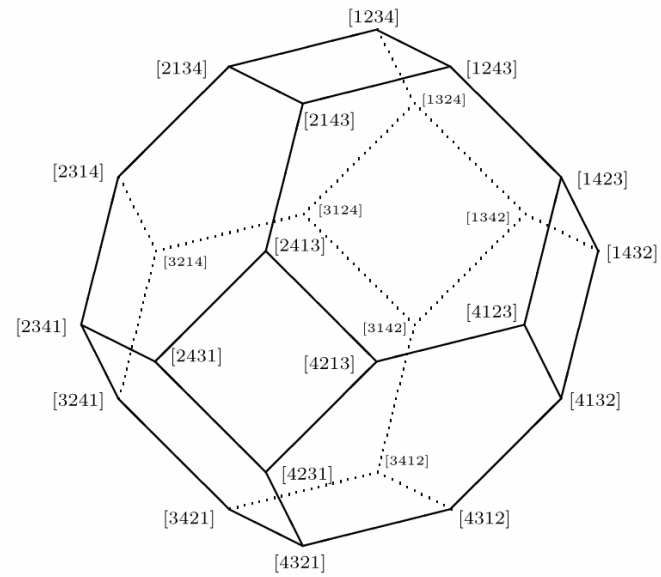
CS3334 Data Structures



Chee Wei Tan

Lecturer and Instructors

- Prof. Tan, Chee Wei

Office hours: Monday 12pm-1pm, AC 1 Room G7307
Email: cheewtan@cityu.edu.hk
Website: http://www.cs.cityu.edu.hk/~cheewtan/teaching.html

Teaching Assistants

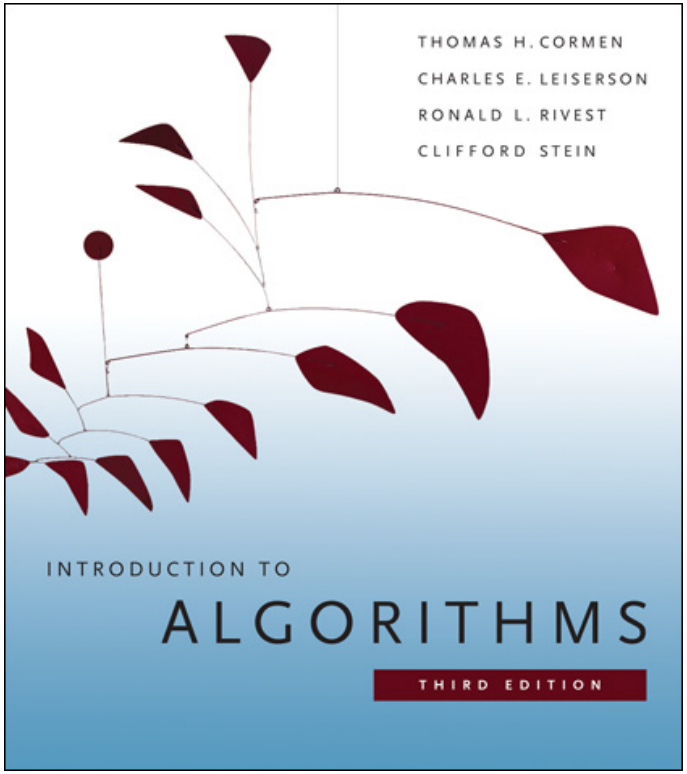
- Ms. Li, Jingting
- Mr. Ling, Alex
- Nemo Bot (Facebook Chatbot Tutor)

Goals

- In this course, you should have some familiarity with C/C++ programming.
- You can use other programming languages you are familiar with in the coursework
- After this course, you should know
  - Sorting and search algorithms
  - Data structures
  - Time and space complexity analysis
  - Problem solving skills

Reference Book

Thomas Cormen, Charles Leiserson, Ronald Rivest, and Clifford Stein. **Introduction to Algorithms**. 3rd ed. MIT Press, 2009. ISBN: 9780262033848.



(Image source: http://en.wikipedia.org/wiki/Introduction_to_Algorithms)

Tentative Schedule

Week	Date	Topics
1	Jan 14	Complexity Analysis
2	Jan 21	Recursive Functions & Binary Search
3	Jan 28	Array, Linked List, Stack & Queue
4	Feb 11	Bubble Sort & Insertion Sort
5	Feb 18	Merge Sort
6	Feb 25	Quick Sort
7	Mar 4	**Mid-term**
8	Mar 11	Heap Sort, Bucket Sort & Radix Sort
9	Mar 18	Hash Table
10	Mar 25	Binary Search Tree
11	Apr 1	AVL Tree
12	Apr 8	Disjoint-set
13	Apr 15	Revision and Demo

Assessments

- **Course Work: 30%**
  - Class Quiz: 4%
  - One Assignment: 6%
  - One Course Project: 10%
  - One Mid-term Examination*: 10%

  (*A make-up mid-term examination will only be arranged for a student who is absent from the examination due to extenuating circumstances such as illness, injury or other personal emergencies. The student must contact me within three working days from the date of the missed examination and provide official written documents. If the absence is due to illness, the student should include both a sick leave certificate completed and signed by a qualified medical practitioner recommending for sick leave on the date of the missed examination.)
- **Final Examination: 70%**
  - For a student to pass the course, at least 30% of the maximum mark of the examination must be obtained.

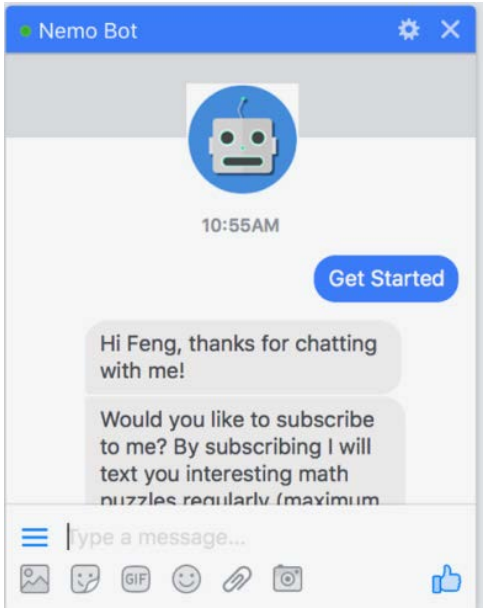
- **You must adhere to CityU's Rules on Academic Honesty** (http://www.cityu.edu.hk/provost/academic_honesty/rules_on_academic_honesty.htm)

Software for Learning

- Nemo Bot

https://www.facebook.com/Nemo-Bot-454163798317367

Used to administer in-class quizzes that are multiple-choice questions



https://xkcd.com/329/

09:41 100%

Home Nemo Bot

Nemo Bot

51 people like this, including Chee Wei Tan and 30 friends

Education

17:25

Get Started

Welcome back, Alex!

Learn

Which topic would you like to learn about?

Algorithms Introduct...

Time/Space Complexit...

Big-O Notation

Aa

09:41 100%

Home Nemo Bot

hardware independent. An algorithm that needs 10 seconds to run on computer A may only need 5 seconds to run on a more powerful computer B, so measuring time complexity using execution time is not an ideal choice.

On the other hand, Big-O notation counts the number of operations an algorithm needs, which only depends on the algorithm itself

Here are a few examples to help you understand the concept. Please choose one to see the details.

Bubble Sort

Binary Search

Power

Aa

09:41 100%

Home Nemo Bot

Hi Alex, here's a new puzzle for you!

What is the worst-case time complexity of bubble sort?

```
def bubble_sort(lst):
    for num in range(len(lst)-1, 0, -1):
        for i in range(num):
            if lst[i] > lst[i + 1]:
                tmp = lst[i]
                lst[i] = lst[i + 1];
                lst[i + 1] = tmp

lst = [3, 2, 1, 4, 0]
print(lst)
bubble_sort(lst)
print(lst)
```

A: O(n)

B: O(log n)

C: O(n^2)

D: O(n log n)

Please select the correct answer by clicking one of the

Aa

09:41 100%

Home Nemo Bot

C

D

C

Your answer C is correct ✅! Well done!

Do you want a brief explanation on this puzzle?

Sure

No, Thanks

Sure

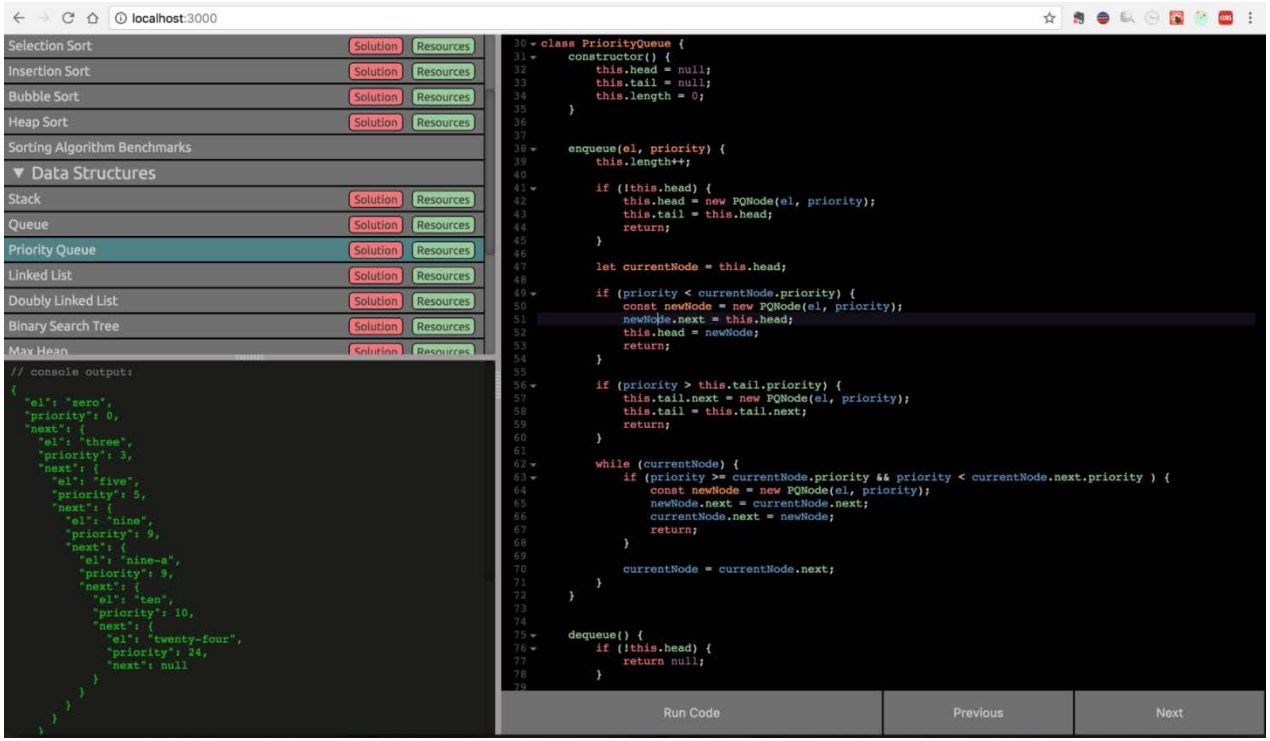
If the array is initially reversely sorted, the algorithm has to go through the array n - 1 times, making the worst-case time complexity O(n^2)

Aa

Software for Learning

- http://cs-playground-react.surge.sh

in-browser JavaScript sandbox for learning and practicing algorithms and data structures



- Alex Li's Algorithm Anthology Github

https://github.com/alxli/Algorithm-Anthology/tree/master/Section-3-Data-Structures

Motivation

- "Before there were computers, there were algorithms."
  - Cormen, Leiserson, Rivest and Stein, Preface of Introduction to Algorithms
- What are some of those algorithms that you know?
- Before there were algorithms, there were abstract data types. What do you see in an integer number ?

Motivation

- "But now there are computers, there are even more algorithms, and algorithms lie at the heart of computing"
  - Cormen, Leiserson, Rivest and Stein,
    Preface of Introduction to Algorithms

- How do you see the role of data structures in computing?

Motivation

- Bitcoin, Blockchain, Crypto-currency
  - The hottest data structure in town