CSE 307 – Principles of Programming Languages
Course Information

Fall 2017
Stony Brook University
Instructor: Dr. Paul Fodor

http://www.cs.stonybrook.edu/~cse307
Course Description

• “Presents examples of important programming languages and paradigms such as LISP, ALGOL, ADA, ML, Prolog, and C++. Students write sample programs in some of the languages studied. The languages are used to illustrate programming language constructs such as binding, binding times, data types and implementation, operations (assignment data-type creation, pattern matching), data control, storage management, parameter passing, and operating environment. The suitability of these various languages for particular programming tasks is also covered.”

• Prerequisites: CSE 219 or CSE 260, and CSE 220 and the CSE major or permission of instructor.
Official Course Outcomes

• The following are the official course goals agreed upon by the faculty for this course:
  • Knowledge of, and ability to use, language features used in current programming languages.
  • An ability to program in different language paradigms and evaluate their relative benefits.
  • An understanding of the key concepts in the implementation of common features of programming languages.
Topics

• **Major Topics Covered in Course:**
  • Principles of Language Design
  • Specification of Language Syntax
  • Survey of Procedural and OO Languages
  • Intro. to Functional Programming
  • Intro. to Logic Programming
  • Programming Language Semantics
  • Values; Bindings; Types;
  • Programming Language Constructs
  • Expressions; Statements
  • Procedures and Environments
  • Parameter Passing
Staff/Instructor Information

- Dr. Paul Fodor
  214 New Computer Science Building
- Office hours: Mondays and Wednesdays 5:30PM-7:00PM
- Email: paul (dot) fodor (at) stonybrook (dot) edu
  - Please include “CSE 307” in the email subject and your name in your email correspondence
General Information

- **Meeting Information:**
  - Lectures: TuTh 7:00PM - 8:20PM, Javits102.

- **Course Web page:**

- Blackboard will be used for assignments, grades and course material
Textbook


---

F: The full-year/self-study plan

R: The one-semester Rochester plan

P: The traditional Programming Languages plan;
   would also de-emphasize implementation material
   throughout the chapters shown

C: The compiler plan; would also de-emphasize design material
   throughout the chapters shown

- Supplemental (CD) section
- To be skimmed by students in need of review
Grading Schema

- Homework assignments = 20%
- Quizzes = 5%
- Midterm exam 1 = 25%
- Midterm exam 2 = 25%
- Final exam = 25%

- Do not miss the exams. Make-up exams will be given only in extenuating circumstances (e.g., doctor's note stating that you were ill and unfit to take the exam). Students who miss an exam for a valid reason may need to take a make-up exam; specific arrangements will be made on a case-by-case basis.
Exam dates

- Midterm exam 1: Tuesday 10/10/2017, classtime, in classroom.
- Midterm exam 2: Tuesday 11/14/2017, classtime, in classroom.
- Final exam: Tue., 12/19/2017, 8:30PM-10:30PM, in classroom. See Final Exams University Schedule here: [http://www.stonybrook.edu/commcms/registrar/registration/exams.html](http://www.stonybrook.edu/commcms/registrar/registration/exams.html)
Grading Schema

- The Pass/No Credit (P/NC) option is not available for this course.
- This policy applies to all CSE/ISE undergraduate courses used to satisfy the graduation requirements for the major.
Grading Schema

• Grade Cutoffs
  • A [95-100], A- [90-95), B+ [87-90), B [83-87), B- [80-83), C+ [77-80), C [73-77), C- [70-73), D+ [65-70), D [60-65), F [0-60)
  • SPECIAL RULE: If all your grades, including homework assignments, quizzes, recitation and your three exam grades are above the respective class averages, you're guaranteed to receive a grade of C or higher for this class.

• There will be extra credit problems as a part of quizzes and homework assignments which values to an increase of less than 4% in the final grade.

• There will be in-class quizzes / brief assessments used to practice the class material and measure growth in knowledge, abilities, and skills. They will be solved in class and they are valued 2 points each.
Grading

- The final grade you receive in this class will reflect, as far as possible, the extent to which you have mastered the concepts and their applications.
- How much someone needs a grade, or how close they are to the next higher grade, will have no effect on grade.
- As the instructor, I want everyone to do well in this course, and will make every reasonable effort to help you understand the material taught.
- However, the grades provided at the end of the semester are final, except for rare situations involving grading errors.
- They will not be altered for any reason, so please do not ask me to do so.
Assignments

- There will be regular programming assignments which must be submitted electronically on **Blackboard** ([http://blackboard.stonybrook.edu](http://blackboard.stonybrook.edu)) by the announced due date and time.
  - no late submission is permitted
- All assignments should be submitted electronically
  - Blackboard
- All code must compile. Code that does not compile will not be graded.
  - Assignments will be graded based on program performance and documentation.
  - Submissions that are no submitted as requested in the assignment will not receive any credit (e.g., a Test.java file cannot be test.java, test.txt, johnSmith.java or anything else but Test.java; same for method arity)
Regrading of Homework/Exams

- Please meet with a TA or the instructor and arrange for regrading.
- You have one week from the day grades are posted or mailed or announced.
- Late requests will not be entertained.
# Class Schedule

<table>
<thead>
<tr>
<th>Week</th>
<th>Lecture Topics</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction to Programming Languages</td>
</tr>
<tr>
<td>2</td>
<td>Python</td>
</tr>
<tr>
<td>3</td>
<td>SML</td>
</tr>
<tr>
<td>4</td>
<td>Programming language syntax</td>
</tr>
<tr>
<td>5</td>
<td>Programming language syntax</td>
</tr>
<tr>
<td>6</td>
<td>Names, Scopes, and Bindings</td>
</tr>
<tr>
<td>7</td>
<td>Names, Scopes, and Bindings</td>
</tr>
<tr>
<td>8</td>
<td>Semantic Analysis</td>
</tr>
<tr>
<td>9</td>
<td>Semantic Analysis</td>
</tr>
<tr>
<td>10</td>
<td>Control Flow, Data Types</td>
</tr>
<tr>
<td>11</td>
<td>Control Flow, Data Types</td>
</tr>
<tr>
<td>12</td>
<td>Subroutines and Control Abstraction</td>
</tr>
<tr>
<td>13</td>
<td>Data Abstraction and Object Orientation, Functional Languages</td>
</tr>
<tr>
<td>14</td>
<td>Logic Languages</td>
</tr>
<tr>
<td>15</td>
<td>Logic Languages, Concurrency</td>
</tr>
</tbody>
</table>
Disability Support Services

- If you have a physical, psychological, medical or learning disability, contact the DSS office at Room 128 ECC. Phone 632-6748/TDD
- If you are planning to take an exam at DSS office, you need to tell me ahead of time for every exam.
- All documentation of disability is confidential.
Academic Integrity

- The following rules are posted in every course syllabus: "Each student must pursue his or her academic goals honestly and be personally accountable for all submitted work. Representing another person's work as your own is always wrong. **Any suspected instance of academic dishonesty will be reported to the Academic Judiciary.** For more comprehensive information on academic integrity, including categories of academic dishonesty, please refer to the academic judiciary website at [http://www.stonybrook.edu/commcms/academic_integrity/](http://www.stonybrook.edu/commcms/academic_integrity/)."
Academic Integrity

• You can discuss general assignment concepts with other students: explaining how to use systems or tools and helping others with high-level design issues

• You **MAY NOT share** assignments, source code or other answers by copying, retyping, looking at, or supplying a file

  • Assignments are subject to manual and automated similarity checking (We do check! and our tools for doing this are much better than cheaters think)

• If you cheat, you will be brought up on academic dishonesty charges - we follow the university policy:
  • [http://www.stonybrook.edu/uaa/academicjudiciary](http://www.stonybrook.edu/uaa/academicjudiciary)
Examples of Academic Dishonesty

- **Unpermitted collaboration** (on a paper, homework, lab reports, etc.). Unless an instructor has explicitly approved working together, students should assume, for their own protection, that it is not permitted.

- **Helping someone else to plagiarize from one's own homework** (for example, by giving them a copy of yours, or doing it for them)
  - This includes having a public repository on Github that other students can copy from.

- **Representing someone else's source code as one's own.** If another person's code is being used, it must be properly cited.

- **Buying or selling source code.**

- **Using source code or pieces of a paper from the internet without properly citing the source.**
Academic Integrity

- The instructor makes a recommendation at the Academic Judiciary office.
- Cheating is cheating! No matter the amount of cheating or if one is the source or destination of cheating.
- Do not cheat! You are cheating yourself.
- Our job is the teach you the material and make sure that you learn it.
- Our recommendation is always F for the cheaters!
Catastrophic events

- Major illness, death in family
- Formulate a plan (with your CEAS academic advisor) to get back on track
- Advice
  - Once you start running late, it’s really hard to catch up
Please

- Please be on time
- Please show respect for your classmates
- Please turn off (or use vibrate for) your cellphones

... 

- On-topic questions are welcome
<table>
<thead>
<tr>
<th>Websites</th>
<th>Popularity (unique visitors)</th>
<th>Front-end (Client-side)</th>
<th>Back-end (Server-side)</th>
<th>Database</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yahoo</td>
<td>750,000,000</td>
<td>JavaScript</td>
<td>JavaScript,[15] PHP</td>
<td>MySQL, PostgreSQL[16]</td>
<td>Yahoo is presently transitioning to a new site</td>
</tr>
<tr>
<td>Amazon.com</td>
<td>500,000,000</td>
<td>JavaScript</td>
<td>Java, C++, Perl[17]</td>
<td>Oracle Database[16]</td>
<td>Popular internet shopping site</td>
</tr>
<tr>
<td>Wikipedia.org</td>
<td>475,000,000</td>
<td>JavaScript</td>
<td>PHP, Hack</td>
<td>MySQL, MariaDB[19]</td>
<td>&quot;MediaWiki&quot; is programmed in PHP, runs on wiki</td>
</tr>
<tr>
<td>Twitter.com</td>
<td>290,000,000</td>
<td>JavaScript</td>
<td>C++, Java, Scala, Ruby on Rails[20]</td>
<td>MySQL[21]</td>
<td>140 characters social network</td>
</tr>
<tr>
<td>Bing</td>
<td>285,000,000</td>
<td>JavaScript</td>
<td>ASP.NET</td>
<td>Microsoft SQL Server</td>
<td></td>
</tr>
<tr>
<td>eBay.com</td>
<td>285,000,000</td>
<td>JavaScript</td>
<td>Java,[22] JavaScript[23]</td>
<td>Oracle Database</td>
<td>Online auction house</td>
</tr>
<tr>
<td>MSN.com</td>
<td>280,000,000</td>
<td>JavaScript</td>
<td>ASP.NET</td>
<td>Microsoft SQL Server</td>
<td>An email client, for simple use. Mostly known</td>
</tr>
<tr>
<td>Microsoft</td>
<td>270,000,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LinkedIn.com</td>
<td>260,000,000</td>
<td>JavaScript</td>
<td>Java, JavaScript,[24] Scala</td>
<td>Voldemort[26]</td>
<td>World's largest professional network</td>
</tr>
<tr>
<td>Pinterest</td>
<td>250,000,000</td>
<td>JavaScript</td>
<td>Django,[26] (a Python framework)</td>
<td>MySQL, Redis[27]</td>
<td></td>
</tr>
<tr>
<td>Ask.com</td>
<td>245,000,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wordpress.com</td>
<td>240,000,000</td>
<td>JavaScript</td>
<td>PHP</td>
<td>MySQL</td>
<td></td>
</tr>
</tbody>
</table>


(c) Paul Fodor (CS Stony Brook) and Elsevier
https://en.wikipedia.org/wiki/Measuring_programming_language_popularity
Welcome and Enjoy!