This syllabus has the final word on course policies. This version is current as of September 25th. Any substantive changes will be announced in class or by e-mail.
See also: `http://www.cs.duke.edu/courses/fall12/compsci201`

What CPS201 is All About

According to the registrar, this is

> Analysis, use, and design of data structures and algorithms using an object-oriented language like Java to solve computational problems. Emphasis on abstraction including interfaces and abstract data types for lists, trees, sets, maps, and graphs. Implementation and evaluation of programming techniques including recursion. Intuitive and rigorous analysis of algorithms.

All of which is true! We'll be approaching it from three different directions at once:

1. *Data Structures*: the various ways that computers store information in order to process it efficiently. This is partly a survey of the classic data structures, but only in service of the classic *techniques* in data structure design. Knowing how to use existing structures is useful; knowing how to design and analyze your own is better.

2. *Algorithmic Analysis*: the mathematical and conceptual tools to measure how algorithms perform. This includes some discussion of correctness ("Does this algorithm do the right thing?"), but is mostly focused on performance ("How efficient is this algorithm?"). Getting a good definition of "efficient" is surprisingly subtle! This gives us a way to answer a fundamental question: given two different algorithms for the same problem, which one should we use? As data structures are themselves algorithms, picking the right data structure for a particular problem is a classic example.

3. *Programming*: The mathematical tools from (1) and (2) don't exist in a vacuum: they help us produce better, more useful, and more interesting computer programs. We'll be discussing techniques in effective programming, including object-oriented design and useful details of the Java language.

All three of these things come together to help us write good software that does interesting and useful things.

Grading

There are four components to grading in this course:

1. *Homeworks*: large, multi-day assignments, with meaningful written and programming components.
2. *In-class work*: Small thinking exercises during class; usually submitted via the web.
3. *APTs*: a Duke CS classic. Short programming problems, graded by a computer.
4. *Tests*: of course. Two midterms and a final.

These contribute to your final grade as follows:

Homeworks	40%
In-class work	5%
APTs	10%
Midterm #1	15%
Midterm #2	15%
Final	15%
Total	100%

Grades are assigned on an absolute scale; there is no curve. Anyone earning 90% or more of points available will receive a grade in the A range (A- to A+), and so on by steps of 10%. We may adjust the entire scale downward (requiring fewer total points for a high grade), but the scale will not be adjusted upward.

A note on APTs: APTs are "interchangeable", as follows: each APT is worth the same number of points. Some APT sets may include optional problems; if these are submitted, they can take the place of later (or earlier) APTs. Put another way: if each APT is worth k points, and we require N APTs, there are a total of kN points available. *Which* APTs you do to get those points doesn't matter. Note that there's no extra credit as such: just some slack in the weekly APT load. Deadlines still apply: you have a fixed window in which to turn in each APT.

Exam Dates

- Midterm #1 will take place on October 8th, at the usual class time and place.

- Midterm #2 will take place on November 14th, at the usual class time and place.
- The final is December 15th at 2PM, in the usual classroom.

Late Work

If something is described as "Due on day X", that means sometime on day X, in Duke's time zone. This means that the timestamp on your submission must be on the day in question.

Different kinds of work have different late policies.

1. *Homeworks*: we're trying something new for homeworks. Everybody in the class starts with three *CPS201 dollars*. Each dollar is exchangeable for one 48-hour extension on any homework assignment, no questions asked, no reason required. CPS201 dollars can't be stacked: you can spend at most one per assignment. CPS201 dollars have no cash value, no other points value, and are non-transferable. There's no reason to hoard them! Work turned in within 24 hours after the deadline will occur a 10% penalty; work more than 24 hours after the deadline will recieve no credit. Note that this *can* be stacked with CPS201 dollars: this means that it's possible to turn work in 72 hours late with a 10% penalty. If you want to use a CPS201 dollar, let us know *when you submit that assignment*, both in the text of the assignment itself, and in a separate e-mail to Mac.
2. *APTs*: We aren't accepting late APTs this semester.
3. *In-class work*: In-class work is due in class, unless you have an excused absence, in which case you should be in touch with us.
4. *Tests*: If you're going to miss a test, let us know *as soon as possible* so that we can work something out.

The standard university exemptions (short-term illness, varsity travel, religious holiday, etc.) apply to all of the above. Please let us know as soon as you can, so that we can work something out. Other than these mandated extensions and the CPS201 dollar policy above, no extensions will be given.

All work is due 24 hours before final grades are due.

Book

The (optional) textbook is "Core Java, Volume 1 — Fundamentals" by Cay Horstmann and Gary Cornell. It's available from Amazon and the University bookstore.

Academic Honesty

Academic honesty is absolutely paramount. The Duke Community Standard states

> 1. I will not lie, cheat, or steal in my academic endeavors, not will I accept the actions of those who do.
> 2. I will conduct myself responsibly and honorably in all of my activities as a Duke student.

All work in CPS201 is expected to be your own. We *strongly* encourage you to talk to your fellow students about assignments; teaching and learning from your peers is one of the best ways to understand material. However, *code and solutions must be your own*. Take understanding and insight away from these conversations, and then produce your own answers. Don't take code or solutions from your classmates, or from people who have taken the course previously, or from the internet, or from anybody else. (Similarly: don't give code or solutions to anybody.) If you talk with somebody, cite your source! There are three exceptions:

1. Some work in this class will be in groups. In that case, the work must be solely by members of the group; all of the same standards apply to the group as to any member of the group. Furthermore, every member of the group is expected to contribute.

2. You may freely use code provided by a member of the course staff (instructor, TA, or UTA), so long as you cite your source.

3. Code provided as part of the assignment itself can be used without citation.

On academic honesty, the late policy, and everything else: if you're unsure about something *ask for clarification*. Permission may be given: forgiveness will not.