

# CS 456: Advanced Algorithms

## Notes #01

**Disclaimer:** The following is an outlined list of topics covered during **Aug. 19 - Sep. 04** lectures. This list is not a substitute for missing classes or not taking your own (much more comprehensive) notes based on in-class discussions, but rather a quick reference guide based on the instructor's own notes and recollection. The list also does not reflect the order in which the topics were discussed in the classes. *The objective* of this list is to help you refresh your memory and fill any gaps in your own notes.

## Topics Covered

- 08/19/2014
  - Defining Algorithms
  - Why learn algorithms
  - What does it mean for an algorithm to be efficient
  - The three important questions to ask about algorithms - correctness, running time, order of growth
    - \* e.g. Fibonacci Numbers
- 08/21/2014
  - Difference between a problem and a problem instance
  - What does it mean for an algorithm to be correct
  - Proving correctness using invariants
    - \* Defining Invariants
    - \* Initialization
    - \* Maintenance
    - \* Termination
  - Partial correctness vs. total correctness
  - How preconditions and postconditions relate to correctness proofs
  - Developing Invariants - e.g. Insertion Sort
- 08/28/2014
  - Analyzing Algorithms (Runtime analysis)
  - e.g. Insertion sort
  - Best case vs worse case analysis
  - Order of growth
    - \* Big O (oh) notation
    - \* Big Omega notation
    - \* Big Theta notation
  - How to use limit rule to compare order of growth
- 09/02/2014
  - Introduction to greedy strategy

- Representative problems - minimum spanning trees, Prim's algorithm, Kruskal's algorithm
- Activity scheduling problem
- Greedy strategies for the activity scheduling problem
- 09/04/2014
  - Activity scheduling problem (*contd...*)
    - \* Proving correctness using mathematical induction
    - \* Runtime analysis

Questions? Comments? - Contact the instructor at his email address.