**** SOLUTIONS ****

Computer Science 202
Homework #8, due in class Thursday, Nov. 15, 2018

SUBMISSION INSTRUCTIONS. Please turn in your homework in THREE parts, each part stapled together, with your name, the homework number and the problem numbers on the front of each part. The three parts will be placed in three separate labeled boxes:

- Problems 1 and 2
- Problems 3 and 4
- Problems 5 and 6

Please state (with problem 1) the names of any persons and references to any resources (including online) you consulted with in connection with this assignment, or state that none were consulted. (There will be 2 points off for failure to do so.)

Graphs and trees in these problems are assumed to have at least one vertex. The degree sequence of a simple undirected graph is the sequence of degrees of its vertices, sorted into non-increasing order. For example, if \( G = (V, E) \) is the simple undirected graph with vertices

\[
V = \{1, 2, 3, 4, 5, 6\}
\]

and edges

\[
E = \{\{1, 2\}, \{2, 3\}, \{1, 3\}, \{2, 4\}, \{4, 5\}\}
\]

then the degree sequence of \( G \) is

\( (3, 2, 2, 2, 1, 0) \).

To construct proofs requested below, you may use definitions and results about graphs and trees given in Chapter 10 of “Notes on Discrete Mathematics” and things you prove yourself, but not facts about graphs and trees from elsewhere. Please cite by number the Lemmas and Theorems you use from Chapter 10. General hints: What lemma relates the degrees of vertices to the number of edges in a graph? What useful theorems about trees are in Chapter 10? What is the definition of a spanning tree? All the graphs and trees in this problem set are assumed to have at least one vertex.

1. (15 points) For each of the following sequences, determine whether it is the degree sequence of some simple undirected graph with 6 vertices. If it is, draw one such graph; if it isn’t, prove that it is not.

(a) \( (2,2,1,1,1) \) Solution. There is no graph with this degree sequence. The sum of the numbers in the sequence is 9, but by the Handshaking Lemma (Lemma 10.10.3), the sum of the degrees in a graph is twice the number of edges, which means it must be an even number.
(b) \((2,2,1,1,1,1)\)

Solution. One such graph is:

\[
\begin{array}{c}
\begin{array}{c}
\end{array}
\end{array}
\]

(c) \((5,3,3,2,2,1)\)

Solution. One such graph is:

\[
\begin{array}{c}
\begin{array}{c}
\end{array}
\end{array}
\]

(d) \((3,3,3,2,2)\)

Solution. One such graph is:

\[
\begin{array}{c}
\begin{array}{c}
\end{array}
\end{array}
\]

(e) \((5,4,3,3,1,0)\)

Solution. There is no such graph. If there were, it would have a vertex of degree 5, which must be adjacent to every other vertex, and a vertex of degree 0, which must not be adjacent to any other vertex, which is a contradiction.

2. (15 points) For each of the following sequences, determine (i) whether or not it is the degree sequence of some tree, AND (ii) whether or not it is the degree sequence of some non-tree. If there is a tree, draw one such; if not, prove there is no such tree. And if there is a non-tree, draw one such; if not, prove there is no such non-tree. There may be one or the other, or both, or neither.

(a) \((3,3,2,2,1,1)\)

Solution.

(i) This is not the degree sequence of any tree. If \(G\) is a graph with this degree sequence, then the sum of its degrees is 12, which by the Handshaking Lemma (10.10.3) implies that \(G\) has 6 edges. However, \(G\) has 6 vertices, and we know that the number of edges in any tree is one less than the number of vertices (Theorem 10.10.8), so \(G\) cannot be a tree.

(ii) One such non-tree is:

\[
\begin{array}{c}
\begin{array}{c}
\end{array}
\end{array}
\]

(b) \((3,3,2,1,1,1,1)\)

Solution.

(i) One such tree is:

\[
\begin{array}{c}
\begin{array}{c}
\end{array}
\end{array}
\]

(ii) One such non-tree is:

\[
\begin{array}{c}
\begin{array}{c}
\end{array}
\end{array}
\]
(c) (4,3,2,1,1,1,1)

Solution.

(i) One such tree is:

(ii) One such non-tree is:

3. (20 points) We define a set $\mathcal{T}$ of graphs inductively as follows. If $u$ is a natural number then $G = (\{u\}, \emptyset)$ is an element of $\mathcal{T}$. If $G = (V, E)$ is any element of $\mathcal{T}$ and $u \in V$ and $v$ is a natural number not in $V$, then $H = (V \cup \{v\}, E \cup \{uv\})$ is an element of $\mathcal{T}$.

(a) (10 points) Prove by structural induction on $\mathcal{T}$ that every graph in $\mathcal{T}$ is a tree.

Solution. We proceed by induction on the number $n$ of vertices in the graphs put into $\mathcal{T}$.

For the base case ($n = 1$), consider any natural number $u$ and the graph $G = (\{u\}, \emptyset)$. This graph is a tree because it is (trivially) connected and contains no cycles. Thus, all of the graphs initially put into $\mathcal{T}$ are trees.

For the inductive case, suppose that for some $k \geq 1$, all of the graphs with $k$ vertices in $\mathcal{T}$ are trees. Consider any graph $H$ with $k + 1$ vertices that is put into $\mathcal{T}$. $H$ is in $\mathcal{T}$ because there is some graph $G = (V, E)$ with $k$ vertices in $\mathcal{T}$ such that for some $u \in V$ and some natural number $v \notin V$, the vertices of $H$ are $V \cup \{v\}$ and the edges of $H$ are $E \cup \{uv\}$. By the induction hypothesis, $G$ is a tree because it is an element of $\mathcal{T}$ with $k$ vertices.

We claim that $H$ is a tree using Lemma 10.10.5. The vertex $v$ in $H$ has degree 1 (since $uv$ is the only edge incident to it) and the graph $G$ is $H - v$ ($H$ with vertex $v$ and its incident edge removed) in the terminology of the Lemma. So, since $G$ is a tree, it is connected, that is, $H - v$ is connected, which implies $H$ is connected. Similarly, because $G$ is a tree, it is acyclic, that is, $H - v$ is acyclic, which implies $H$ is acyclic. Thus, $H$ is connected and acyclic, and hence is a tree. This concludes the induction.

(b) (10 points) Prove that for any tree $T = (V, E)$ such that $V$ is a nonempty set of natural numbers, $T \in \mathcal{T}$. Use induction on the number of vertices in $T$.

Solution. The base case is trees $T$ containing one vertex. A tree with one vertex has no edges, so $T$ must be of the form $T = (\{u\}, \emptyset)$ for some natural number $u$. All of these trees are elements of $\mathcal{T}$.

For the inductive case, we assume that for some $k \geq 1$, every tree with a vertex set containing exactly $k$ natural numbers is a element of $\mathcal{T}$. Let $T = (V, E)$ be
a tree such that $|V| = k + 1$ and $V$ is a set of natural numbers. By the Theorem proved in Lecture 20, because $T$ has at least two vertices, it must contain at least two vertices of degree 1, so let $v$ be one such vertex, and let $u$ be the unique vertex of $T$ adjacent to $v$. Let $T - v$ be the graph with vertices $V \setminus \{v\}$ and edges $E \setminus \{uv\}$. Because $T$ is a tree, it is connected and acyclic, and by Lemma 10.10.5, this implies that $T - v$ is also connected and acyclic, and therefore a tree. By the induction hypothesis, because $T - v$ is a tree with exactly $k$ vertices (that are natural numbers), $T - v$ is an element of $T$. Then, adding the vertex $v$ and the edge $uv$ to $T - v$, we get $T$, so $T$ is also an element of $T$, which concludes the induction.

4. (15 points) Describe a recursive algorithm that takes an integer $n \geq 2$ as input and outputs a Hamiltonian cycle in the cube $Q_n$. Give an inductive proof of the correctness of your algorithm.

**Solution.**

The algorithm QHC takes an integer $n \geq 2$ as input. If $n = 2$, it outputs the sequence 00, 01, 11, 10, which is a Hamilton cycle in the cube $Q_2$. If $n > 2$, it calls itself recursively with $(n - 1)$ to get a sequence of $N = 2^{n-1}$ strings of $(n - 1)$ bits each, say

$s_1, s_2, \ldots, s_N$.

Then it outputs the sequence of $2N = 2^n$ strings of $n$ bits each, say

$t_1, t_2, \ldots, t_{2N}$

defined by $t_i = 0s_i$ if $1 \leq i \leq N$, and $t_i = 1s_{2N-i+1}$ if $N + 1 \leq i \leq 2N$. That is, the first $N$ elements are obtained by prefixing the elements of the sequence for $(n - 1)$ by 0 and the second $N$ elements are obtained by prefixing the elements of the reverse of the sequence for $(n - 1)$ by 1.

For example, for $n = 3$ we get the sequence:

000, 001, 011, 010, 110, 111, 101, 100.

We prove by induction on $n$ that QHC on input $n$ outputs a Hamiltonian cycle for $Q_n$, for all integers $n \geq 2$. The base case is $n = 2$, for which the output is a Hamiltonian cycle for $Q_2$ by inspection.

The inductive hypothesis is that QHC on some input $(n - 1) \geq 2$ outputs a Hamiltonian cycle for $Q_{n-1}$. When QHC is called on input $n$, we have $n > 2$ and it will call itself recursively on $(n - 1)$, and by the inductive hypothesis, the result,

$s_1, s_2, \ldots, s_N$
is a Hamiltonian cycle for $Q_{n-1}$, where $N = 2^{n-1}$. That is, every vertex of $Q_{n-1}$ appears exactly once in this list, and for every $i$ such that $1 \leq i \leq N - 1$, there is an edge from $s_i$ to $s_{i+1}$, and finally there is an edge from $s_N$ to $s_1$.

By prefixing the elements of this list first by 0 and then by 1, we generate all $2N$ distinct vertices of $Q_n$. To see that every required pair is joined by an edge, we have the cases:

(a) For $1 \leq i \leq N - 1$, the pair $s_i$ and $s_{i+1}$ differ in one position, so $t_i = 0s_i$ and $t_{i+1} = 0s_{i+1}$ differ in one position.

(b) For $i = N$, the pair $t_N = 0s_N$ and $t_{N+1} = 1s_N$ differ in one position (the leftmost position.)

(c) For $N + 1 \leq i \leq 2N - 1$, the pair $t_i = 1s_{2N-i+1}$ and $t_{i+1} = 1s_{2N-i}$ differ in one position, because $s_{2N-i}$ and $s_{2N-i+1}$ differ in one position.

(d) Finally, $t_{2N} = 1s_1$ differs from $t_1 = 0s_1$ in one position (the leftmost position.)

Thus, by induction we conclude that QHC with input $n$ outputs a Hamiltonian cycle for the cube $Q_n$ for every integer $n \geq 2$.

5. If $n$ is a positive integer, let $I_n$ denote the set $\{1, 2, \ldots, n\}$. We consider an algorithm $A$ that takes as input an undirected graph $G = (V, E)$ specified by a positive integer $n$ and a list of pairs $[(i_1, j_1), (i_2, j_2), \ldots, (i_m, j_m)]$, where $V = I_n$, $|E| = m$ and each pair $(i_k, j_k) \in I_n \times I_n$ specifies an edge of $G$.

The algorithm $A$ initializes a list $T$ to be empty, and an array $L$ of length $n$ to have $L[k] = k$ for all $k \in I_n$. $A$ then processes the pairs from the input list one at a time. For $(i_k, j_k)$, if $L[i_k] = L[j_k]$ then $A$ goes on to the next pair. Otherwise, it adds $(i_k, j_k)$ to the end of $T$ and scans $L$, changing every entry equal to $L[i_k]$ to have value $L[j_k]$, and then goes on to the next pair in the list. When all the pairs have been processed, $A$ halts.

(a) (5 points) Show the intermediate and final values of $L$ and $T$ computed by $A$ for the inputs $n = 6$ and the list of pairs $[(1, 2), (3, 2), (1, 3), (2, 4), (1, 4), (6, 5)]$.

Solution. Initially,

$L = [1, 2, 3, 4, 5, 6]$ and $T = []$.

After pair $(1, 2)$ is processed

$L = [2, 2, 3, 4, 5, 6]$ and $T = [(1, 2)]$. 

5
After pair \((3, 2)\) is processed,
\[ L = [2, 2, 2, 4, 5, 6] \text{ and } T = [(1, 2), (3, 2)]. \]

After pair \((1, 3)\) is processed, \(L\) and \(T\) do not change. After pair \((2, 4)\) is processed,
\[ L = [4, 4, 4, 5, 6] \text{ and } T = [(1, 2), (3, 2), (2, 4)]. \]

After pair \((1, 4)\) is processed, \(L\) and \(T\) do not change. After pair \((6, 5)\) is processed,
\[ L = [4, 4, 4, 5, 5] \text{ and } T = [(1, 2), (3, 2), (2, 4), (6, 5)]. \]

(b) (5 points) Prove that when \(A\) halts, \(L[i] = L[j]\) if and only if \(i\) and \(j\) are in the same connected component of \(G\).

Let \(G_\ell\) be the graph with vertices \(\{1, 2, \ldots, n\}\) and edges \(E_\ell\) consisting of the first \(\ell\) pairs from the input. We prove by (finite) induction on \(\ell\) that \(P(\ell)\) holds for \(\ell = 0, 1, \ldots, m\), where \(P(\ell)\) is the predicate that after \(A\) has processed the first \(\ell\) pairs from the input, \(L[i] = L[j]\) if and only if \(i\) and \(j\) are in the same connected component of \(G_\ell\).

The base case, when \(A\) has processed no pairs, is the graph \(G_0\) with \(n\) vertices and no edges. Each vertex is in its own connected component, and for each \(k\), \(L[k] = k\), so \(P(0)\) holds.

The inductive assumption is that for some \(k \geq 0\), \(P(k)\) holds. If \(k < m\), then consider the processing of pair \((k + 1)\), denoted by \((i_{k+1}, j_{k+1})\). By the inductive hypothesis, if \(L[i_{k+1}] = L[j_{k+1}]\) then \(i_{k+1}\) and \(j_{k+1}\) are in the same connected component of \(G_k\), so adding the edge between them to get \(G_{k+1}\) preserves the same connected components. Because in this case, the values in \(L\) do not change, \(P(k + 1)\) is true. In the case that \(L[i_{k+1}] \neq L[j_{k+1}]\), then \(i_{k+1}\) and \(j_{k+1}\) are in different connected components of \(G_k\). Adding the edge between them joins those two connected components, and in this case, all the vertices in the component with vertex \(i_{k+1}\) have their \(L\) values changed to the \(L\) value associated with the component with vertex \(j_{k+1}\), so in this case also, \(P(k + 1)\) is true, concluding the induction.

Thus, when \(A\) halts, \(G_m\) is equal to \(G\), and \(P(m)\) is true, that is, \(L[i] = L[j]\) if and only if \(i\) and \(j\) are in the same connected component of \(G\).

(c) (5 points) Prove that if the input graph \(G\) is connected then when \(A\) halts, the pairs in the list \(T\) form a spanning tree of \(G\).

Solution. As in part (b), let \(G_\ell\) be the graph with vertices \(\{1, 2, \ldots, n\}\) and the edges determined by the first \(\ell\) pairs of the input. Also, let \(T_\ell\) be the list of pairs in \(T\) after \(A\) has processed the first \(\ell\) pairs in the input. Clearly, the pairs in \(T_\ell\) are a subset of the edges in \(G_\ell\). For \(\ell\) and a connected component \(C\) of \(G_\ell\), we
6. If $G_\ell(C)$ be the subgraph of $G_\ell$ induced by the vertices in $C$, and $T_\ell(C)$ be the graph with vertices $C$ and edges from $T_\ell$ with both endpoints in $C$.

We prove that $P(\ell)$ is true for $\ell = 0, 1, \ldots, m$, where $P(\ell)$ is the claim that for every connected component $C$ of $G_\ell$, $T_\ell(C)$ is a spanning tree of $G_\ell(C)$. The base case, $P(0)$ is true because each vertex of $G_0$ is a separate connected component, and $T_0$ contains no edges.

The inductive hypothesis is that $P(k)$ is true for some $k \geq 0$. If $k < m$, then when $A$ considers pair $(k + 1)$, that is, $(i_{k+1}, j_{k+1})$, if $L[i_{k+1}] = L[j_{k+1}]$ then by (b), $G_{k+1}$ and $G_k$ have the same connected components, and $T_{k+1}$ is equal to $T_k$ to $P(k+1)$ holds. In the case that $L[i_{k+1}] \neq L[j_{k+1}]$, then $i_{k+1}$ and $j_{k+1}$ are in different connected components, say $C_1$ and $C_2$, of $G_k$, which are joined in $G_{k+1}$ to form the connected component $C = C_1 \cup C_2$ when the edge $(i_{k+1}, j_{k+1})$ is added. In this case, the edge $(i_{k+1}, j_{k+1})$ is also added to $T_k$ to get $T_{k+1}$. By the inductive hypothesis, $T_k(C_1)$ is a spanning tree for $G_k(C_1)$ and $T_k(C_2)$ is a spanning tree for $G_k(C_2)$. Then once the edge $(i_{k+1}, j_{k+1})$ is added, the vertices in $C_1$ and $C_2$ can reach each other, so $T_{k+1}(C)$ is connected, and no cycle is created (because no vertex in $C_1$ can reach any vertex in $C_2$ in $G_k$), so $T_{k+1}(C)$ is a tree with vertices $C$, and so is a spanning tree of of $G_\ell(C)$, concluding the induction.

If the input graph $G$ is connected, then when $A$ halts we have $G = G_m$ consists of one connected component, and $T = T_m$ is a spanning tree of $G$.

6. If $G = (V, E)$ is a simple undirected graph, then its complement is the simple undirected graph $G' = (V, E')$ where $E'$ is the set of all $uv$ such that $u$ and $v$ are distinct vertices in $V$ and $uv \not\in E$.

(a) (5 points) Show that the complement of $P_3$ is isomorphic to $P_3$.

Solution. $P_3$ has four vertices, $\{0, 1, 2, 3\}$, and three edges $\{01, 12, 23\}$. Its complement $P'_3$ has the same four vertices and the edges $\{02, 03, 13\}$. We can define an isomorphism $f$ from $P_3$ to $P'_3$ by $f(0) = 2$, $f(1) = 0$, $f(2) = 3$ and $f(3) = 1$. This is a bijection of the vertices mapping edge 01 of $P_3$ to edge 02 of $P'_3$, edge 12 of $P_3$ to edge 03 of $P'_3$, and edge 23 of $P_3$ to edge 13 of $P'_3$. Thus $f$ is an isomorphism, showing that $P'_3$ is isomorphic to $P_3$.

(b) (5 points) Prove that if $T$ is a tree that is not a star, then $T$ contains a simple path of length 3. (Note that all trees with 1, 2 and 3 vertices are stars.)

Solution. Let $T$ be a tree that is not a star. Then by the comment, $T$ has at least 4 vertices, and by the Theorem in Lecture 20, $T$ must have at least two vertices of degree 1. Let $u$ be a vertex in $T$ of degree 1, and $v$ the unique vertex of $T$ adjacent to $u$. Let $w$ be a vertex at the end of a longest simple path from $v$ to any other vertex in $T$. The simple path from $v$ to $w$ must have length greater
than 1, because otherwise every vertex besides \( v \) is adjacent to \( v \) and \( T \) is a star. Thus, the simple path from \( u \) to \( v \) and from \( v \) to \( w \) has length at least 3.

(c) (10 points) Prove that if \( T \) is a tree that is not a star, then the complement of \( T \) contains a Hamiltonian path. (Hints: try examples, try induction, be careful.)

Solution. Let \( P(n) \) be the claim that if \( T \) is any tree with \( n \) vertices that is not a star then its complement \( T' \) contains a Hamiltonian path. We prove \( P(n) \) is true for all integers \( n \geq 4 \).

The base case is \( n = 4 \). If \( T \) is any tree with 4 vertices that is not a star, then by part (b) it must contain a simple path of length 3, and must therefore be isomorphic to \( P_3 \). Its complement \( T' \) is isomorphic to the complement of \( P_3 \), which by part (a) is isomorphic to \( P_3 \) itself, and therefore contains a Hamiltonian path.

The inductive hypothesis is that for some \( k \geq 4 \), \( P(k) \) is true. Let \( T \) be any tree with \( k + 1 \) vertices that is not a star. By the Theorem in Lecture 20, because \( k + 1 \geq 5 \), \( T \) contains at least two vertices of degree 1. Let \( u \) and \( v \) be two distinct vertices of \( T \) of degree 1. By Lemma 10.10.5, \( T - u \), the graph obtained from \( T \) by removing vertex \( u \) and its incident edge, is also connected and acyclic, and therefore a tree, and similarly for \( T - v \).

We claim that at least one of \( T - u \) and \( T - v \) is not a star. Suppose \( T - u \) is a star with center vertex \( w \), adjacent to every other vertex of \( T - u \). Because \( k \geq 4 \), \( w \) is adjacent to \( v \) and at least two other vertices, say \( w_1 \) and \( w_2 \), in \( T - u \). Because \( T \) is not a star, \( u \) is not adjacent to \( w \) in \( T \), so the simple path from \( u \) to \( w \) in \( T \) has length 2. Thus, in \( T - v \), there is a simple path of length 3 from \( u \) to \( w \) to \( w_1 \), and \( T - v \) is not a star. By renaming \( u \) and \( v \), we can assume that \( T - v \) is a tree with \( k \) vertices that is not a star.

Because \( P(k) \) is true, the complement \( (T - v)' \) of \( T - v \) has a Hamiltonian path, say \( v_1, v_2, \ldots, v_k \). The complement \( T' \) of \( T \) has all the edges of \( (T - v)' \) plus an edge between \( v \) and every vertex of \( T \) except for the one vertex it is adjacent to in \( T \). Thus, in \( T' \), \( v \) is adjacent to at least one of the vertices \( v_1 \) and \( v_k \), so we can construct a Hamiltonian path in \( T' \) by adding \( v \) before \( v_1 \) (if it is adjacent to \( v_1 \)) or after \( v_k \) (otherwise). Hence the complement of \( T \) has a Hamiltonian path, which concludes the induction.