Course Wrap-up

2 April 2014

Course Evaluations

Very important! They're used by:

- Future students in choosing courses
- Faculty for improving the course
- University for evaluating faculty

The evaluations will only be done online (not on paper).

Please complete them! Deadline April 8th.

Current Response Rate:

- 6pm: 22.39%

Administration

Last day for A2 remark requests is Friday 4 April.

Term marks (except A3) are posted.

- See Marks webpage.
- Term mark out of 47 (excludes a3, w12 exercise)
  - (Exercises out of 18, because of best 9 out of 10)

See Jen with any questions about your marks.

When A3 results are available, we'll post an announcement on Coursera.

Office Hours/Help Sessions

Mon 7 Apr	11am-1pm	Yan (TA)	BA2230
Tues 8 Apr	3-5pm	Myrto	BA4237
Thu 10 Apr	12-1pm	Jen	BA4238
Fri 11 Apr	12-1pm	Jen	BA4238
Mon 14 Apr	1-3pm	Jen/Myrto	BA1190
Tues 15 Apr	3-5pm	Myrto	BA4237

The April 14th session lasts only from 1pm until questions run out.

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/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.

Passing functions as arguments.

Defining your own exceptions.