

CSE 3341 / 5341
Principles of Programming Languages
Autumn 2012
www.cse.ohio-state.edu/~rountev/3341

Syllabus

Instructor: Atanas (Nasko) Rountev

Course Summary

This course discusses various programming language concepts, as well as design and implementation topics for several language families. Topics include formal languages and grammars; recursive descent parsing; data types, expressions, control structures, and parameter passing; compilers and interpreters; memory management; functional programming principles.

Prerequisites and Co-requisites

- CSE 560: Elements of Computer Systems Programming
- CSE 625: Introduction to Automata and Formal Languages
- CSE 680: Introduction to Analysis of Algorithms

Exclusions

Not open to students with credit for CSE 655

General Information

- Credits: 3 semester credits for 3341, 2 semester credits for 5341
- Place and time: Monday, Wednesday, Friday 1:50 pm – 2:45 pm, Bolz Hall (BO) 412
- Instructor: Atanas (Nasko) Rountev, rountev@cse.ohio-state.edu, 292-7203
- Instructor's office hours: DL 685, Wednesday 12:30 pm – 1:30 pm, Friday 10:30 am – 11:30 am, or by appointment
- Grader: ?, ??@cse.ohio-state.edu, ?
- Grader's office hours: ?, ?, or by appointment.
- Course web page: www.cse.ohio-state.edu/~rountev/3341

Topics

- Overview of programming languages; language design and implementation
- Grammars (regular expressions, context-free grammars); parse trees; recursive descent parsing, interpretation, and compilation
- Expressions; control structures; subroutines
- Storage management; scoping rules; bindings for names
- Principles of object-oriented languages (data abstraction, encapsulation, inheritance, polymorphism); implementation mechanisms
- Functional programming principles; Scheme
- Data types
- Other topics

Objectives

- Master using syntax-directed parsing, printing, execution, and compilation for simple imperative language constructs
- Master distinguishing between compile-time vs. run-time activities
- Be competent with using syntax-related concepts including regular expressions and context-free grammars to describe the structure of languages
- Be competent with analyzing programming language design issues related to data types, expressions, control structures, parameter passing
- Be competent with principles of object-oriented languages
- Be competent with implementing object-oriented languages
- Be familiar with memory management techniques for imperative languages, including object-oriented languages
- Be familiar with using functional programming languages
- Be exposed to analyzing variable bindings and scope rules

Reading

- Michael L. Scott, *Programming Language Pragmatics*, 3rd edition
- Not all the material we cover will be from the book, nor will the order in which we cover this material be the same as in the book. You will have to refer to copies of the slides (which I will distribute in class and on the course web page), your own notes, as well as the book in order to keep up with the course.

Course Web Page

www.cse.ohio-state.edu/~rountev/3341: the course web page will contain all notes, handouts, assignments, schedule, pointers to reading materials, etc. Copies of assignments etc. handed out in class become official, regardless of whether they are on the web page or whether you are able to access the page. Occasionally I will post on the page some up-to-date announcements, so you should check it regularly.

Assignments

- There will be several assignments, typically due in 7 to 10 days.
- Assignments should be done independently. General high-level discussion of assignments with others in the class is allowed, but **all of the actual work** should be your own. Assignments that show excessive similarities will be taken as evidence of cheating and dealt with accordingly.
- Assignments should be turned in **by the end of class** on the due day. Late assignments turned in by the end of the next class will be graded with 30% reduction. Assignments turned in later than that will not be accepted.
- Make the assignments readable and understandable. They should be handed in on regular paper, legibly written or typed. If you have more than one sheet, **staple the sheets together**. If the grader has trouble reading or understanding what you have done, points will be deducted even if it can finally be determined that you have the correct answer.
- Your solutions have to be **precise and detailed**: you have to work out **all** details that are necessary to solve the problem using the approaches discussed in class. You also have to write your solutions in a way that convinces the grader that you understand all these details. Be careful, precise, and thorough.

Programming Projects

- There will be two programming projects, which have to be submitted electronically on *stdlinux* by midnight on the due date. The projects **must compile and run** on *stdlinux*. Some people prefer to implement the projects on a different machine, and then port them to *stdlinux*. This usually creates ugly problems in the last minute, and I would advise against it. Still, if you decide to use a different machine, it is **entirely** your responsibility to make the code compile and run correctly on *stdlinux* before the deadline.
- The project should be done independently from other students in the class. General discussion of the project with others in the class is allowed, but you have to do all the design, programming, testing, and debugging independently. Projects that show excessive similarities will be taken as evidence of cheating and dealt with accordingly.
- The projects are due by 11:59 pm on the due day. **Absolutely no exceptions** will be made to this deadline: if you submit at 12:00 am, your submission will be considered to be late. The time stamp on the electronic submission will be used to determine the submission time. A reduction of **10% per day** will be applied to late submissions. Submissions more than three days late will **not** be accepted.

Exams

- There will be a midterm exam and a final exam:
 1. Midterm: TBD, 1:50 pm – 2:25 pm, BO 412 (will be announced at least 10 days in advance)
 2. Final: Monday, December 10, 4:00 pm – 5:45 pm, BO 412

Both will be comprehensive, in-class, closed book. You will be allowed to use a cheat sheet — one standard-sized piece of paper, with notes on both sides.

- The exam questions will typically require creative application of the general approaches discussed in class. Memorizing things will not be enough; you need to have conceptual understanding of the techniques we have covered, and how these techniques could be applied to small problems. Exam questions will be very similar to the questions from the homeworks; thus, you should make sure that you have very solid understanding of all details in the homework solutions.

- Missing the midterm or the final without prior written (e-mail) approval from me will result in a score of zero for that exam. To get my approval to reschedule an exam, e-mail me **at least one week** before the exam is scheduled. I will not give such approval unless the reasons are justifiable.

Grading

Homework assignments	20%
Programming projects	30%
Midterm	15%
Final	35%

Grading Policy

The entire course will be graded on a curve. I expect the average grade to be around B. For homeworks, exams, and projects, I will provide statistics that will help you understand your standing in the class.

Whoever graded something will be responsible for handling grading disputes. I will grade the midterm exam and the final exam. The grader will grade the homeworks and the projects. Grades become final one week after an assignment or an exam is handed back. This should leave plenty of time to resolve grading disputes.

Honesty

I will treat you as professionals, and you should conduct yourselves as such. You are free to discuss the assignments and the project with others. However, the solutions you submit should be developed by yourself. **Cheating is a very serious offense and will not be tolerated.** Supplying others with materials is also against this rule. Additional details on academic integrity are available at <http://oaa.osu.edu/coamresources.html>. **Please read this information carefully.**

Students with Disabilities

Any student who feels he or she may need an accommodation based on the impact of a disability should contact me privately to discuss his or her specific needs. Please contact the Office of Disability Services at (614) 292-3307, or visit 150 Pomerene Hall, to coordinate reasonable accommodations for students with documented disabilities.

Religious Obligations

I will do my best to accommodate any religious obligations you may have. Please contact me privately, at least a week in advance, to work out any relevant details.