Course Wrap-up

6 August 2015
Course Evaluations

Very important! They’re used by:

Future students in choosing courses
Instructors for improving the course
University for evaluating instructors

The evaluations will only be done online (not on paper).
Please complete them! Deadline August 10th.

Current Response Rate:
14.85%
Administration

Last day for A2 remark requests is Friday August 14th.

I will post your term marks before the final exam on MarkUs.

When A3 results are available, we’ll post an announcement on Portal.

Please come by to pick up your midterm if you haven’t done so already.

If you have any questions about any of the posted marks please come and talk to me.
Want to do more CS?

Next courses:

CSC148: Introduction to Computer Science.
More on algorithms, data structures, analysis.

CSC165: Mathematical Reasoning for CS
Analyzing complexity and correctness (among other things) requires math. Possibly a different kind of math than you’ve experienced.
CS Directions

Some possibilities

- Traditional directions, like software engineering.
- Artificial intelligence.
- Human-computer interaction.
- Computational biology.
- CS + something else.
- CS minor
Whatever you choose...

Get to know your profs!

Get involved in research & development:

- project course (CSC494/5)
- “capstone” course (CSC490)
- NSERC summer program (for $)
Decided on CS?

Consider doing a Professional Experience Year! (PEY)

http://engineeringcareers.utoronto.ca/students/undergraduate-internship/pey/

Open to second- and third-year U of T undergraduate students

Run by Engineering, but CS students encouraged to participate

12-16 month work placements
The Exam

3 hours
Check the schedule for time and ROOM
Bring student card
Study with old exams (from the website)
  Do the questions on paper
  Try typing in your answers
  Then check posted solutions
Covers the entire term
What to expect

Sample question styles:

write code, trace code, debug code, discuss time complexity, design test cases, short-answer questions, etc.

Help pages posted in advance

Remember: no cell phones!
What’s not on the exam?

Writing unittest test suites. But you may be asked to choose test cases.

You won’t be asked to implement particular sorting algorithms, but you should understand each algorithm and how to apply it, and be able to read and understand the code.

Passing functions as arguments (from Week 12 Prepare).

Defining your own exceptions (marked as Bonus Material in Week’s 12 Prepare).
Don’t forget the road traveled :) 

Remember this from the first lecture?

https://www.youtube.com/watch?v=nKlu9yen5nc

Or how almost all of you were still standing after I asked anyone who had ever written a program to sit down?

You’ve come a really long way in these 12 weeks! Think about all the cool things you can now do in Python!

I hope you’ll be curious to learn and code more in the future.

Keep on Learning (and coding)!

Thank you!