Data Structures

Lecture 1

Christian Lim

Monday, January 22, 2024

Motivations

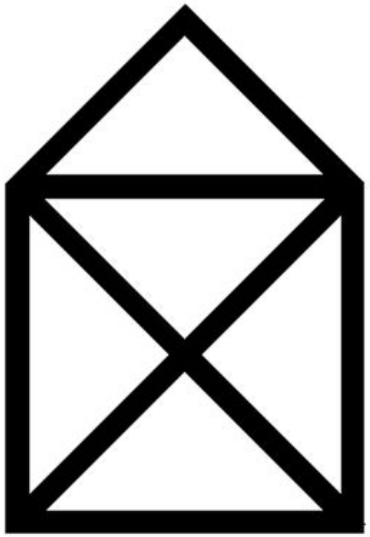
- Writing code that runs efficiently.
  - Good algorithms.
  - Good data structures.
- Writing code efficiently.
  - Building, testing, and debugging.
  - Use of programming tools.
    - vscode, unit test, and various command line tools.
  - C++ (not the focus of the course!)

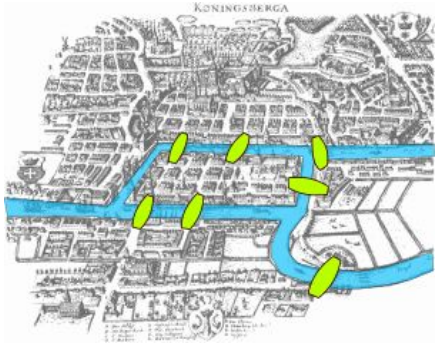
Why study data structures?

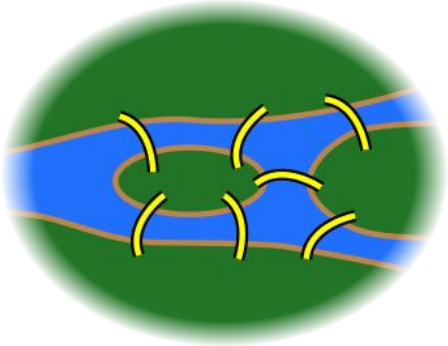
- To become a better programmer.

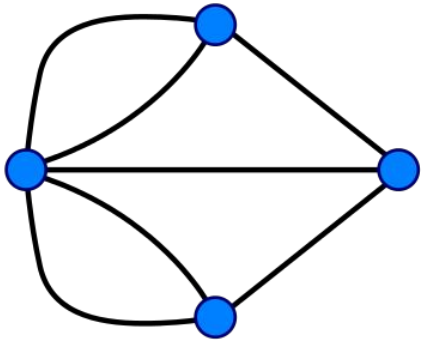
- Being an efficient programmer means using the right data structures and algorithms for the job.

Why study data structures?









Christian Yongwhan Lim

Education

Stanford University

Part-time Jobs

Full-time Job

Google Research

TWO SIGMA

Workshops

Stanford ENGINEERING | Stanford Computer Forum

Coach/Judge

icpc International Collegiate Programming Contest

icpc.foundation

advancing the art and sport of competitive programming

https://www.yongwhan.io

Christian Yongwhan Lim

- Currently:
  - **CEO** (Co-Founder) in a Stealth Mode Startup;
  - **Co-Founder** in Christian and Grace Consulting;
  - ICPC **Internship Manager**;
  - ICPC **North America Leadership** Team;
  - Columbia ICPC **Head Coach**;
  - ICPC **Judge** for NAQ and Regionals;
  - ICPC NAPC **Trainer**;
  - **Adjunct** (Associate in CS) at Columbia;

https://www.yongwhan.io

Lectures

- **Mondays** and **Wednesdays** from **1pm ET** to **2:15pm ET**
- @ **LL-914**, in-person only!
- All course materials will be posted on Fordham Blackboards: https://fordham.blackboard.com/ultra/courses/_6182305_1/outline

No Textbook

- There is no required textbook

Allowed Languages

- **C/C++**
- Java
- Python

Course Descriptions

- A survey of the major types of structures in programs used to handle data, including arrays, lists, stacks, queues, trees, etc.
- Methods for data organization and access will be covered across data structures.
- Space and time efficiencies will also be discussed.
- This course builds on principles in discrete mathematics and in fundamental programming practices.

Proper Citation

- If you refer to an additional resource, you **MUST** cite the source using comments.
- To put the citation at the top of the code,
  - In C++, for example, you MUST use "//" or "/* */".
  - In Python, you MUST use '#".

- **Take a special care with the proper citation as, there is:**

NO EXCEPTION TO THIS CITATION RULE

Proper Citation

- **CodeForces Contest Rules** (**MikeMirzayanov**)

- Especially, take a careful note of the "Can-do's and Can't-do's" section.

Programming Assignments (50%)

- A problem set will be assigned through CodeForces.
- Make sure to create an account in CodeForces.
- It will be assigned on a biweekly basis (every two weeks).

LeetCode (30%)

- Weekly or biweekly contests will serve as quizzes.
- Top 3 results will be used for evaluation.

Final Exam (20%)

- There is one cumulative final exam.

Deliverables

- Nothing, since they are **auto-tracked**!

Attendance

- Attendance is **required**.

Course Structure

- Each week, we will have:
  - (in-person) On Mondays and Wednesdays, there will be a 75-minute lecture with **required attendance**.
  - (online) On Saturdays, there will be a weekly or biweekly LeetCode contest.
  - (online) Every other week, programming assignments will be due.

Grade Breakdown

- **(50%)** Programming Assignments
- **(30%)** LeetCode weekly or biweekly
- **(20%)** Final Exam

C++ Tips and Tricks: best to learn those through practice!

- **C++ Tricks (HosseinYousefi)**
- **C++ tips and tricks (Golovanov399)**
- **Some Tips for Coding in C++ in Competitive Programming (Nea1)**

- Use "**#include <bits/stdc++.h>**" header to include **almost everything**.

Standard Input/Output (stdio)

- **Yet again on C++ input/output (andreyv)**

- **scanf/printf** vs **cin/cout**
  - Often, use "ios::sync_with_stdio(0); cin.tie(0); cout.tie(0);"

Setting up accounts in LeetCode

- https://leetcode.com/

Setting up accounts in CodeForces

- https://codeforces.com/

In closing...

- As you can see, we have **A LOT** of topics to cover...

- So, the focus will be on learning data structures in terms of its:
  - core/essential ideas
  - implementation details
  - application
- The explicit non-goals are:
  - proofs
  - rigorous treatments


THANK YOU