CS2020: Data Structures and Algorithms (Accelerated)

Problems 22–23

Released: Tuesday, April 5th 2011, Due: Wednesday, April 13th 2011, 13:59

Overview. Finally, your last problem set for this semester. The problems are from the recent Singapore National Olympiad in Informatics (NOI) 2011.

Collaboration Policy. As always, you are encouraged to work with other students on solving these problems. However, you **must** write up your solution **by yourself**. In addition, when you write up your solution, you **must** list the names of every collaborator, that is, every other person that you talked to about the problem (even if you only discussed it briefly). Any deviation from this policy will be considered cheating, and will be punished severely, including referral to the NUS Board of Discipline.

Problem 22. TOUR

Visit `http://www.comp.nus.edu.sg/~noi/2011/noi-tasks-2011.pdf` and read the third problem `TOUR`. This problem was set by Steven.

This time, skeleton code is NOT given. Your task is to write a Java file: `Tour.java` that will read in the input file (Singapore map) and then print out one integer (to screen), the maximum satisfaction points achievable given that map.

To facilitate marking, please write clearly (as comments inside Java file) your base case(s) and recurrence(s). Then explain briefly the meaning of your DP formulation. Finally, analyze the space and time complexity of your DP solution.

The official test data is available at `http://www.comp.nus.edu.sg/~noi/2011/TOUR/` and should be 100% correct as it has been verified many times.

Hint: Isn't this problem looks very similar to what we have discussed in Lecture20? (be careful, there is one twist to the original problem, but it is not that hard to solve that twist). Of course, you have to pre-process the map first with something that you have learned in Lecture14+17.

Problem 23. TUTOR

Again, visit `http://www.comp.nus.edu.sg/~noi/2011/noi-tasks-2011.pdf` and read the fourth problem `TUTOR`. This problem was originally set by Dr Golam Ashraf, SoC, adapted to contest setting by Steven, and verified by Steven's younger brother, Felix Halim.

This time, skeleton code is also NOT given. Your task is to write a Java file: `Tutor.java` that will read in the input file (game parameters) and then print out one integer (to screen), the maximum cash that you can gain in that simulation game.

To facilitate marking, please write clearly (as comments inside Java file) your base case(s) and recurrence(s). Then explain briefly the meaning of your DP formulation. Finally, analyze the space and time complexity of your DP solution.

The official test data is available at `http://www.comp.nus.edu.sg/~noi/2011/TUTOR/` and should be 100% correct as it has been verified many times.

Hint: This problem is going to test your understanding of DP technique to near limit. Please spend some time understanding Lecture21 and discuss with your classmates on how to solve this problem. Note that the test data shown above contains different level of difficulties. There are 5 out of 8 test cases that are solvable even with 'naive' DP formulation. All the best :).