaneous uploads dropbox should also notify the instructor directly.
6. We try hard to ensure that grading is fair and accurate. But we also acknowledge that, as with any human process, there will always be a possibility for mistakes. In addition, even if the grading is correct, a student may not always understand why they have been assigned a specific score. For these reasons, students are welcome to seek clarification/explanation for grading decisions at any time during the term before final grades are submitted. We will never reduce the grade awarded as a result of reviewing work in this way, but there is also no guarantee that any grades will be increased as a result of review. Reviews of grading are appropriate whenever they might help to improve understanding or learning, but they should not be used as a matter of routine, and should not be used primarily as a way to try to boost scores. Reviews of this kind are typically only useful after sample solutions and grading guides have been distributed by the instructor and reviewed by the student.
7. We will try to be as flexible as possible in accommodating special circumstances. If you anticipate a specific issue, please talk to us to make arrangements in advance.
8. As specified in Oregon law (ORS 165.40 6(b)), recording of oral communications that are part of regularly scheduled classes is permitted if the recording device is not concealed. To respect the privacy of students and to avoid actions that might discourage or inhibit participation in class, please treat any such recordings as "for personal use only".