CMSC 451:Fall 2017 Dave Mount

Practice Problems for the Final Exam

The final will be **Tue, Dec 19, 10:30-12:30am** in our usual classroom. The exam will be closed-book and closed-notes, but you will be allowed two sheets of notes (front and back of each sheet).
Disclaimer: These are practice problems, which have been taken from old homeworks and exams. They do not necessarily reflect the actual length, difficulty, or coverage for the exam. For example, we have covered some topics this year that were not covered in previous semesters. So, just because a topic is not covered here, do not assume it will not be on the exam.

Problem 0. You should expect one problem in which you will be asked to work an example of one of the algorithms or NP-complete reductions that we have presented in class.

Problem 1. Short answer questions.

(a) Suppose that you perform a DFS on an undirected graph $G = (V, E)$, and for each vertex $u \in V$, you compute the discovery time $d[u]$ and finish time $f[u]$. Let u be a descendant of v in the DFS tree. What can be said about the relative order of $d[u]$, $f[u]$, $d[v]$, and $f[v]$?

(b) What is the longest common subsequence of the strings $X = \langle ababa \rangle$ and $Y = \langle babab \rangle$? (If there are multiple, list any one. I don't need to see a trace of the algorithm, just the final answer.)

(c) Give a definition of the k-center problem. (What is the input and what is the output?) What does it mean to say that an algorithm yields a factor-2 approximation to this problem?

(d) Given a flow network G, let (X, Y) denote the minimum capacity cut in G.

  (i) True or False: If we double the capacities of all the edges (x, y) that cross the cut, then the value of the maximum flow doubles.

  (ii) True or False: If we reduce by half the capacities of all the edges (x, y) that cross the cut, then the value of the maximum flow reduces by half.

(e) True or False: Suppose that the capacities of the edges of an s-t network are integers that are all evenly divisible by 3. (E.g., 3, 6, 9, 12, etc.) Then there exists a maximum flow, such that the flow on each edge is also evenly divisible by 3.

(f) True or False: It is possible to determine in polynomial time whether a graph G has an independent set of size 100.

Problem 2. Recall that in the longest common subsequence (LCS) problem the input consists of two strings $X = \langle x_1, \ldots x_m \rangle$ and $Y = \langle y_1, \ldots, y_n \rangle$ and the objective is to compute the longest string that is a subsequence of both X and Y.

(a) (LCS with wild cards) Each of the strings X and Y may contain a special character "?", which is allowed to match *any single character* of the other string, *except* another wild-card character (see Fig 1(a)).

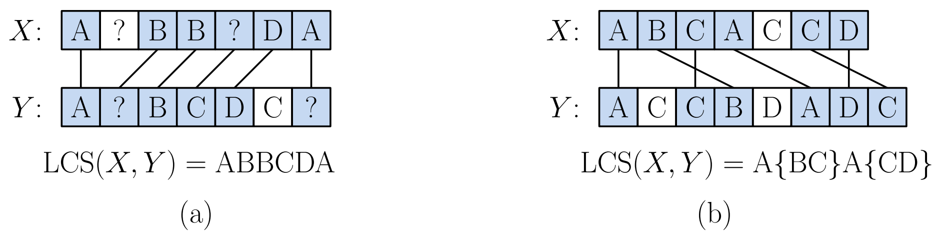

Figure 1: LCS variants.

(b) (LCS with swaps) Any two consecutive characters of either string are allowed to be swapped before matching in the LCS (see Fig 1(b)).

In all cases, your revised rule should admit an $O(mn)$ time solution.

Problem 3. Your local pharmacy has asked you to help set up the work schedule for the next month. There are n pharmacists on the staff and m days in the month. Each pharmacist gives a list of the days of the month that he/she is available to work. For the ith pharmacist, this is given as a list A_i, where each number in the list is in the range from 1 to m. For example, if $A_i = \langle 3, 5, 9, 15, 23 \rangle$, then the ith pharmacist is available to work on days 3, 5, 9, 15, and 23 of the month. Let $d_i = |A_i|$ denote the number of days that pharmacist i is available to work. Then he/she should be scheduled to work at least $\lceil d_i/2 \rceil$ of these days. Each day there must be exactly 3 pharmacists on duty. (An example is shown in the figure below. There are 5 pharmacists and 4 days in the month.)

Availability Lists
$A_1 = \langle 1, 2, 4 \rangle$
$A_2 = \langle 1, 2, 3 \rangle$
$A_3 = \langle 1, 2, 3, 4 \rangle$
$A_4 = \langle 2, 4 \rangle$
$A_5 = \langle 2, 3, 4 \rangle$

(Possible) Final Schedule:

Day	Pharmacists working
1	1, 2, 3
2	1, 4, 5
3	2, 3, 5
4	3, 4, 5

Figure 2: Pharmacy problem.

Present an efficient algorithm that is given the values of n, m, and the availability lists $A_1, \ldots, A_n$, and determines whether there exists a schedule that satisfies all the above requirements. (Hint: Reduce to either Max-Flow or Circulation. You may give a figure for the above example, but your description should work for any valid input.) Present a *brief* proof that your algorithm is correct.

Problem 4. You are given a collection of n points $U = \{u_1, u_2, \ldots, u_n\}$ in the plane, each of which is the location of a cell-phone user. You are also given the locations of m cell-phone towers, $C = \{c_1, c_2, \ldots, c_m\}$. A cell-phone user can connect to a tower if it is within distance Δ of the tower. For the sake of fault-tolerance each cell-phone user must be connected to at least three different towers. For each tower c_i you are given the maximum number of users, m_i, that can connect to this tower.

Give a polynomial time algorithm, which determines whether it is possible to assign all the cell-phone users to towers, subject to these constraints. Prove its correctness. (You may assume you have a function that returns the distance between any two points in $O(1)$ time.)

Problem 5. You are given a directed network $G = (V, E)$ with a root node r and a set of terminal nodes $T = \{t_1, \ldots, t_k\}$. Present a polynomial time algorithm to determine the minimum number of edges to remove so that there is no path from r to any of the terminals (see Fig 3). (Hint: Use network flow.) Prove that your algorithm is correct.

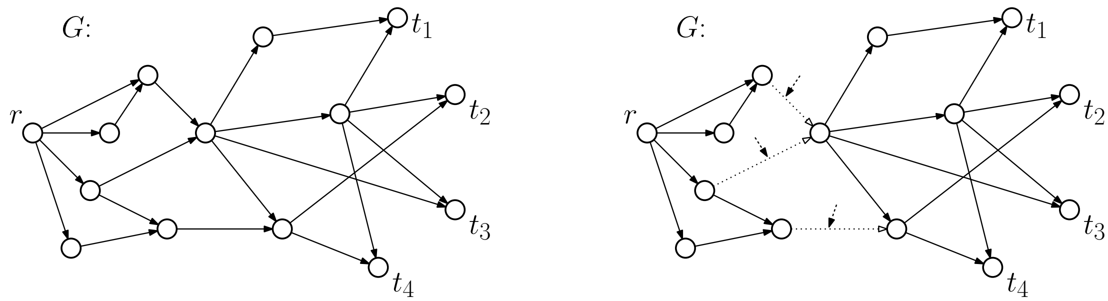

Figure 3: Eliminating edges to separated r from terminals.

Problem 6. In the High-Degree Independent Set (HDIS) problem, you are given a graph $G = (V, E)$ and an integer k, and you want to know whether there exists an independent set V' in G of size k such that each vertex of V' is of degree at least k. (For example, the graph in Fig. 4 has an HDIS for $k = 3$, shown as the shaded vertices. Note that it does *not* have an HDIS for $k = 4$. Although adding the topmost vertex would still yield an independent set, this vertex does not have degree at least four.)

(a) Show that HDIS is in NP.

(b) Show that HDIS is NP-hard. (Hint: Use standard independent set (IS).)

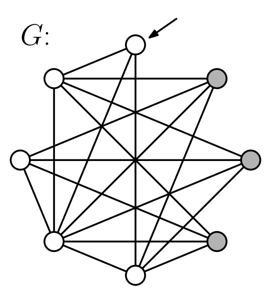

Figure 4: High-degree independent set for $k = 3$.

Problem 7. Show that the following problem is NP-complete.

Balanced 3-coloring (B3C): Given a graph $G = (V, E)$, where $|V|$ is a multiple of 3, can G can be 3-colored such that the sizes of the 3 color groups are all equal to $|V|/3$. That is, can we assign an integer from $\{1, 2, 3\}$ to each vertex of G such that no two adjacent vertices have the same color, and such that all the colors are used equally often.

Hint: Reduction from the standard 3-coloring problem (3COL).

Problem 8. Given an undirected graph $G = (V, E)$, a *Hamiltonian path* is a simple path (not a cycle) that visits every vertex in the graph. (The graph shown in Fig. 5 has a Hamiltonian path.) The *Hamiltonian Path problem* (HP) is the problem of determining whether a given graph has a Hamiltonian path.

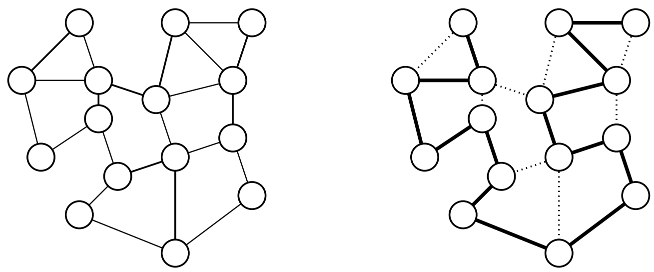

Figure 5: Hamiltonian Path.

(a) Show that HP is in NP.

(b) Professor Farnsworth observes that if a graph has a Hamiltonian Cycle, then it also has a Hamiltonian Path. He suggests the following trivial reduction in order to prove that HP is NP-hard. Given a graph G for the Hamiltonian Cycle problem, simply output a copy of this graph. Explain why Professor Farnsworth's reduction is *incorrect*.

(c) Give a (correct) proof that HP is NP-hard. (Hint: The reduction is from the Hamiltonian Cycle problem, HC.)