Flipped Lecture Concepts

Guna

Focus this week

- Union-Find
  - What is it?
  - How to implement it?
  - Cost of union and find operations?
  - Applications?
- Algorithm Analysis
  - Why we need it?
  - How to estimate
  - Tilda, big O
  - Empirical to asymptotic

Concepts worth discussing

- Complexity of algorithms
  - Recursion to recurrence
  - Solving recurrences using telescoping sequence
  - Eg: $T(n) = 2T(n/2) + 1$
- From iterative code to recursive code
  - Binary search
- Counting operations in nested loops

Concepts worth discussing

- Why we remove lower order terms from the order of growth functions?
- Why are Upper bounds and lower bounds of algorithm analysis important?
- What is easier to prove? lower or upper bounds? Why?
- Are lower and upper bounds obtained from the same algorithm for solving the problem?
- How do we compare two algorithms? One f(n) and other g(n). Which one runs faster?

Cost of computing

- What are the factors affecting cost of computing?
- How expensive are these operations?
  - Cost to compare two ints, strings?
  - Cost to swap two variables?
  - Cost to read and write variables into memory
- Given an algorithm, how do we develop a cost model?
- what is abusive string concatenation?

Questions worth asking

- Java Arrays take 24 bytes? Why?
- Should we really care about worst case analysis? Why or why not?
- How do we choose worst case and best case data? Is it possible to do that for a given algorithm?
- What is your question worth asking?