COMP 3760

Design and Analysis of Algorithms

BCIT Computing

Computer Systems Technology

Today’s plan

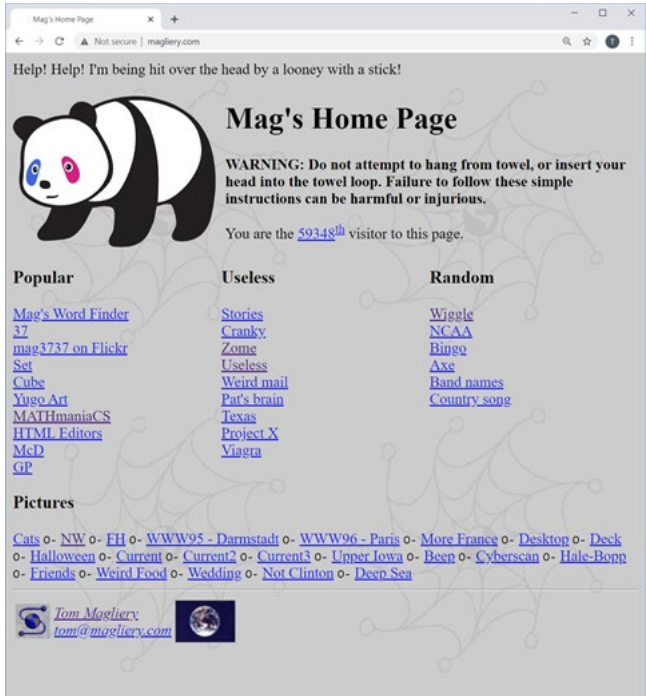
- Course intro slides (this)
- Reminder of Math You Should Know
  - *Offline review*
  - See slides on Learning Hub
  - Bring questions to Discord or Lab
- Week 1 Lecture
  - Slides on Learning Hub, lecture here & now

About me

- Tom Magliery
- Email: tmagliery@bcit.ca
- Office hours:
  - This means “Pledge to be online and reachable now”
    - Now that I am PH, change “pledge” to “do my very best”
  - Days/times TBD
  - Note: I’m online lots of other times
- Other communication:
  - Discord (server link TBD)

My history

- Undergrad: Math and Computer Science (Kansas)
- Grad: Computer Science (Illinois)
- Professional background:
  - NCSA Mosaic, HTML, XML, XMetaL
- My home page has existed since 1994 →
- Joined BCIT in 2019



Course learning objectives

- Understand the basic framework of algorithm analysis.
- Analyze pseudo-code using asymptotic notations.
- Compare the order of growth of different algorithms.
- Understand the differences between nonrecursive and recursive algorithms.
- Describe some common algorithm design strategies: Divide and Conquer, Transform and Conquer, Greedy Technique, Dynamic Programming, etc.
- Recognize different types of computing problems and how to solve them.
- Apply algorithm design techniques to solve some practical problems.
- Specify algorithms in pseudocode.
- Implement solutions by using appropriate data structures.
- Deduce the complexity of a program by running different experiments.
- Argue the correctness of the algorithms.
- Find lower bounds for some simple problems.

Textbook ("recommended")

- Introduction to The Design and Analysis of Algorithms, 3rd Ed.
  - Author: Anany Levitin

- Read the sections that relate to class material
- Look at the suggested sample problems

Course sessions

- Lectures:
  - Attendance is required
  - Lectures are NOT RECORDED

- Labs:
  - Attendance is required
  - Quizzes
  - Supplemental activities
  - Question period
  - Time to work on lab assignments

Course grade components

Course grade components

- Quizzes – 30%
- Labs – 15%
- Midterm – 25%
- Final – 25%
- Participation – 5%

- You must pass (score ≥50% on) at least *two out of three* of {Quizzes, Midterm, Final}

Quizzes – 30%

- Held during lab (EVERY WEEK!)
  - Quiz covers most recent lecture material
  - 15-20 minutes, 5-8 pts
  - Usually online (LH)
- There will be 11 quizzes
  - *Lowest quiz score is dropped*
- Each quiz is smallish, but they will add up
- Also: good practice for exams

PROBLEM:
Tuesday
stat holidays

Lab assignments – 15%

- Experimenting with the algorithms we study
- Reinforcing lecture/course concepts
- 6 assignments (all count; none dropped)
- Weekly or bi-weekly, precise timing TBD

- Other requirements
  - Language: Java
  - Environment: Whatever
  - Coding: More details with first coding assignment

Midterm (25%) and Final (25%)

- Details when the time comes

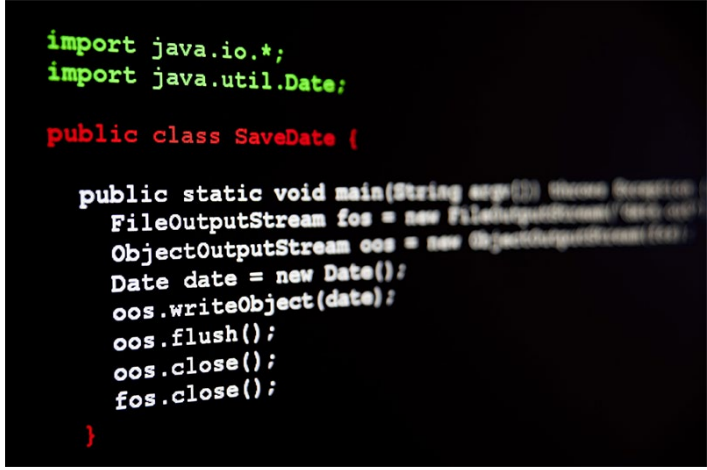
- You must pass (score ≥50% on) at least *two out of three* of {Quizzes, Midterm, Final}

Participation – 5%

- Things that can affect this:
  - Attendance
  - Raising questions/comments in class
  - Participating in lab discussions/activities
  - Generally engaging beyond minimum required levels

Other points of interest

This is not a programming course!



- Is this paradoxical? Ironic? Just plain false?
  - Maybe paradoxical – we certainly will write code
- But: We are mostly studying the *abstract idea* of algorithms
- You may actually need to suspend some of your coding skilz at times
- But we do need *some* kind of language to represent, communicate, and discuss algorithms ...

Pseudocode

- We will use pseudocode a LOT
  - Lectures, textbook, quizzes, labs, exams
- Pseudocode expresses basic programming statements
  - Variables, assignments, expressions, conditional statements, loops, subroutines
- My own style is ... ill-defined
  - I follow Postel's Law: Be conservative in what you send, and liberal in what you accept
- If ever in doubt, ask

Tips for success

- Practice!
- Keep up with the material
- Interrupt me ANY time in class
  - If I say something that confuses you, don’t assume that I know what I’m talking about
  - I make mistkaes*!

* Facts: I made that typo for real the first time I made a slide like this years ago. It was so good I left it ever since.

Virtual donuts 

- I award virtual donuts
  - Could be: planned, spontaneous, pre-announced, secret, easter eggs, unlocked achievements, ...
  - Surprise me, impress me, amuse me, ...
- Maybe someday virtual donuts will turn into *real* donuts