CSE101: Design and Analysis of Algorithms

Ragesh Jaiswal, CSE, UCSD

Administrative Information

Administrative Information

- Course Instructor:
  - Ragesh Jaiswal
  - Office: 4122, CSE
  - *Email*: `rajaiswal@ucsd.edu`
  - Office hours: 11:00 - 1:00, Wednesday
- Course Time/Place:
  - Lectures:
    - Section B00: Tu, Th, 3:30 - 4:50pm, PCYNH 109
    - Section C00: Tu, Th, 5:00 - 6:20pm, WLH 2205
  - Discussion:
    - Section B01: F, 10:00-10:50am, CENTR 105
    - Section C01: W, 2:00 - 2:50pm, CENTR 212
- Teaching Assistants:
  - Fang Qiao
  - Anand Desai
  - Apoorve Dave
  - Yizhen Wang

Administrative Information

- Grading Scheme
  1. *Homework*: 15%
  2. *Quizzes*: 15%
  3. *Midterms*: 30% (2 midterms in class, 15% each)
  4. *Final*: 40%
- Homework and Quizzes:
  - Homework will be posted every Wednesday by 6pm and will be due the following Wednesday at 6pm. Late submissions will not be allowed.
  - There will be a quiz every Thursday on the material of the homework submitted on Wednesday.
  - You will have to upload the PDF of your homework on Gradescope. When uploading, you may be asked to indicate which page(s) correspond to which problems.
- Policy on cheating: **Students using unfair means will be severely penalised.**

Administrative Information

- <u>Textbook</u>: Algorithm Design by *Jon Kleinberg and Eva Tardos*.
  - I will be following this book very closely. So, it will be a good idea to get a copy of this book.
- Other reference books:
  - Algorithms by *Sanjoy Dasgputa, Christos Papadimitriou, and Umesh Vazirani*.
  - Introduction to Algorithms by *Thomas H. Cormen, Charles E. Leiserson Ronald L. Rivest, and Cliff Stein*.
- <u>Course webpage</u>: `http://www.cs.ucsd.edu/~rajaiswal/Winter2016/cse101/`.
  - The site will contain course information, references, homework, course slides etc. Please check this page regularly.

Recap. of Data Structures and Algorithms

Recap.

- What is an algorithm?

Recap.

- What is an algorithm?
  - A step-by-step way of solving a problem.
- How do we measure the performance of an algorithm?

Recap.

- What is an algorithm?
  - A step-by-step way of solving a problem.
- How do we measure the performance of an algorithm?
- Main ideas for performance measurement:
  - Worst-case analysis: Largest possible running time over all input instances of a given size n and then see how this function scales with n.
  - Asymptotic order of growth: The worst-case running time for large n (e.g., $T(n) = 5n^3 + 3n^2 + 2n + 10$)

Recap.

- What is an algorithm?
  - A step-by-step way of solving a problem.
- How do we measure the performance of an algorithm?
- Main ideas for performance measurement:
  - Worst-case analysis: Largest possible running time over all input instances of a given size n and then see how this function scales with n.
  - Asymptotic order of growth: The worst-case running time for large n (e.g., $T(n) = 5n^3 + 3n^2 + 2n + 10$)

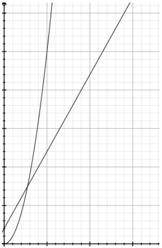

Figure : Plot of n^2 and $2n + 2$

Recap.

- What is an algorithm?
  - A step-by-step way of solving a problem.
- How do we measure the performance of an algorithm?
- Main ideas for performance measurement:
  - Worst-case analysis: Largest possible running time over all input instances of a given size n and then see how this function scales with n.
  - Asymptotic order of growth: The worst-case running time for large n (e.g., $T(n) = 5n^3 + 3n^2 + 2n + 10$)
- Asymptotic order of growth (O, Ω, Θ):
  - $T(n)$ is $O(f(n))$ (or $T(n) = O(f(n))$) iff there exists constants $c > 0, n_0 \geq 0$ such that for all $n \geq n_0$, we have $T(n) \leq c \cdot f(n)$.

Recap.

- Growth rates:
  - Arrange the following functions in ascending order of growth rate:
    - n
    - $2^{\sqrt{\log n}}$
    - $n^{\log n}$
    - $2^{\log n}$
    - $n / \log n$
    - n^n

- Algorithm: A step-by-step way of solving a problem.
- Design of Algorithms:
  - *"Algorithm is more of an art than science"*
  - However, we will learn some basic tools and techniques that have evolved over time. These tools and techniques enable you to effectively design and analyse algorithms.
- Analysis of Algorithms:
  - Proof of correctness: An argument that the algorithm works correctly for all inputs.
  - Analysis of worst-case running time as a function of the input size.

Introduction

- <u>Algorithm</u>: A step-by-step way of solving a problem.
- Design of Algorithms:
  - *"Algorithm is more of an art than science"*
  - However, we will learn some basic tools and techniques that have evolved over time. These tools and techniques enable you to effectively design and analyse algorithms.
- Analysis of Algorithms:
  - <u>Proof of correctness</u>: An argument that the algorithm works correctly for all inputs.
    - <u>Proof</u>: A valid argument that establishes the truth of a mathematical statement.
  - Analysis of worst-case running time as a function of the input size.

End