CS 170 (Clancy) Exam 2
November 9, 2000

Read and fill in this page now.
Do NOT turn the page until you are told to do so.

Your name: ____________________

Your login name on the EECS instructional computers: ____________________

Your discussion section day and time: ____________________

Your discussion t.a.: ____________________

Name of the person sitting on your left: ____________________

Name of the person sitting on your right: ____________________

This is an open-book test. You have approximately 80 minutes to complete it. You may consult any books, notes, or other paper-based inanimate objects available to you. To avoid confusion, read the problems carefully. If you find it hard to understand a problem, ask us to explain it. If you have a question during the test, please come to the front or the side of the room to ask it.

This exam comprises 15% of the points on which your final grade will be based. Partial credit may be given for wrong answers. Your exam should contain six problems (numbered 0 through 5) on eight pages. Please write you answers in the spaces provided in the test; in particular, we will not grade anything on the bacl of an exam page unless we are clearly told on the front of the page to look there.

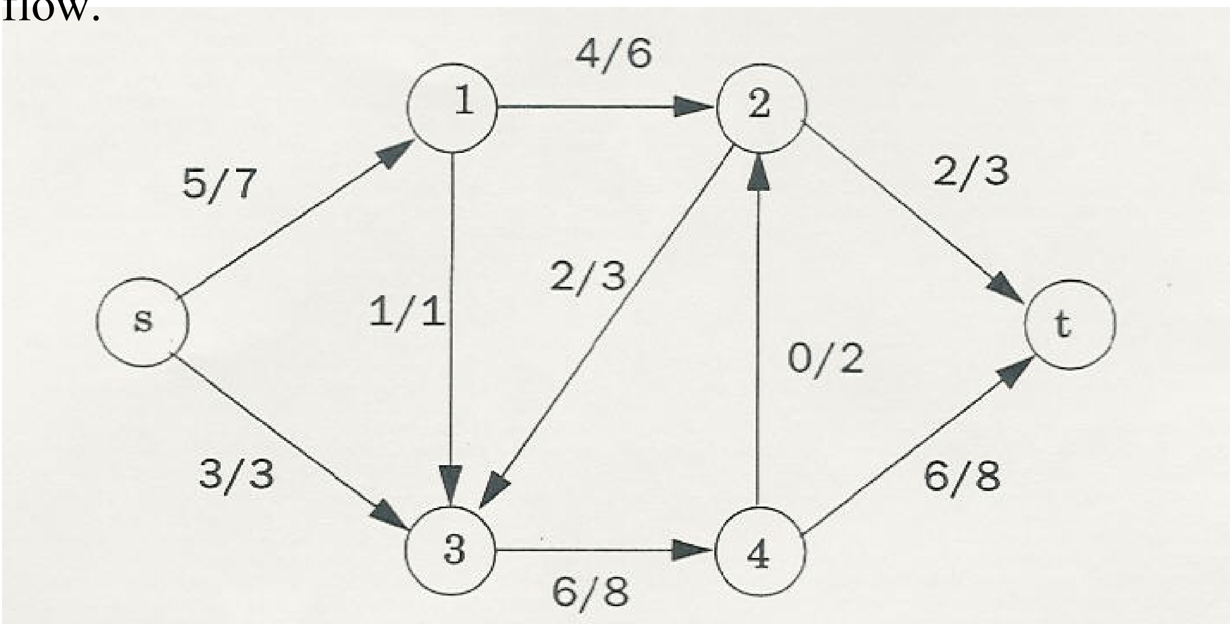
Relax- this exam is not worth having heart failure about.

Problem 0 (1 Point, 1 minute)

Put your name on each page. Also make your sure you have provided the information requested on the first page.

Problem 1 (6 Points, 10 minutes)

Consider the following flow network and indicated flow.

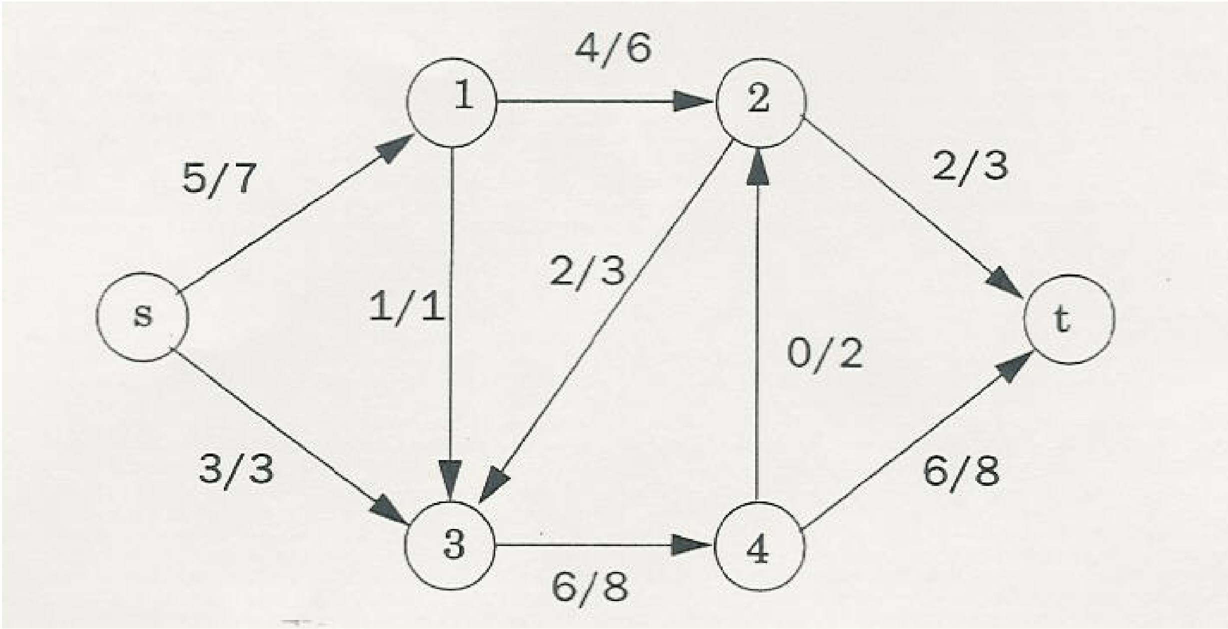


Part a

Display the corresponding residual graph.

Part b

The network's flow of 8 is not as large as possible. In the diagram below, change the network by reducing the capacity of a single edge in such a way that this flow of 8 is optimal in the modified network. Indicate clearly which edge you change and what its new capacity is.



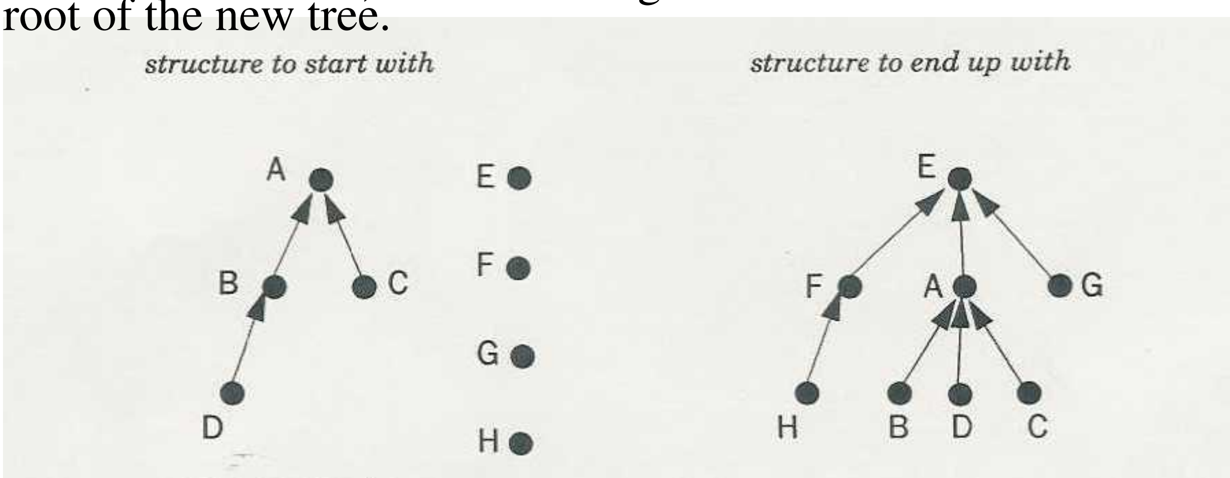
Part c

Find a cut of capacity 8 in the modified network.

Problem 2 (5 Points, 12 minutes)

Consider the disjoint-set union- find structure pictured on the left below. Give a sequence of calls to union that produces the structure pictured on the right below.

Assumce the four-element set on the left has rank 2. Also assume that the code on pages 448 and 449 in CLR is used, implementing both union-by-rank and path compression; note that when two trees of equal rank are combined, the second argument contains the root of the new tree.



Problem 3 (6 Points, 15 minutes)

Consider the following linear programming problem that corresponds to finding a minimum vertex cover in a graph $G = (V, E)$. It has a variable x_k for each vertex v_k in V and a constraint $x_u + x_v >= 1$for each edge (u, v) in E. The objective function to minimize is $x_1 + ... + x_n$

One can prove two properties about this problem.
- There is an optimum solution $(x_1, x_2, ... , x_n)$ for which x_k in $\{0, 1, 0.5\}$ for all $k = 1, ... , n$
- There is an efficient algorithm that, given G, produces the optimum solution just mentioned.

Part a

Use both these properties to derive an efficient approximation algorithm for the minimum vertex cover problem, and argue that your algorithm indeed produces a vertex cover.

You will receive no credit for reproducing the approximate vertex cover algorithm described in class and in CLR section 37.1. Your algorithm should run in time $O(|V|)$ once the linear programming problem has been solved

Part b

Determine a ratio bound for the algorithm from part a, and briefly indicate how you determined it.

Problem 4 (6 points, 20 minutes)

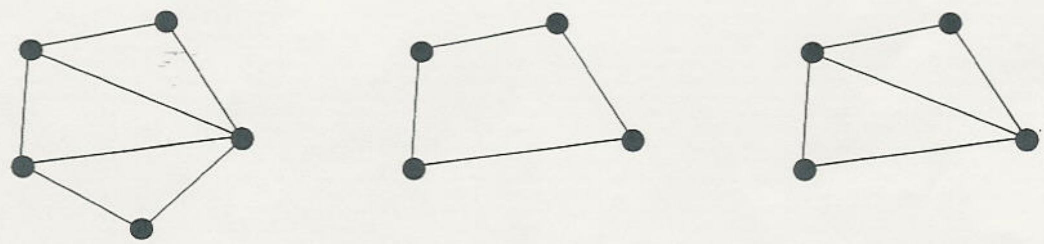
The maximum common induced subgraph problem is stated as follows: given two graphs G_1 and G_2, find the size of the largest graph (the graph with the largest number of vertices) that's an *induced* subgraph of both G_1 and G_2. Equivalently, this is to find the largest k such that there is a way to delete all but k vertices from G_1 and G_2 so that two isomorphic graphs result.

Recall (from CLR page 88) the definitions of subgraph and induced subgraph:

- G' = (V', E') is a *subgraph* of G = (V, E) if V' ⊆ V and E' ⊆ E.
- Given a set V' ⊆ V, the subgraph of G *induced* by V' is the graph G' = (V', E'), where E' = {(u,v) ∈ E: u, v ∈ V'}.

Examples:

graph — *a subgraph but not an induced subgraph* — *an induced subgraph*



Part a

State the decision problem that corresponds to the maximum common induced subgraph problem.

Part b

Show that your answer to part a is NP-complete. You may use any of the problems proved NP-complete in CLR, in Manber, or on the homework to do this.

Problem 5 (6 points, 22 minutes)
Suppose we are given a list [X_1, X_2,...., Xn] of points on the real number line, arranged in increasing order, and we are to determine the smallest set of unit-length closed intervals that together contain all of the given points. A greedy algorithm that does this is the following:

Initialize the set S of intervals to { }.
While the list of points isn't emty, do the following:
Choose the first point in the list; call it x.
Add the interval [x, x+1] to S.
Remove all points in this interval from the list of points.

Prove that this algorithm correctly produces the smallest set of intervals that collectively contain all the points in the original list.