An introduction to **software design** and development concepts, methods, and tools using a statically-typed object-oriented programming language such as **Java**. Topics from: **version control, unit testing**, **refactoring**, **object-oriented design and development**, **design patterns**, **advanced IDE usage**, **regular expressions**, and **reflection**. **Representation of floating-point numbers** and introduction to numerical computation.
Prerequisite

Prerequisite: 60% in CSC148 or CSC150


If you do not have the prerequisite but have a strong grasp of the topics listed above, please contact me about a prerequisite waiver TODAY!

Include your complete academic history from ROSI.
Prerequisite knowledge

Language structures:

- variables, aliasing, control structures (if, while and for)
- strings, lists, dictionaries, linked lists, trees, classes

Concepts:

- recursion, searching, sorting, object-orientation

Skills:

- good coding style, solid tracing and debugging
- top-down design, OO decomposition
- asking (and answering) questions; knowing what you don’t know and how to find out!
Do you know this material already?

Contact me if you are comfortable with all of the following:

- Java, JUnit, and IDE usage
- using a UNIX shell (the basics)
- using version control
- object-oriented design and design patterns
- regular expressions
- working in a software development team

Regardless, you’re welcome to stay!
Enrolment woes

Most sections are full with waiting lists, so why is L0301 nearly empty?

A new round of CSC subject POSt admission is happening at the end of this week.

To give new admits a chance to get in, the course is currently restricted to students in a CSC subject POSt.

Once the new admits have had a chance to enrol, the course will open to others. Probably Tuesday or Wednesday.

Questions? The undergrad office is glad to help.
Coursework

Labs (8): 4%
Exercises (3): 7%
Assignment: 8%
Project: 24% (divided into phases; completed in teams)
Midterm: 12%, on paper, 50 minutes
Final examination: 45%, on paper, 3 hours
  • must get 40% on the exam to pass the course; otherwise the final grade will be no more than 47
Coursework — Labs

There will be eight required labs.

These are worth 4% of your grade (0.5% each).

Lab room assignments will be posted on the course website before the first lab. They will take place in Bahen.

There will also be a couple of labs during which team meetings or demos will take place.

Ask your TA questions — they are there to help you!
Coursework — Exercises

Three exercises, worth 1%, 3%, and 3% each.

Small, done individually.

Topics are chosen to prepare you for more significant pieces of work: the assignment, project, and tests.

Pre-marked:

- All marking is automatic, done by a script.
- The auto-tester will run several times over the course of a couple of days prior to the submission deadline.
- The results of the auto-tester will appear on MarkUs.

To earn the marks for an exercise, all tests must pass. Otherwise, the exercise mark will be 0.
Coursework — Assignment

One assignment. (Don’t worry, there’s a project later.)

Done individually.

Topics:

• Java
• Basics of Object-Oriented Design
• Java Generics
• Understanding Basic UML
Coursework — Project

Phase 0:
• Register partnership; submit teamwork-related information.

Phase 1: Design (teams of two)
• You choose your partner from your tutorial section.

Phase 2: Implementation and Testing (teams of four)
• Instructor forms teams by merging 2-person P1 teams.

Phase 3: Implementation and Testing (teams of four)
• Same team as in Phase 2.

We will use elements of Scrum, a modern software development process widely used in the industry today.

Your grade will depend on the finished product and also significantly on your development process and teamwork.
Coursework — Project Demos

Twice in the term, you will give a demo to your TA.

• As part of Phase 1, each team will meet with their TA to demonstrate their P1 submission.

• Phase 3 is divided into two sub-phases and as part of Phase 3a, each team will meet with their TA to demonstrate their P3a submission.

Every member of the team must contribute to the demo, and be prepared to answer questions on any aspect of the work.
Coursework “handouts” and submission

• All work will be submitted using subversion — you will learn more about subversion soon.

• You will have your own individual repository, and repositories for your teams.

• You will get the assignment/exercise/project handouts by checking them out from the repository.

• You will submit your work by checking it in into your repository.

**Important:** You will NOT submit online using MarkUs. Web submission is disabled for this course.
Textbook

There is no required textbook.

If you wish to own a good book on object-oriented software design, we recommend:

Academic Offences

Don’t cheat.

Seriously, don’t cheat.

It is an offence to claim someone else’s work as your own or to give someone your work.

You must not share work outside your team or seek out inappropriate aid.

If in doubt of what’s appropriate, ask me.
Getting (and giving) help

Course website:
http://www.cdf.toronto.edu/~csc207h/fall/

Discussion board (Piazza):
http://piazza.com/utoronto.ca/fall2015/csc207

Labs: the TAs are there to help!

Office hours (you are free to see both Jonathan and me):
• See website for who, when and where

Help Centre:
• Mon to Thu, 4-6 pm in BA 2230

Email:
• Do not email your TAs: they are not supposed to answer.
• Email me for personal matters only.
• Post your questions on the discussion board instead!
Doing your work

• To install Subversion, Java, and Eclipse on your own computer, follow the instructions on the Software webpage.

• All of the tools you need are also available on CDF.

• The Software webpage includes instructions for how to access CDF remotely, but working via a remote connection will be slow compared to working locally.
Questions?

. . .  First topic: version control