Advanced Analysis of Algorithms

COMS3005

Semester II

2020

Algorithm

- Self-contained sequence of actions performed to solve a problem

Analysis of Algorithms

- Determining the amount of time, storage (space) and/or other resources that are necessary to execute a given algorithm
- Specifically interested in determining a function that maps the length of an algorithm input to the number of steps it takes (time) and the number of storage locations (space).

Theoretical Analysis: Asymptotic Notations

- Big O Notation O
- Big Omega Notation Ω
- Big Theta notation Θ

Reasons for using asymptotic notation:

- Different implementations of the same algorithm have different asymptotic functions
- Allows a reasonable estimation to the complexity of the algorithm

Learning Objective

- Discuss
- Analyse
- Compare
- Implement

various classical algorithms and data structures

Learning Outcomes:

At the end of this course you should, among others, be able to:

- List the different algorithms available for a given problem
- Perform theoretical and empirical analysis of a given problem
- Identify optimal algorithms based on their time and space complexity for a given problem
- Apply simple algorithmic changes to existing algorithms to improve the time and space complexity

Learning Outcomes:

At the end of this course you should, among others, be able to:

- List the different algorithms available for a given problem
- Perform theoretical and empirical analysis of a given problem
- Identify optimal algorithms based on their time and space complexity for a given problem
- Apply simple algorithmic changes to existing algorithms to improve the time and space complexity

Required background knowledge

1. Proof techniques (direct proof, inductive proof and proof by contradiction)
2. Fundamentals in graph theory
3. Asymptotic notation
4. Basic algorithms on binary trees and graphs

Overall approach

- Introduce problem
- Discuss brute force approach
- Look at possible improvements which can be achieved from
  - Improved design techniques
  - Considering the form of the input data and pre-processing
  - Choosing good data structure
- Consider practical implications of the above improvements