HAUSCR Summit for Young Leaders in China 2020

Introduction to Data Structures and Algorithms

Max Guo

Shanghai

Introduction to Data Structures and Algorithms

Introduction

Dear HSYLC Participant,

Welcome to HSYLC 2020! I am Max Guo, the seminar leader for Introduction to Data Structures and Algorithms. This seminar will provide an overview of fundamental concepts in data structures and algorithms. Throughout the course of HSYLC, we will cover various methods of analyzing and constructing algorithms. We will go over many well-known algorithms and their implementations in modern programming languages (especially Python). Don't worry if you do not have much exposure to computer science or coding! A background in mathematics or programming will help, but an eagerness to learn is all that is required to gain from this course. I do believe that these topics can sometimes be quite challenging when trying to learn by yourself, so it is my hope that this seminar may propel you in your coding journey!

Personally, I became interested in mathematics through my middle school and early high school years, and this interest has extended to topics in computer science (which is quite related to mathematics in many ways!). Now a rising sophomore at Harvard, I study math and computer science, and I actively teach these concepts through conferences like HSYLC. I am also involved in machine learning research on campus through the School of Engineering and Applied Sciences. Other than academics, I enjoy playing the piano and playing board games with friends, especially those involving strategy and social deduction.

The purpose of this seminar is to introduce fundamental concepts of data structures and algorithms to students who are interested in but relatively unfamiliar with the field. The topics are intended to be challenging, so don't be worried if you don't understand much at first! Though I do not expect all the information to necessarily be retained upon the first encounter, I hope that this seminar exercises your problem-solving skills and sparks your curiosity about computer science!

I look forward to meeting you virtually soon!

Max Guo

Seminar Description

Seminar Leader: Max Guo
Concentration/Class Year: Math/Computer Science, Class of 2023
Title of Seminar: Introduction to Data Structures and Algorithms
Seminar Category: Sciences and Engineering
Seminar Description: This seminar will provide an overview of fundamental concepts in data structures and algorithms. Students will develop a foundation in applying mathematical thinking to the design and analysis of algorithms. Topics will include big-O notation, induction, recurrences, basic data structures (linked lists, stacks, queues), mergesort, graphs, graph representations, graph search algorithms (DFS, BFS), and applications of graph search algorithms. The seminar will also include code implementations of these algorithms so that students may have an idea of how these topics are relevant in practice. Students should have a solid mathematical background - **familiarity with proofs and general problem-solving would be very helpful, and some prior coding experience is not required but recommended** (students should be familiar with concepts of conditionals, loops, etc.).

Syllabus

Day By Day Outline:

1. **Day 1: Overview: Math review, Terminology, Big-O notation**
    a. Introductions and Icebreakers
        i. We will get to know each other a little better by playing a few icebreaker games through Zoom!
    b. Overview of Algorithms
        i. We will give the definition of an algorithm and several examples of classic problems that algorithms may solve, as well as where all of this discussion fits into the real world.
    c. Big-O notation
        i. We will give the definition of Big-O notation and then illustrate the results through the graphs of various functions. Many examples will be given.
    d. Induction
        i. We will show how to prove the correctness of algorithms that undergo many iterations through the mathematical concept of induction. We will go through examples that involve proving identities regarding the integers.
2. **Day 2: Basic Data Structures and Search Algorithms**
    a. Basic Data Structures

i. Some of the basic data structures are arrays, linked lists, stacks, and queues. We will cover many of their basic operations and analyze their runtimes.

b. Search Algorithms

i. There are several different search algorithms depending on the assumptions of the data structure being searched, but linear and binary searches will be covered. Depending on time, proof of the correctness of binary search may be covered.

3. **Day 3: Recurrences and Sorting**

a. Sorting

i. Insertion sort, selection sort, bubble sort, and mergesort will be covered in varying depths. A proof of the correctness of mergesort will lead straight into the next section on recurrences.

b. Recurrences

i. Various methods of guessing and using recursion trees will be covered to give intuition behind solving recurrences, and finally an overview of the master theorem. Some time will be given for the class to work together on an application problem.

4. **Day 4: Graphs and Graph Searches (I)**

a. Graphs

i. We will cover the mathematical definition of a graph, vertices, edges, cycles, trees, directed/weighted graphs, paths, and more. Some properties of graphs may be covered, as well as different representations of graphs in computer science (adjacency matrix, list).

b. Searching a graph - DFS

i. We will cover the motivation behind searching graphs (with many real world applications), and then walk through a common graph traversal algorithm (depth-first search). The Python implementation of this will be discussed. We will also have an interactive walkthrough of the DFS algorithm.

5. **Day 5: Graphs and Graphs Searches (II)**

a. DFS Part II

i. The many applications of DFS will be discussed, including various mathematical theorems regarding the DFS tree, preorders and postorders, cycle detection, and more. This will include a **class activity**.

b. BFS

i. We will introduce and explain the other primary graph search algorithm, breadth-first search, and include a walkthrough and a discussion of its implementation in Python. Applications including shortest-path algorithms will be discussed.

6. **Day 6: More Applications of Graph Searches**

a. Continued Applications of DFS and BFS
    i. This section will emphasize the importance of graph searching through many example problems that can be solved with various graph searching techniques. This will likely be a group activity and problem-solving session.
b. Where to go from here?
    i. Much of this course served to simply introduce the student to concepts in computer science, but there is so much more out there to discover and learn! The concluding time in the seminar will give resources for the students who are interested in learning more in the field or similar areas of study.

Optional Readings:

1. **Reading 1: Introduction to Data Structures and Algorithms.**
    a. StudyTonight. https://www.studytonight.com/data-structures/introduction-to-data-structures.
    b. This source provides a very good outline of many topics we will cover in our seminar, so it's worth a skim. Don't be too bothered if you don't understand everything!
2. **Reading 2: Induction Overview.**
    a. PurpleMath. https://www.purplemath.com/modules/inductn.htm.
    b. This source gives a great overview of induction. It's worth reading in depth, although the concept of induction will also be covered in depth on the first day of the seminar.

Glossary:

- **Algorithm**: A process or set of rules to be followed in calculations or other problem-solving operations.
- **Data Structure**: Data organization, management, and storage format that enables efficient access and modification
- **Python (3)**: A programming language that can be used to implement algorithms and data structures
- **Sorting**: The problem ordering a list of numbers (usually from least to greatest). Usually a very common problem in the context of algorithms.
- **Recursion**: Repeatedly calling a function inside another function. If you are unfamiliar with this term, it may be useful to look it up beforehand as well.
- **Proof**: A rigorous argument for the correctness of a statement.
- **Induction**: A method of proof that involves two steps for proving a statement holds for all positive integers. The first step is proving the statement is true for a base case. The second step is proving that, if a statement is true for one case, it is true for the next case.

Homework:

1. **Homework Assignments 1**
    a. This homework assignment will consist of a short set of problems that covers material from the seminar. The first homework will likely focus on applications of sorting, searching, and recursion.
    b. Weighed 25% of the total grade.
2. **Homework Assignment 2**
    a. Much like homework assignment 1, this assignment will also consist of a short set of problems that covers seminar material. This assignment will focus on reviewing depth-first and breadth-first search.
    b. Weighed 25% of the total grade.
3. **Class Participation**
    a. Students should actively participate in discussion and activities, as it is a valuable part of the learning experience.
    b. Weighted 30% of the total grade.

Final Exam (Discussion): Grade Weight: 20%

The final will consist of a discussion where the students may ask the instructor about any topics they might have more interest in learning about, academics in high school and/or college, college life, or any other "bigger picture" topics. This discussion will be evaluated based on student engagement.