Course Information

<table>
<thead>
<tr>
<th>Course Staff</th>
<th>Room</th>
<th>Phone</th>
<th>Email @psu.edu</th>
<th>Office Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prof. Adam Smith</td>
<td>IST 338K</td>
<td>863-0076</td>
<td>asmith+465</td>
<td>MWF, 11:05am-noon.</td>
</tr>
<tr>
<td>TA Megan Heysham</td>
<td>IST 357</td>
<td>meh339+465</td>
<td>t.b.d.</td>
<td></td>
</tr>
<tr>
<td>TA Yang Xiao</td>
<td>IST 357</td>
<td>yux106+465</td>
<td>t.b.d.</td>
<td></td>
</tr>
</tbody>
</table>

Lectures, All Sections MWF 10:10–11:00am, Room 114, EES Building

Recitations: Mondays
Section 1: 12:20 PM - 1:10 PM in Willard 105
Section 2: 1:25 PM - 2:15 PM in Willard 269
Section 3: 4:40 PM - 5:30 PM in Willard 268

Webpage [http://www.cse.psu.edu/~asmith/cmpsc465](http://www.cse.psu.edu/~asmith/cmpsc465)

Course email is sent through Angel ([http://cms.psu.edu](http://cms.psu.edu)). Check it often or forward your mail.

Prerequisites CMPSC 122 (basic data structures, recursion), CMPSC 360/CSE 260/MATH 311W (mathematical induction, graphs, trees, equivalence relations, basic probability) or equivalent.

Textbook: Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. Introduction to Algorithms. 3rd ed. Cambridge, MA: MIT Press. ISBN: 0262032937. (Using the second edition is ok, but you may find that chapter and page numbering has changed, so make sure to check with a friend who has the third edition.)

Syllabus: Classical algorithms and data structures; techniques for the design and analysis of efficient algorithms. Topics include sorting; search-related data structures such as trees, heaps, and hash tables; graph algorithms; divide-and-conquer algorithms and recurrences; dynamic programming; greedy algorithms; amortized analysis; NP-completeness and approximation algorithms.

Homework There will be an assignment due every Friday before the lecture. The assignments will be posted one week in advance, and will depend only on the material covered up to the distribution date. Several assignments will contain programming projects that you can implement in Python.

Late homework will generally not be accepted. If there are extenuating circumstances, you should make arrangements at least 48 hours in advance with a TA. Only serious excuses will be considered in cases where prior arrangements were not made.

You should be as clear and concise as possible in your write-up of solutions. Understandability of your answer is as desirable as correctness, because communication of technical material is an important skill. A simple, direct analysis is worth more points than a convoluted one, both
because it is simpler and less prone to error and because it is easier to read and understand. Points might be subtracted for illegible handwriting and for solutions that are too long.

Electronic submissions (via Angel drop boxes) are acceptable. Emailed assignments are not.

**Collaboration and Honesty Policy** Collaboration on homework problems, with the exception of programming assignments, is permitted. If you choose to collaborate on some problems, you are allowed to discuss each problem with at most 3 other students currently enrolled in the class. Before working with others on a problem, you should think about it yourself for at least 45 minutes.

*You must write up each problem solution by yourself without assistance, even if you collaborate with others to solve the problem.* You must also identify your collaborators. If you did not work with anyone, you should write "Collaborators: none." It is a violation of this policy to submit a problem solution that you cannot orally explain to an instructor or TA.

*Finding answers to problems on the Web or from other outside sources (these include anyone not enrolled in the class) is strictly forbidden.*

*No collaboration whatsoever is permitted on exams.* The collaboration policy for programming problems will be specified in the assignments.

Violations of this policy will be dealt with according to University regulations.

**“I’ll take 20%” option on homework** Partial credit is only given for answers that make significant progress towards correct solution. Understanding whether a solution is correct is an important skill. If you realize that you cannot solve a homework problem, you have an option of writing “I’ll take 20%” instead of your answer. In this case, you will get 20% for this problem (or part of the problem). If you do write an answer, that answer will be graded and your score will be 0 if your solution is completely wrong. *You cannot use the 20% option on programming problems or on exams.*

**Optional problems** Some homework assignments will include optional problems, marked by *. Later, if you ask me for a recommendation or express an interest in working on a research project with me, I will definitely check how well you did on the optional problems. “I’ll take 20%” is not available for optional problems.

**Exams and Grading** The grade will be calculated as follows:

- **40%** Homework
- **20%** Midterm 1
- **20%** Midterm 2
- **20%** Final exam

**Honors credit** is available for this course. Please discuss with instructor.