CS 170 HW 8

Due on 2018-10-14, at 9:59 pm

1 Study Group

List the names and SIDs of the members in your study group.

2 A Dice Game

Consider the following 2-player game played with a 6-sided die. On your turn, you can decide either to roll the die or to pass. If you roll the die and get a 1, your turn immediately ends and you get 1 point. If you instead get some other number, it gets added to a running total and your turn continues (i.e. you can again decide whether to roll or pass). If you pass, then you get either 1 point or the running total number of points, whichever is larger, and it becomes your opponent's turn. For example, if you roll 3, 4, 1 you get only 1 point, but if you roll 3, 4, 2 and then decide to pass you get 9 points. The first player to get to N points wins, for some positive N.

Alice and Bob are playing the above game. Let $W(x, y, z)$ be the probability that Alice wins given that it is currently Alice's turn, Alice's score (in the bank) is x, Bob's score is y and Alice's running total is z.

(a) Give a recursive formula for the winning probability $W(x, y, z)$.

(b) Based on the recursive formula you gave in the previous part, design an $O(N^3)$ dynamic programming algorithm to compute $W(x, y, z)$. Briefly describe your algorithm, prove its correctness and runtime.

3 Knightmare

Give an algorithm to find the number of ways you can place knights on an N by M ($M < N$) chessboard such that no two knights can attack each other (there can be any number of knights on the board, including zero knights). Clearly describe your algorithm and prove its correctness. The runtime should be $O(2^{3M} \cdot N)$.

4 Triangulating a polygon

You are given a convex polygon, P, of n vertices, $(x_1, y_1), \ldots, (x_n, y_n)$. A triangulation of P is a collection of $n - 3$ diagonals of P such that no two diagonals intersect inside the polygon. A triangulation splits the polygon's interior into $n - 2$ disjoint triangles. The cost of a triangulation is defined to be the sum of the lengths of the diagonals forming the triangulation. Your task is to devise a dynamic programming algorithm to compute the minimum cost triangulation of a given polygon. Please provide a three part solution describing your algorithm, a proof of correctness and a runtime analysis. (*Hint*: First order the points $(x_1, y_1), \ldots, (x_n, y_n)$ in a clock-wise manner and index each sub-problem with a pair of indices $1 \leq i < j \leq n$).

5 Three Partition

Given a list of positive numbers, $a_1, \dots, a_n$, can we partition $\{1, \dots, n\}$ into 3 disjoint subsets, I, J, K such that:

$$\sum_{i \in I} a_i = \sum_{j \in J} a_j = \sum_{k \in K} a_k = \frac{\sum_{i=1}^n a_i}{3}$$

Devise and analyze a dynamic programming solution to the above problem that runs in time polynomial in $\sum_{i=1}^n a_i$ and n.

6 2-SAT

Please provide solutions to parts (d), (e) and (f) of Question 3.28 from `http://algorithmics.lsi.upc.edu/docs/Dasgupta-Papadimitriou-Vazirani.pdf`.