CS 161 Fall '18 Section 7 / Homework 8

Investing

Suppose you are investing. You want to buy low, then sell high. You have an array A of integers representing future prices, and can make one buy followed by one sell. Regardless of the price at which you buy, you will only purchase a single unit. What is the maximum profit you can make on this investment?

(a) Design an $O(n \log n)$ divide-and-conquer algorithm to return the maximum potential profit, and justify its runtime.

(b) Design an O(n)-time algorithm to solve this problem, and justify its runtime.

Quicksand

We are travelling through a marsh which can be mapped to an $M \times N$ grid. The marsh is mostly solid ground, but some parts are quicksand pits located throughout the marsh that are unsafe to travel through. In addition, the locations adjacent (up / down / left / right) to the quicksand pits are also unsafe. At each timestep, you can travel to any of the locations adjacent to your current location in the marsh (diagonal moves are *not* allowed). Design an algorithm that returns a shortest safe path from one side of the marsh to the other (starting at any location in the leftmost column of the grid and ending at any location in the rightmost column), and analyze its runtime.

Longest Palindromes

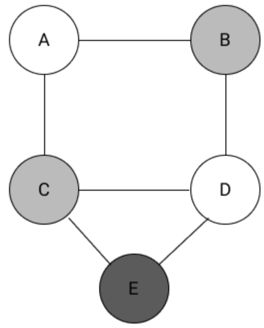
A string is a *palindrome* if it is the same both forwards and backwards. For example, "kayak" is a palindrome, but "canoe" is not. Similarly, "aa" is a palindrome, but "abaa" is not. ("a" is also a palindrome.)

A *subsequence* of a string is any sequence of characters that can be derived from the original string by deleting characters from that string. For example, the subsequences of the string "aid" are "aid", "ai", "ad", 'a", "id", "i", "d", and " ".

Design an algorithm that takes a string and returns the length of the longest subsequence that is a palindrome. Analyze the runtime of your algorithm.

Graph Coloring

Minimum graph coloring is an open NP-hard problem for finding the minimum number of colors needed to color all the nodes in the graph such that no nodes of the same color share an edge. Below is an example of a minimum-color graph.



(a) Although the problem is NP-hard, we can use greedy algorithms to obtain a pretty good solution. Describe a greedy algorithm that never uses more than d+1 colors, where d is the maximum degree of a vertex in the given graph. Your algorithm should run in $O(V^2 + E)$.

(b) Prove by counter example that your greedy algorithm does not always return the correct minimum coloring. Your solution should include a graph, the correct minimum coloring, and the minimum coloring returned by the greedy algorithm.

(c) Prove that the graph will return at most a colorization of $d+1$. (note: You may use proof by induction, but you do not need to for this problem.)

Covering a Number Line

n is the initial position of a person on a number line, and l is the probability of the person of going left. Given n and l, find the probability of reaching all points on the number line after n moves. Each move is either to the right or to the left.

Minimum Spanning Trees

Given a weighted, undirected graph $G = (V, E)$, with edge weights in the set $W = \{1, 2, 3\}$, design an algorithm to find a minimum spanning tree, and analyze its runtime.