CS2040S+IT5003 Semester 1 2024/2025

Data Structures and Algorithms

Tutorial+Lab 01

Basic Java/Python, Basic OOP, Analysis, Hands-on 1

For Week 03

Document is last modified on: July 17, 2024

1 Introduction and Objective

The purpose of this first tutorial+lab session is to recap the first few sessions of CS2040S/IT5003: Introduction, basic Java/Python, basic analysis of algorithm, and to ensure that all students can code a simple Java/Python program using their own computer/laptop at home and submit code to Kattis for automatic judging. The first half of the session is generally the 'tutorial' part and the second half of the session is generally the 'hands-on/lab' part. The tutors will control the timings and they don't have to divide the sessions exactly by half. There will be a short break during the transition.

As this is the first session, we will do a quick ice breaking at the start of the session. Your TA is your main contact person of CS2040S/IT5003 related queries this semester. Only contact Prof Halim for questions that you are sure that your TA cannot (or has no privilege to) answer, e.g., questions about class policy, appeal for plagiarism verdict (if caught beyond reasonable doubt and you want to plead your case), etc.

To get the most out of the tutorial part of these sessions, please try out all the questions in the tutorial component and give some answer even if you encounter difficulties in answering some of them. Before, during, or after the tutorial session, don't hesitate to clear up all doubts and questions you might have, with the tutor.

Every week, you will try to solve one medium (or two easy) selected Kattis problem during the 'hands-on/lab' component. The tutors already know the selected Kattis problems for this semester. However, these selected problems will be revealed to you on the spot each week (if you happen to already solve it, then you are free to just leave the session or actually you can stay back to help your peers – you can learn more things by observing how others approached the problem that you have solved – possibly with just one way). Tutor will guide all students to get (near) Accepted solution for each problem. These problems are not graded but attempting them during the hands-on time

(and possibly to fully complete them afterwards) is beneficial to better understand CS2040S/IT5003 material.

The tutorial/lab participation marks are there to encourage class participation. These marks will be given by the tutor **at the end of the semester** using the following guideline:

- 0% if you only attend ≤ 5 out of 9 for IT5003/11 for CS2040S tutorial/lab sessions,
- 1% for **at most the bottom three** most-passive students (assuming these students attend > 5 tutorial/lab sessions),
- 3% for **at least the top three** most-active students (answering questions when asked by TA – the correctness of your answers are secondary; or even just by asking your own questions to TA before/during/after class/during consultation); in each tutorial group, and
- 2% for the rest.

2 Tutorial 01 Questions

Q1). Please scrutinize `ListArrayTest.java` (CS2040S) or `ListArrayTest.py` (IT5003).

This code will be revisited soon during discussion of List ADT (read `https://visualgo.net/en/list?slide=2-1` until 2-8). For now, please answer the following sub-questions (see the comments inside the source code):

(a) What does this line means? (for IT5003: can Python accept ListArray of any type?)

(b) Anything wrong with this line? (for IT5003: is it really an issue?)

(c) Any potential issue with this line?

(d) What is the difference of this 'add' versus the other 'add' (for IT5003: why can't we do that, i.e., the second 'add' is 'add_at_index')?

(e) What does this line means?

(f) What if we use this commented line instead of the line before it? Any potential issue?

(g) Any potential issue with this line?

(h) What does this line means? (for IT5003: isn't it easier to do this in Python?)

Analysis/Order of Growth

Q2). What is the tightest bound of the following function? $\mathbf{F}(n) = \log(\mathbf{2}^n) + \sqrt{n} + \mathbf{100\,000\,000}$

1. $O(n)$
2. $O(n \log n)$
3. $O(n^2)$
4. $O(1)$
5. $O(2^n)$

Q3. What is the tightest bound of the following functions below: $\mathbf{F}(n)$, $\mathbf{G}(n)$, and $\mathbf{H}(n)$?
The options:

1. $O(2^n)$
2. $O(n^2)$
3. $O(n \log n)$
4. $O(n)$
5. $O(\log^2 n)$

6. O($\log n$)
7. O(1)
8. none of the above

Q3.a). $\mathbf{F}(n) = n + \frac{1}{2}n + \frac{1}{3}n + \frac{1}{4}n + \frac{1}{5}n + \frac{1}{6}n + \frac{1}{7}n + \frac{1}{8}n + \mathbf{... + 1}$

Q3.b). $\mathbf{G}(n) = n + \frac{1}{2}n + \frac{1}{4}n + \frac{1}{8}n + \frac{1}{16}n + \frac{1}{32}n + \frac{1}{64}n + \frac{1}{128}n + \mathbf{... + 1}$
$\mathbf{G}(n)$ is basically the sum of the reciprocals of powers of two up to n.
To simplify the analysis, assume that n is a powers of two.

Q3.c). $\mathbf{H}(n) = n + \frac{1}{2}n + \frac{1}{3}n + \frac{1}{5}n + \frac{1}{7}n + \frac{1}{11}n + \frac{1}{13}n + \frac{1}{17}n + \mathbf{... + 1}$
$\mathbf{H}(n)$ is basically the sum of the reciprocals of prime numbers up to n.
To simplify the analysis, assume that n is a prime number.
The analysis is very mathematical...
For CS2040S/IT5003 level, just estimate the time complexity of $\mathbf{H}(n)$ w.r.t $\mathbf{F}(n)$ and $\mathbf{G}(n)$.

Hands-on 1

TA will run the second half of this session with one chosen Kattis problem involving List ADT.

Problem Set 1+2

We will end the tutorial with a short PS1 debrief and an overview of PS2. Recall that:

- IT5003 students are expected to solve problems A+B (with the (much) harder and/or out-of-syllabus problem C optional and not-graded for IT5003 students)
- CS2040S students are expected to solve problems B+C (with the easiest problem A as optional and non-graded for CS2040S students)