Rommel Gay

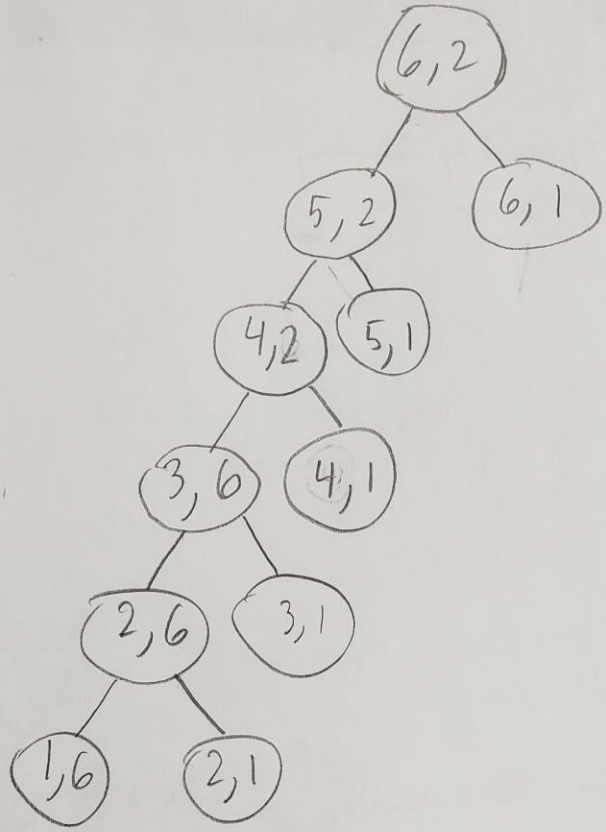
11/28/2019

Professor Yao

1) Run trials and tribulations.

A) Check the speeds before the injury and select the fastest speed.

C)


6,2
5,2
6,1
4,2
5,1
3,6
4,1
2,6
3,1
1,6
2,1

D) There are 11 distinct subproblems.

E) N^M

F) To memoize run trials recursion a 2-D array should be created in order to store the subproblems so that the values can be returned when called. This increases the speed of the recursive function by not having to solve for the values again.

2) Holiday Special

A) The optimal substructure would split the chefs into groups and see which chefs can do the most steps. Then it will move on to see which steps are needed between the chefs the were broken into groups and select them based on the amount of steps they can do.

B) Since the list of steps is already sorted, we just need to check which chefs can do the most steps. Then we get a chef that can do as much of the remaining steps. Finally, we get a chef that can complete the missing steps.

D) N^M

E) Assume that there is a way to have even less chefs switch out of steps. Let Z(a...j) be the optimal choice. Assume you have up to k choices. Then split the path choice into Z(a...k) + Z(k...j). Assume S is the shorter path instead of Z. S(a...k) + S(k...j) If S(a...k) < Z(a...k) then S(a...k) + Z(k...j) < Z(a...k) + Z(k...j) Cannot be true because Z(a...k) + Z(k...j) is optimal therefore it does not exist.

3) League of Patience

A) Create a set of shortest path tree set that keeps track of the shortest path. Assign the distances as infinite and the starting vertex 0 so that the it will start from the 0 vertex. While all the vertices are not in the shortest path tree set, pick a vertex that is not in the set and has a minimum distance value. Add it

to the shortest path tree set. Update the adjacent vertices distance through iteration. If the sum of the distance of value u and weight of edge is u-v is less than v update the distance value of v.

B) O(V^2)

D) Dijsktra's Algorithm

E) The current time complexity is O(V^2). To make "genericShortest" faster you can use the Boruvka's algorithm which is O(ElogV).