Massachusetts Institute of Technology
6.042J/18.062J: Mathematics for Computer Science
Professor David R. Karger

Handout 22
April 3, 1997

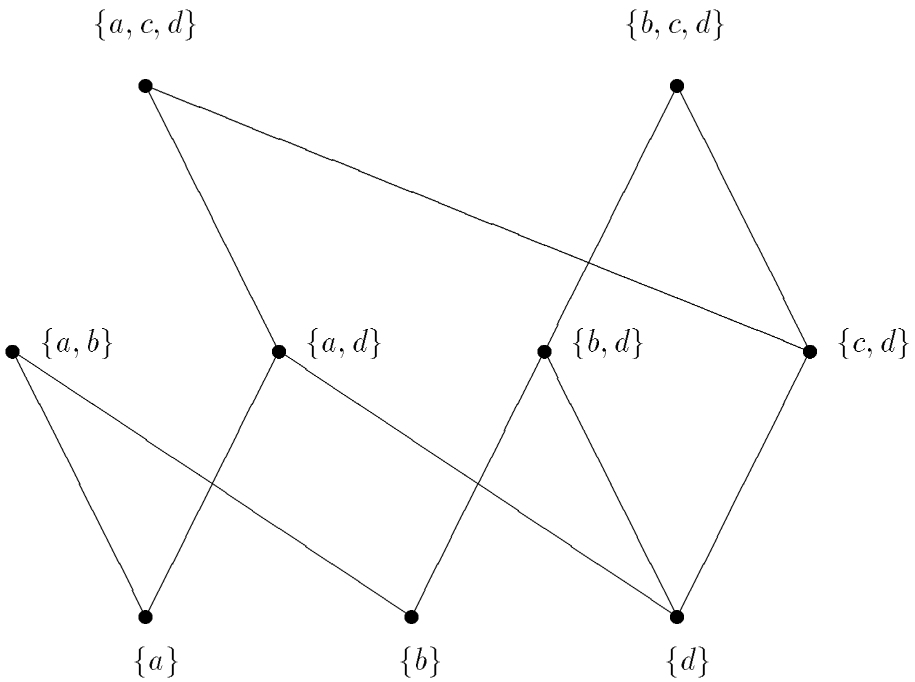
Problem Set 8

Due: April 10, 1997.

Reading: Rosen 7.1-7.4 (skip isomorphism), 7.8, 8.1

Problem 1. Answer the following questions concerning the poset consisting of the elements {{a},{b},{d},{a,b},{a,d},{b,d},{c,d},{a,c,d},{b,c,d}} under $\subseteq$.

(a) Draw the Hasse diagram for this poset.



(b) Find the maximal elements
{a,b}, {a,c,d}, and {b,c,d}.

(c) Find the minimal elements
{a}, {b}, and {d}.

(d) Is there a greatest element?
No.

(e) Is there a least element?
No.

(f) Find all upper bounds of {{b},{d}}.
{b,d} and {b,c,d}.

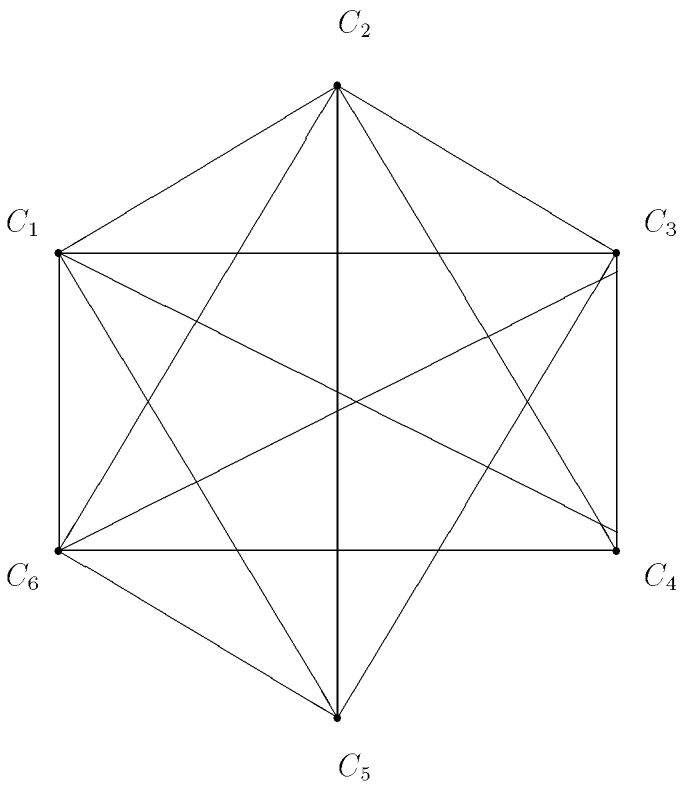
(g) Find the least upper bound of {{b},{d}} if it exists.
{b,d}.

(h) Find all lower bounds of {{a,c,d},{b,c,d}}.
{c,d} and {d}.

(i) Find the greatest lower bound of {{a,c,d},{b,c,d}} if it exists.
{c,d}.

Problem 2. Each of the following sets of people needs to hold a meeting. Draw a conflict graph for these meetings, and schedule them into the smallest possible number of time slots without sending someone to two meetings at the same time.

$$\begin{aligned}
C_1 &= \{\text{Karger}, \text{Sussman}, \text{Abelson}\} \\
C_2 &= \{\text{Sussman}, \text{Ward}, \text{Minsky}\} \\
C_3 &= \{\text{Karger}, \text{Minsky}, \text{Abelson}\} \\
C_4 &= \{\text{Ward}, \text{Minsky}, \text{Abelson}\} \\
C_5 &= \{\text{Karger}, \text{Sussman}\} \\
C_6 &= \{\text{Sussman}, \text{Minsky}, \text{Abelson}\}
\end{aligned}$$



The conflict graph is like the complete graph on these vertices, except it has no edge between C_4 and C_5. The number of slots required for scheduling is the number of colors required for a valid graph coloring of the conflict graph. Every pair of vertices in the set $\{C_1, C_2, C_3, C_4, C_6\}$ are connected by an edge in the conflict graph. So no two of them can be colored the same color. Thus coloring the conflict graph requires at least 5 colors. The graph can, in fact, be colored with 5 colors by giving each vertex in the above set one of the colors and assigning C_5 the same color as C_4. Thus the only two vertices that are identically colored are C_4 and C_5, and they are not adjacent in the conflict graph. This corresponds to scheduling C_1, C_2, C_3, and C_6 in time slots by themselves and scheduling C_4 and C_5 in a common time slot.

Problem 3. Give the adjacency matrix of the graph W_4. How many (simple or not) paths of length 2, 3 and 4 are there between the center vertex and each other vertex?

Suppose we number the center vertex 1 and the remaining vertices consecutively around their common cycle. Let A be the resulting adjacency matrix. Then

0 1 1 1 1

$$A = \begin{array}{ccccc} 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 \end{array}$$

$$A^2 = \begin{array}{ccccc} 4 & 2 & 2 & 2 & 2 \\ 2 & 3 & 1 & 3 & 1 \\ 2 & 1 & 3 & 1 & 3 \\ 2 & 3 & 1 & 3 & 1 \\ 2 & 1 & 3 & 1 & 3 \end{array}$$

$$A^3 = \begin{array}{ccccc} 8 & 8 & 8 & 8 & 8 \\ 8 & 4 & 8 & 4 & 8 \\ 8 & 8 & 4 & 8 & 4 \\ 8 & 4 & 8 & 4 & 8 \\ 8 & 8 & 4 & 8 & 4 \end{array}$$

$$A^4 = \begin{array}{ccccc} 32 & 24 & 24 & 24 & 24 \\ 24 & 24 & 16 & 24 & 16 \\ 24 & 16 & 24 & 16 & 24 \\ 24 & 24 & 16 & 24 & 16 \\ 24 & 16 & 24 & 16 & 24 \end{array}$$

As discussed in class, the number of paths of length exactly k from vertex i to vertex j is the entry in row i and column j of A^k. Thus the number of paths of length 2 between the center vertex and each other vertex are found in the first column (in the rows corresponding to these other vertices) of A^2. The number of paths of length 3 and 4 are found in the same positions of A^3 and A^4, respectively.

Problem 4. Prove that any tree with a degree k vertex has at least k leaves.

Proof 1:

We use strong induction on k. For $k = 0$, the statement is vacuous. For $k = 1$, the tree must be non-trivial, so it must have a leaf, as proven in class.

Inductive Step: Consider a tree T with a degree $k > 1$ vertex v and let $u_1, \ldots, u_k$ be its neighbors.

If any of the u_i is a leaf then deleting u_i still gives a tree (as shown in class). In this tree, v has degree $k-1$. By the IH, the new tree has at least $k-1$ leaves. These leaves must also be leaves of T. Also considering u_i, T must then have at least k leaves.

If none of the u_i is a leaf then all u_i have degree at least 2. Removing v from T we obtain a forest of k trees, each one containing a node u_i. In each tree u_i has degree at least one, and by IH the tree has at least one leaf w_i. Since each w_i is also a leaf of the original tree, T has at least k leaves.

Proof 2:

Let T be a tree containing a vertex v of degree k. For each edge e incident to v, let P_e be a maximal simple path containing e. Define a function f from the edges incident to v to the leaves of T by letting $f(e)$ be the end of P_e that you reach by starting at v, following e, and continuing along P_e. So $f(e)$ is connected to v in T by a simple path containing e. Clearly $f(e)$ cannot be connected to v in T by a simple path containing any other edge e' incident to v, since this would create a cycle. This verifies that f is an injective map from the edges incident to v to the vertices of T. It was shown in class that the end of a maximal simple path in a tree is a leaf of that tree, so f does indeed map to the leaves of T. Since f is an injection, the number of leaves of T is at least the number of edges incident to v, which is k.

Problem 5. What is wrong with the following "theorem" and "proof?"

False theorem: If every vertex in a graph has degree at least one then the graph is connected.
False proof: We prove the result by induction on the number of vertices. Let $P(n)$ be "if every vertex in a graph with n vertices has degree at least one then the graph is connected."
Base case. $P(2)$ is true since the only graph satisfying the assumption is that with two vertices and one edge between them which is easily seen to be connected.
Inductive step. Assume $P(n)$ is true. Now, take any graph with n vertices with minimum degree at least one and add a new vertex to it. By inductive hypothesis the graph on n vertices is connected. Since the new vertex satisfies the assumption it must have degree at least one, so it must have an edge connecting it to one of the remaining n vertices. Thus the whole graph is connected. Hence $P(n+1)$ is true. Thus, $P(n)$ is true for all n.

The proof has buildup error. In the inductive step instead of considering a graph with $n+1$ vertices, a vertex is added to a pre-existing graph of n vertices. It is not true however, that any n-subgraph of a graph with $n+1$ vertices of degree at least one, has also vertices of degree at least one. Consider for example the graph of 4 vertices consisting of two disjoint line segments.

The next problems are going to prove an interesting fact about graphs that that can be colored with two colors. These graphs are called *bipartite* because you can divide them into two groups (colors) such that every edge has one endpoint in each group. We prove:

Theorem 0.1 *A graph is bipartite if and only if it contains no cycles of odd length.*

It is enough to prove this theorem for a connected graph, because then we can apply it separately to each component of an arbitrary graph.

Problem 6. Prove that a graph contains an odd-length circuit if and only if it contains an odd length cycle (hint: consider the shortest odd length circuit).

If a graph contains an odd-length cycle, then obviously it contains an odd-length ciruit (the cycle itself).

Now suppose that the graph contains an odd-length circuit, and consider an odd-length circuit of minimum length (say n). If it is a cycle then we are done. Suppose, to arrive at a contradiction, that the circuit is not a cycle. Then some vertex $\bar{i}$ (other than the initial) is repeated. The circuit has the form

$$(i_1, i_2), \ldots, (i_k, \bar{i}), \ldots, (\bar{i}, i_l), \ldots, (x_n, x_1)$$

Consider the circuits $(i_1, i_2), \ldots, (i_k, \bar{i}), (\bar{i}, i_l), \ldots, (x_n, x_1)$ and $(\bar{i}, i_{k+1}), \ldots, (i_{l-1}, \bar{i})$.

Their lengths are less than n, and since the sum of their lengths is odd, one of them has odd length. So we have a smaller circuit of odd length and we obtain a contradiction.

Problem 7. Prove that if G is bipartite then it contains no circuits of odd length.

Let G be bipartite and suppose that it has a circuit of length $2n+1$. Then (by pigeonhole) there are $n+1$ nodes of the same colour, so two of them must be adjacent. Thus we obtain a contradiction.

Problem 8. Suppose G is a connected graph with no odd cycles (or circuits). Fix a particular vertex u.

(a) Prove that for any vertex v, the length of every path from u to v is odd or the length of every path from u to v is even.

Let $p1$ and $p2$ be two paths connecting u and v. If one has odd length and the other even, than the concatinated path is an odd cycle. This contradicts the hypothesis.

(b) Suppose I color a vertex accoring to the odd/evenness of its paths from u. Prove this is a legal coloring. Deduce that G is bipartite.

Suppose v has even color. Then there is an odd length path to each of its neighbors, so they have odd color. Analogously, if v has odd color, all its neighbors must have even color because they can be reached from u via an even length path.

Optional Problem 9. Suppose we are in a connected maze (undirected graph) and explore it according to the following rules:

1. Never traverse an edge more than once in each direction
2. The first time we enter a room, mark our entering edge. Do not use the entering edge to exit a room unless it is the only remaining choice under the first rule.

Prove that we get stuck (can't apply either rule) only at our starting point, and that by then every edge has been traversed in both directions.

First we show that we can only get stuck at the origin. For any vertex i, let $index(i)$ denote the number of unused incoming edges minus the number of unused outgoing edges. We claim the following invariants.

For any node i that is not the starting point,

$$index(i) = \{ \begin{array}{cc} -1 & \text{if we are in } i \\ 0 & \text{otherwise} \end{array}$$

while for the starting point o we have

$$index(o) = \{ \begin{array}{cc} 0 & \text{if we are in } o \\ 1 & \text{otherwise} \end{array}$$

These invariants are easily proved by induction on the number of traversed edges.

Base case: if no edge has been traversed yet, then we are in o and $index(n) = 0$ for all nodes because all edges have been used neither to enter nor to exit any given node.

Inductive step: assume that after n edges have been traversed the invariant hold true. We traverse one more edge (v, w). $index(v)$ is incremented by one, and $index(w)$ is decremented by one. A simple case analysis show that if the invariant is true before (v, w) is traversed, then it is true also after traversing that edge.

We can only get stuck in a node v only if all edges of this node have been traversed in the outgoing direction. But in this case $index(n) \leq 0$. This is possible only if n is the starting node, since otherwise $index(n) = -1$. This proves that we can only get stuck at o.

Suppose, to obtain a contradiction, that (i, j) is an edge that has not been traversed in both directions. Then node i must also have a free incomming edge (because it's index is 0). We can assume without loss of generality that this edge is marked (becasue the marked edge is the last one to be used as an exit). Since a marked edge indicates a first visit, and we started at o, there is a path of incomming edges from i to o. So o has an outgoing edge, which contradict either the fact that it's index is 0, or the assumption that we are stuck.