Department of Computer Engineering
University of Peradeniya

CO 322 Data Structures and Algorithms

Lab 02 - Dynamic Programming

23^{th} of July 2020

1. Aim

Aim of this laboratory is to get an understanding about the applications of dynamic programming and how it can be achieved by using the primitives provided by a common programming language.

2. Objectives

(a) Find a recursive solution to a problem with overlapping subproblems.
(b) Use memorization to improve the runtime.
(c) Evaluate the runtime of the implementations.

3. Problem

There are N stations on a train route numbered from 0 to N-1. The ticket cost between two stations (say i, j) is given in a global, two dimensional matrix called **cost**. The ticket cost between two stations can be represented by cost[i][j]. Further, the train travels only in one direction from station 0 to N-1.

Your task is to implement a function called **int minCost(int fromStation, int toStation)** using Java which returns the minimum possible cost to travel between given two stations. Note that depending on the station to station ticket price it may even be attractive to break the journey (get down from some intermediate station and restart the journey) while traveling from i to j.

You may assume that there are less than 100 stations. (might be useful when deriving a key for hashing). Use the provided skeleton code.

4. Task

You're required to do the following:

(a) Using recursion (or otherwise) implement the *minCost* function.
(b) What is the runtime complexity of your implementation.
(c) Argue that dynamic programming can be used to improve the runtime.
(d) Use dynamic programming to improve the runtime.
(e) Calculate the runtime of your implementation in part 4 above. Assume, hashing is O(1).

5. Submission

task1.java

Submit the answers to part 1 and 4 in two different java files named 1.java and 4.java. Answers to other questions should be submitted in a written.pdf file. All files should be in a single zip file. Deadline for submission **6th August 2020**.