CSCI 270 Lecture 9: Shortest Path

Subset Sum

Given positive integers $w_1, w_2, ..., w_n$ and a target W, is there a subset of the integers which add up exactly to W?

Sample instance: integers $= \{2, 5, 7, 13, 16, 17, 23, 39\}, W = 50$

- Is there a subset for the above instance?
- I need to break this up into bite-size decisions. What should my first decision be?
- What information do I need to pass down to the next level of recursion?

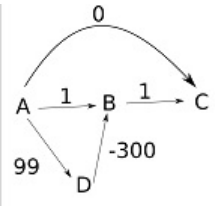
$SS[x, t]$ will store 1 if you can use a subset of the integers $w_x, w_{x+1}, ..., w_n$ to add up exactly to t.

- If I do not include w_x, what recursive call should I make?
- If I do include w_x, what recursive call should I make?
- If one answer returns 1, and the other answer returns 0, then what should my recursive function return?
- Under what base case conditions do I return 1?
- Under what base case conditions do I return 0?

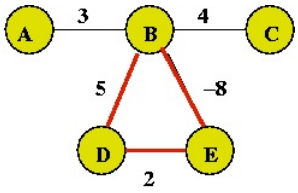
$SS[x, t] = \max(SS[x + 1, t], SS[x + 1, t - w_i])$
$SS[x, 0] = 1$
$SS[n + 1, t] = 0, \forall t \neq 0$

- What order should I fill the array?
- Where is the answer stored?
- What is the runtime of the algorithm?
- Is this a polynomial-running time algorithm?

Shortest Path (Again!)



- What is the length of the shortest path from A to C?
- What will Dijkstra's Algorithm find?
- Why didn't Dijkstra's Algorithm work?



What is the length of the shortest path from A to C?

We will assume there are no negative weight cycles, since this leads to nonsense answers and situations.

We want to find the length of the shortest path from s to t on a graph with no negative weight cycles.

- What should my first decision be?
- What information do I need to pass down to the next level of recursion?

Attempt 1: $SP[x]$ will be the length of the shortest path from x to t.
$SP[x] = \min_{(x,y)\in E} c_{(x,y)} + SP[y]$
$SP[t] = 0$

What order should I fill the array?

Attempt 2: $SP[i, x]$ will be the length of the shortest path from x to t using no more than i edges.
$SP[i, x] = \min_{(x,y)\in E} c_{(x,y)} + SP[i-1, y]$
$SP[i, t] = 0$
$SP[0, x] = \infty, \forall x \neq t$

- What order should I fill in my array?
- Where is the answer stored?
- What is the runtime of this algorithm?

This is known as the Bellman-Ford algorithm.