CSE 207: Data Structure and Algorithms - II

Dr. Mohammed Eunus Ali

Professor

CSE, BUET

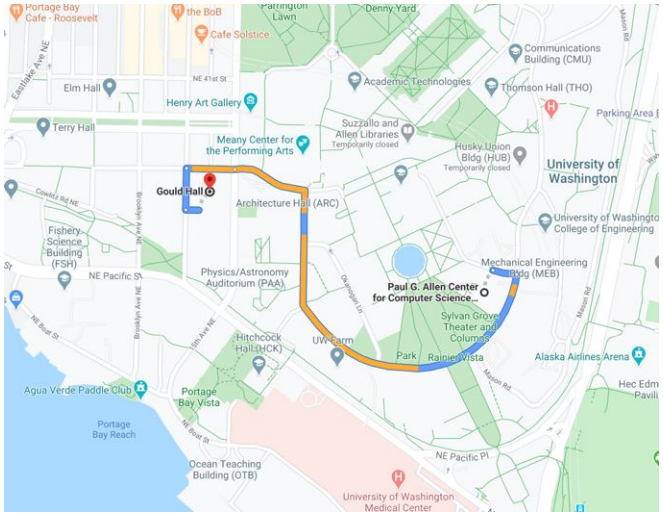
Course Teachers & Textbook

- Instructors
  - Dr Mohammed Eunus Ali (Part A)
  - Abdur Rashid Tushar (Part B)

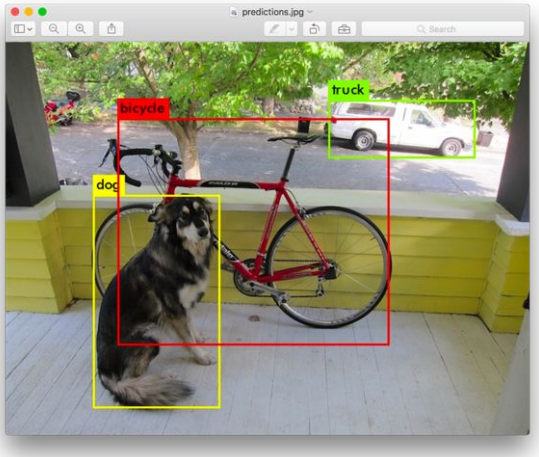
- Resources
  - Slides
  - INTRODUCTION TO ALGORITHMS (3rd Edition)
    - Cormen, Leiserson, Rivest, Stein
  - Data Structures and Algorithms
    - Goodrich, Tamassia

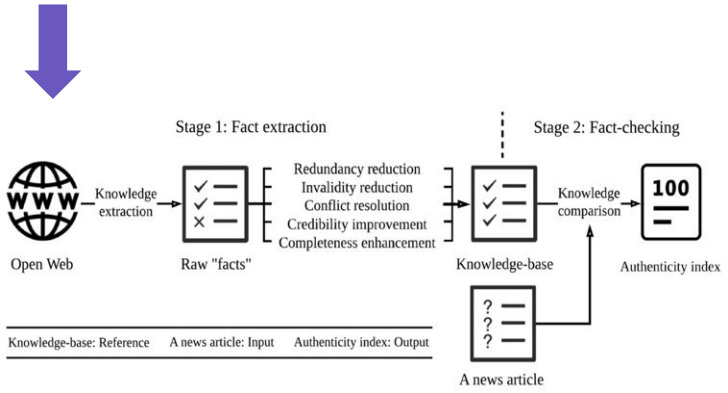
Why 207?

1. Build a strong foundation of data structures and algorithms that will let you tackle the biggest problems in computing



207 Data Structures & Algorithms II





Fake News: A Survey of Research, Detection Methods, and Opportunities (Xinyi Zhou, Reza Zafarani/arXiv:1812.00315)

Why 207?

2. Pick up the advanced vocabulary, skills, and practice needed to make **design decisions**. Learn to **evaluate** the tools in your CS toolbox

What will we study?

- Advanced Graph Algorithms
  - Minimum Spanning Trees (MST), Shortest Paths etc.
  - Maximum flow and maximum bipartite matching, etc.
- Advanced Data Structures
  - Balanced binary search trees -AVL trees
  - Red-black trees, splay trees etc
  - Advanced heaps (Fibonacci heaps, binomial heaps)
- Algorithmic Theory
  - Lower bound theory
  - NP-Hard, NP-Completeness
- Complex algorithm design
  - Coping with hardness: Backtracking, branch and bound
  - Approximate Algorithms

Data Structures & Algorithms

- Data Structure:
  - A way of organizing, storing, accessing, and updating data
  - **Examples**: Arrays, Linked Lists, Stacks, Queues, Trees, Heaps
- Algorithm:
  - A series of precise instructions to produce a specific outcome
  - **Examples**: Binary Search, Backtracking, Greedy, Dynamic
- Program:
  - A program is the expression of an algorithm in a programming language

Data Structure + Algorithms

Examples: Graph + Breadth First Search

Correctness

- How do you know an algorithm is correct?
  - For every input instance, it halts with the correct output
  - Since there are usually infinitely many inputs, it is not trivial

Efficiency

- Correctness alone is not sufficient
- Brute-force algorithms exist for most problems
- To sort *n* numbers, we can enumerate all permutations of these numbers and test which permutation has the correct order
  - Why cannot we do this?
  - Too slow!
  - By what standard?

Why Study Algorithms and Data Structure

- You will write better, faster, more elegant code.
- You will be able to solve new problems.
- You will be able to give non-trivial methods to solve problems.
- You will improve your research skills in almost any area.
- It’s one of the most challenging and interesting area of Computer Science.

Why Study Algorithms and Data Structure

- Almost all big companies want programmers with knowledge of algorithms: Microsoft, Apple, Google, Facebook, Oracle, IBM, Yahoo, NIST etc.
- In most programming job interviews, they will ask you several questions about algorithms and/or data structures. They may even ask you to write pseudo or real code on the spot.
- Your knowledge of algorithms will set you apart from the masses of interviewees who know only how to program.

.

Course Outline (Part I)

- Week 1
  - Introduction to Graph Algorithms
- Week 2
  - MST, Shortest Paths
- Week 3
  - All Pair Shortest Paths
- Week 4
  - Maximum flow and maximum bipartite matching
- Week 5
  - Balanced binary search trees -AVL trees
- Week 6
  - Red-black trees, splay trees etc.
- Week 7
  - Advance Heaps

The End