CS 477: Problem Set 2

Section: MW 2-3:50 pm

Total: 150pts Due: 04/29/2020

Instructions:
1. I leave plenty of space on each page for your computation. If you need more sheet, please attach your work right behind the corresponding problem. If your answer is incorrect but you show the computation process, then partial credits will be given. Please staple your solution and use the space wisely.

2. You are allowed to work on the homework in a group (you can stick to your presentation group). No late assignment is accepted. Identical solutions (same wording, paragraph, code), turned in by different groups (persons), will be considered cheating.

3. Full credit will be given only to the correct solution which is described clearly. Convoluted and obtuse descriptions might receive low marks, even when they are correct. Also, aim for concise solutions, as it will save you time spent on write-ups, and also help you conceptualize the key idea of the problem.

4. CLRS refers to the Introduction to Algorithms (3rd edition) textbook while SW refers to the Algorithms (4th edition) textbook specified in the syllabus.

First Name:

Last Name:

Group ID:

Score: **/ 150**

Problem 1 DFS

Give a counterexample to the conjecture that if a directed graph G contains a path from u to v and if $u.d < v.d$ in a DFS of G, then v is a descendant of u in the dpth-first forest produced.

Problem 2 BFS

What is the running time of BFS if we represent its input graph by an adjacency matrix and modify the algorithm to handle this form of input?

Problem 3 MST

Let e be a maximum-weight edge on some cycle of connected graph $G = (V, E)$. Prove that there is a minimum spanning tree of $G‘ = (V, E - \{e\})$ that is also aminimum spanning tree of G. That is, there is a minimum spanning tree of G thatdoes not include e.

Problem 4 Kruskal

Suppose that all edge weights in a graph are integers in the range from 1 to $||V|$. How fast can you make Kruskal's algorithm run? What if the edge weights are integers in the range from 1 to W for some constant W ?

Problem 5 Dijkstra's

Run Dijkstra's algorithm on the directed graph of Figure 24.2 on CLRS, first using vertex s as the source and then using verte z as the source. In the style of Figure 24.6 on CLRS, show the d and π values and the vertices in set S after each iteration of the while loop.

Problem 6 MaxFlow-MinCut: Ford-Fulkerson

Show how to find a maximum flow in a network $G = (V, E)$ by a sequence of at most $|E|$ augmenting paths. (Hint: Determine the paths after finding the maximum flow.)

Problem 7 Maxflow: Escape problem: 20pts

Please do (a) and (b) of problem 26-1 on CLRS (p.760 - 761)

Problem 8 Linear Programming:20pts

(a) Show that the following linear program is infeasible:
maximize: $3x_1 - 2x_2$
subject to (1)$x_1 + x_2 \leq 2$ (2) $-2x_1 - 2x_2 \leq -10$ (3)$x_1, x_2 \geq 0$

(b) convert the following into slack form:
maximize $2x_1 - 6x_3$
subject to

$$x_1 + x_2 - x_3 \leq 7, 3x_1 - x_2 \geq 8$$

$$-x_1 + 2x_2 + 2x_2 \geq 0, x_1, x_2, x_2 \geq 0$$

What are the basic and nonbasic variables?

Problem 9 MaxFlow-MinCut: Ford-Fulkerson

Show how to find a maximum flow in a network $G = (V, E)$ by a sequence of at most $|E|$ augmenting paths. (Hint: Determine the paths after finding the maximum flow.)

Problem 10 String Matching: KMP

Compute the prefix function π for the pattern p. In this case we have p is exactly the same as our text T. We know that text $T = ababbabbabbababbabb$

Problem 11 String Matching: Repetition factors based: 30pts

Let y^i denote the concatenation of string y with itself i times. We say that a string $x \in \sum^*$ has repetition factor r if $x = y^r$ for some string $y \in \sum^*$ and some $r > 0$. Let $\rho(x)$ denote the largest r such that x has a repetition factor r.
(a) Give an effiicent algorithm that takes as input a patter $P[1 \cdots m]$ and compute the value $\rho(P_i)$ where $1 \leq i \leq m$. What is the running of your algorithm?

(b) For any pattern $P[1, ..., m]$, let $\rho^*(P)$ be $max_{1 \leq i \leq m} \rho(P_i)$. Prove that if the pattern P is chosen randomly from the set of all binary strings of length m, then the expected value of $\rho^*(P)$ is O(1).

c. Argue that the following string-matching algorithm on p. 1013 on CLRS correctly finds all occurrences of pattern P in a text $T[1, ..., n]$ in time $O(\rho^*(P)n + m)$