
LeagueOfPatience.java

a. I will use the Bellman-Ford Algorithm
b. The Bellman-Ford algorithm would have a time of: O((number of edges)*(number of vertices))
c. The algorithm used in the generic method is Djikstra's shortest path with adjacency matrix representation1
d. Some modifications that can be added would be the consideration of the time until the next quest. So, we would consider not only the duration of the quest, but also the waiting time to start the next quest and perhaps add that on the weight of the edges.
e. The current time complexity is O(v^2). It can be improved to become O(E*logV) by using adjacency list representation instead of matrix representation.
f. Extra Credit is done

Holiday Special

a. The subproblem would be finding the chef who can do the initial step from the remaining steps and then do as many steps as possible afterwards. The steps would have to be performed sequentially.
b. We would start with the initial step and whoever can cook that step and as many steps as possible gets assigned those steps. Afterwards, we assign the initial step to become the next step that is in the queue. And, we again find cooks who can do those steps. We repeat until there are no more steps, or initial step is >= the last step.
c. Coding is done.
d. The algorithm has a complexity of O(number of cooks * number of steps)
e. Let's say we have an optimal list of chefs OPT = {O_1, O_2, …, O_k} and a list of chefs formed by the greedy algorithm ALG = {G_1, G_2, …, G_h} where k, by definition, is less than h. If we replace some O_r, where 1<r<k, with G_r then that way made a better solution than the optimal solution because G_r can do more steps. The optimal solution is no longer optimal by definition.

RunningTrials

a. The structure would have a case of injury or no injury. runTrialsRecur(speed - 1, days - 1),
runTrialsRecur(possibleSpeeds - speed, days)
d. 6*2 = 12 Subproblems
e. N*M = NM Subproblems
f. We would have an array in the recursive method that is referred to in case the same computations is needed