Mid-Quarter Summary

CSE 421 Winter 2023
Lecture 16

What have we seen so far?

Stable Matchings

Graph Search

- BFS/DFS
- Graph modeling

Greedy Algorithms

Divide and Conquer

Dynamic Programming

Stable Matchings

Modeling matters!

It's better to be a proposer than a chooser!

Algorithms can be used to prove 'non-computational' facts

Stable Matchings always exist is easiest to prove by saying "here's how to find one."

Reductions

Sometimes there's a clever way to use an existing library (we'll need these a lot later in the quarter).

Graph Search

BFS and DFS search through a graph differently

So you can adapt them to solve different problems!

Use libraries

Finding SCCs and Topological sorting are “almost free” preprocessing

2-Coloring can be performed in linear time.

Greedy

Code is easy; proofs are hard.

Generating examples is **extremely** important.

To frame your thinking for proofs

Greedy stays ahead

Exchange argument

Structural result

Divide and Conquer

Trust the recursion.

Don’t be afraid to change what the recursive call gives you!

Add extra parameters!

State in English what the recursive call gives you.

In your “combine” step, make sure you’re beating baseline!

Dynamic Programming

Focus on solving the problem recursively; everything else is (mostly) formulaic once you've done that.

Write exactly the problem you're solving in English.

It's better to get down a "guess" at the problem and then see where you get stuck.

Don't be afraid to add a second recurrence or extra parameters.

Don't try to cleverly figure out which option is best. Try them all.

The magic of recursion tells you which is best for a particular situation.

How To Approach Problems

In section, we've made you follow these steps:

1. Read the problem carefully (make sure you know what problem you're actually solving)

2. Make some sample inputs/outputs

3. Set a "baseline."

4. Then try to generate the algorithm.

It's hard to take the time to do these in an exam, but at least make sure you do #1. Solving the wrong problem is not good for test-taking.