

# Discussion 4

Sunday, September 23, 2018

7:23 PM

## Topic: Greedy Algorithms

main idea: build up solution piece by piece

by choosing the next piece that offers the most immediate benefit without thinking about future.

Seems like a pretty worrisome strategy ... but sometimes it gives optimal solution!

## Minimum Spanning Tree

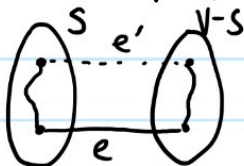
- MST: tree, containing all vertices, minimum total cost
- Greedy algorithm 1: Kruskal's algorithm  
main idea: pick the next cheapest edge that doesn't create a cycle

runtime:  $O(|E| \log |V|)$  ← sorting edges. "Find"

- Greedy algorithm 2: Prim's algorithm  
main idea: grow the subtree by adding the cheapest edge from tree to another vertex

runtime:  $O(|E| \log |V|)$  ← priority queue

- Correctness: cut property:



Any edge of minimal weight in a cut is in some MST.

In general, you could make a similar "exchange argument" to prove the correctness of other greedy algorithms.