MATH20902: Discrete Maths, Solutions to Problem Set 3

These solutions, as well as the corresponding problems, are available at

`http://bit.ly/DiscreteMancMaths`.

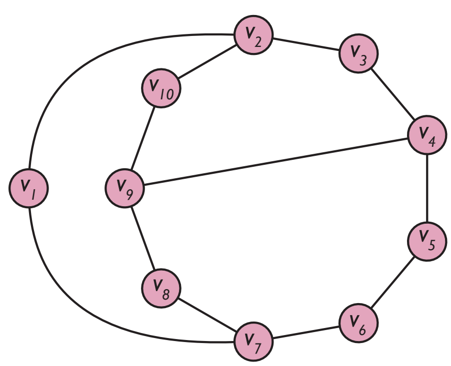
---

(1) Let a and b be any two vertices in the graph. Then the connected component containing a must include a itself, as well as all of a's neighbours, and so must contain at least

$$1 + \deg(a) \;\geq\; 1 + \frac{n-1}{2} \;=\; \frac{n+1}{2}$$

vertices: the same bound holds for the connected component containing b. But this means that the two connected components each contain slightly more than half the vertices in the graph, and hence must overlap (that is, have a non-empty intersection). And if they overlap, they coincide, as connectedness of vertices is an equivalence relation.

(2) The desired graph contains a cycle of length nine, so one might as well start the construction by drawing that and then add extra edges and/or vertices to create the shorter cycles. A bit of trial-and-error produced the graph illustrated below, which has ten vertices and at least one cycle with each of the lengths five through nine.



The table below gives one example for each of the desired lengths, but is not exhaustive: several examples exist for some of the lengths.

Length	Vertex list for cycle
5	$\{v_2, v_3, v_4, v_9, v_{10}, v_2\}$
6	$\{v_4, v_5, v_6, v_7, v_8, v_9, v_4\}$
7	$\{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_1\}$
8	$\{v_2, v_{10}, v_9, v_4, v_5, v_6, v_7, v_1, v_2\}$
9	$\{v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}, v_2\}$

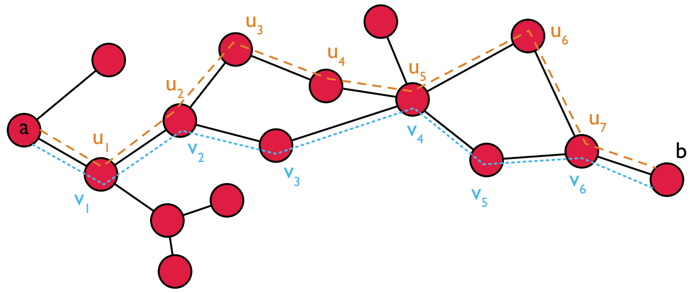


Figure 1: *Two different paths, one shown in orange with long dashes and the other in blue, with shorter dashes, connect vertices a and b. In the terms used in the answer to Problem (3),* $n_1 = 7$ *and* $n_2 = 8$ *while* $j_0 = 2$, $l_1 = 2$ *and* $l_2 = 3$.

(3) The first of the proofs needed here follows from one of the lemmas we proved in lecture:

(i) $\implies$ *(ii)*
To say that G is a tree means that it is a connected acyclic graph, so the first part of (ii) is immediate. The second part, that adding any edge must create a cycle, follows from the fact—proved in lecture—that G must have exactly $(n-1)$ edges and from our lemma about edges in acyclic graphs, which states that an acyclic graph on n vertices has at most $(n-1)$ edges.

(ii) $\implies$ *(iii)*
Statement (iii) has two aspects: first, there is a path connecting any two vertices in G (in other words, G has a single connected component) and second, there is only one such path. One can prove the first part by contradiction: suppose that G has more than one connected component. Then it is possible to add an edge that connects two previously disjoint components and this process cannot create a cycle (otherwise the two components would have been connected already). But this contradicts (ii).

The second aspect, that the path is unique, also follows from the absence of cycles. Suppose there are two vertices—call them a and b—that are connected by more than one path. That is, there exist two sequences of vertices

$$(v_0, \ldots, v_{n_1}) \qquad \text{and} \qquad (u_0 \ldots, u_{n_2})$$

with $u_0 = v_0 = a$ and $u_{n_1} = v_{n_2} = b$ such that all the edges (v_{j-1}, v_j) $1 \leq j \leq n_1$ exist, as do all those given by (u_{k-1}, u_k) $1 \leq k \leq n_2$. Further, the two lists of vertices have to differ: there must be at least one vertex that appears in one path, but not in the other. Then, tracing over one path in the original direction and the other in reverse, we can construct a walk that starts and finishes at $a = u_0 = v_0$

Now, as Figure 1 illustrates, this walk may not be a cycle. It's possible, for example, that the edges (v_0, v_1) and (u_0, u_1) are the same. But as the original paths differ from each other, there must be some smallest index j_0 such that $v_{j_0} = u_{j_0}$, but $v_{j_0+1} \neq u_{j_0+1}$. Some time later, the two paths must come back together (they finish in the same place, at b), so let us find the smallest numbers $l_1 > 0$ and $l_2 > 0$

such that the two paths meet up again after l_1 and l_2 steps, respectively. That is, $v_{j_0+l_1} = u_{j_0+l_2}$. Then

$$(u_{j_0} = v_{j_0}, u_{j_0+1}, \ldots, u_{j_0+l_1} = v_{j_0+l_2}, v_{j_0+l_2-1}, \ldots, v_{j_0} = u_{j_0})$$

is a cycle, contradicting (ii).

(iii) $\implies$ *(i)*
If any two vertices of G are connected, then obviously G is a connected graph. And if the paths that connect pairs of vertices are unique, the graph must be acyclic. For suppose the graph did contain a cycle $(v_0, v_1, \ldots, v_{n-1}, v_0)$: then there are two paths connecting v_0 to v_1. The first consists of the single edge (v_0, v_1) while the second comes from going around the cycle in the opposite sense: $(v_0, v_{n-1}, \ldots, v_1)$.

(4) The problem discusses a graph G with n vertices:

(i) $\implies$ *(ii)*
If G is unicyclic, take e to be any edge in the cycle. The $H = G \backslash e$ is acyclic as (a) the removal of e will destroy the lone cycle in G and (b) removing an edge cannot create any cycles. And H is still connected—part of the definition of *unicyclic* is that the graph is connected and removing a single edge from a cycle still leaves a path that threads through all the vertices in the cycle—so H is a connected, acyclic graph: a tree.

(ii) $\implies$ *(iii)*
From (ii) we know that G is connected, for it is still a tree (and thus connected) even if we remove certain of its edges. Also from (ii), we know that G has one more edge than a tree. But by Theorem 8.1 from lecture, a tree on n vertices has exactly $(n-1)$ edges, so G must have n edges.

(iii) $\implies$ *(i)*
From (iii), we have that G is connected. We also know, from the lemma about acyclic graphs, that if G has n vertices and n edges it must contain at least one cycle, for the lemma says that an acyclic graph on n vertices has at most $n-1$ edges. So all we need to prove is that G has *exactly one* cycle.

Suppose otherwise. That is, suppose G has two or more distinct cycles. Focus attention on a pair of distinct cycles and choose an edge e that appears in one of these cycles, but not in the other. Now consider $H = G \backslash e$. This graph still contains a cycle (the one that didn't include e), but on the other hand it must, by the main theorem about trees mentioned in lecture, be a tree as it's connected (cutting a single edge out of a cycle doesn't spoil connectedness) and it has $n-1$ edges. And trees are acyclic, which yields a contradiction. Thus a connected G with $|E| = |V| = n$ cannot have two distinct cycles and so must be unicyclic.

(5) The theme of this problem is that a tree can't have very many edges, so if it has a vertex of high degree, it must compensate by having a lot of vertices with low degree. I'll give two solutions, one that involves the Handshaking Lemma and another that's based on a clever bit of graph surgery.

Solution based on the Handshaking Lemma

Call the tree under discussion $G(V, E)$ and say there are $|V| = n$ vertices. We're interested in the case

$$\max_{v \in V} \deg(v) = k$$

and we're free to imagine that the vertices are numbered in order of increasing degree, so that the leaves come first and the vertices with maximal degree come last. Suppose there are L leaves and M vertices of maximal degree, then our numbering scheme means

$$\deg(v_j) = \begin{cases} 1 & \text{If } 1 \leq j \leq L \\ k & \text{If } n - M + 1 \leq j \leq n \end{cases}$$

and all other vertices, those v_j with index $L < j \leq (n - M)$, have $2 \leq \deg v_j < k$.

Now, the Handshaking Lemma says

$$\sum_{j=1}^{n} \deg(v_j) = 2|E| = 2(n-1) \tag{5.1}$$

where the second equality follows because G is a tree, so $|E| = n - 1$. On the other hand,

$$\begin{aligned} \sum_{j=1}^{n} \deg(v_j) &= \sum_{j=1}^{L} \deg(v_j) + \sum_{j=L+1}^{n-M} \deg(v_j) + \sum_{j=n-M+1}^{n} \deg(v_j) \\ &= L + \left(\sum_{j=L+1}^{n-M} \deg(v_j) \right) + Mk \\ &\geq L + 2(n - M - L) + Mk \\ &\geq 2n - 2M - L + Mk \end{aligned} \tag{5.2}$$

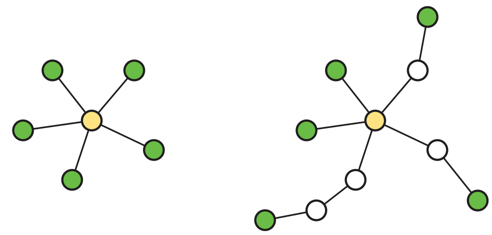
where the penultimate line follows because those v_j that contribute to the sum in the previous line have degree at least two.

Putting (5.1) and (5.2) together yields

$$2n - 2 \;\geq\; 2n - 2M - L + Mk \qquad \text{or} \qquad L \;\geq\; (k-2)M + 2. \tag{5.3}$$

The smallest possible number of leaves thus occurs when there is only a single vertex of maximal degree, so $M = 1$ and (5.3) becomes $L \geq k$.

Conversely, if $L = k$, (5.3) implies $M = 1$ and then the Handshaking Lemma implies that all other vertices must have degree two. Thus trees with maximal degree k and exactly k leaves are "star shaped": they have a single, central degree-k vertex and k "arms", each of which consists of a chain of zero or more degree-two vertices that ends in a leaf. See the examples for $k = 5$ below.



Solution based on graph surgery

This lovely argument, which is illustrated in Figure 2 was invented by Goran Malic, a postgrad who helped with the example classes in 2014.

Choose a vertex of maximal degree—call it, say, v—such that $\deg(v) = k$, and form the graph $G' = G \backslash v$. Since the original graph G was a tree, the new one must have k connected components, each of which is itself a tree: let's call them $T_1, T_2, \dots T_k$. Each of the T_j must be either

- an isolated vertex
- a tree with with two or more vertices. In this case T_j must have at least two leaf nodes (this follows from a technical lemma proved in the lecture about trees).

Either way, the edge connecting T_j to v can eliminate at most one leaf node, so each T_j contains at least one vertex that is a leaf node in the original graph G. But this means that G has at least k leaves, which is the result we sought.

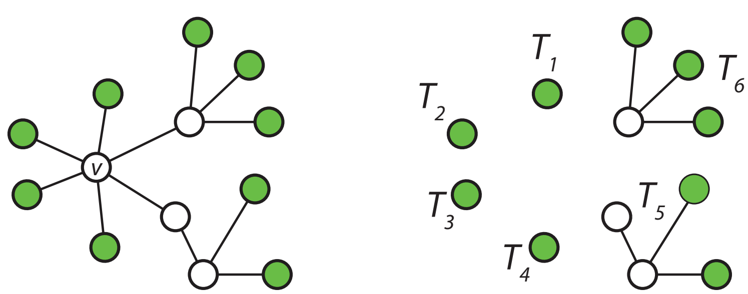


Figure 2: *An approach to Problem (5). The graph at left has a single vertex of maximal degree $k = 6$ that's labelled v. The graph at right is $G \backslash v$ and is a forest consisting of six trees: $T_1 \cdots T_4$ are isolated vertices while T_5 and T_6 are smaller trees, each of which has at least two leaves.*

(6) If $(d_1, d_2, \ldots, d_n)$ is the degree sequence of a tree $G(V,E)$ then $|V| = n$ and $|E| = |V| - 1 = n - 1$. And if we number the vertices in order of increasing degree, the Handshaking Lemma says

$$\sum_{j=1}^{n} d_j = \sum_{j=1}^{n} \deg v_j = 2|E| = 2(n-1), \tag{6.1}$$

which proves the easier half of the if-and-only-if statement in the problem.

The other half says that if a degree sequence $(d_1, d_2, \ldots, d_n)$ satisfies (6.1), then there exists a tree that actually has this degree sequence. I'll offer two proofs of this, one by induction on the number of vertices (or, equivalently, on the number of entries in the degree sequence) and another, constructive proof invented by Simon Ge, a student who did Discrete Maths in 2016.

Inductive argument:
Take $n = 2$ as the base case. The only two-vertex degree sequence that is compatible with (6.1) is $(d_1, d_2) = (1, 1)$, which is also the degree sequence of the only two-vertex tree.

Now for the inductive step: suppose the result is true for all degree sequences that include n_0 or fewer entries and satisfy (6.1). Then consider a degree sequence

$$(d_1, d_2, \ldots, d_{n_0}, d_{n_0+1}) \tag{6.2}$$

that has one more entry, but still satisfies (6.1) with $n = n_0 + 1$. We know—following an argument made in lecture—that any such sequence must include at least two ones, so $d_1 = d_2 = 1$. We also know that $d_{n+1} > 1$, as the entries in the degree sequence are arranged in ascending order and (6.1) can't be satisfied if all the entries d_j are one. But this means, perhaps after some rearrangement to put the entries back into increasing order, that

$$(d_2, \ldots, d_{n_0}, d_{n_0+1} - 1) \tag{6.3}$$

is also a degree sequence satisfying (6.1), but with $n = n_0$ entries. That is, we can remove $d_1 = 1$ from the beginning of (6.2) and reduce the final, largest entry by one and we'll get a new, shorter sequence that still satisfies (6.1). This new, shorter sequence has only n_0 entries so, by the inductive hypothesis, there is a tree $G(V,E)$ with $|V| = n_0$ whose degree sequence is composed of the same numbers as (6.3). The tree G thus contains a vertex of degree $(d_{n_0+1} - 1)$: call this vertex v. We can now make a new graph $H(V', E')$ by adding a single new leaf node to G whose sole edge attaches it to v: the graph H produced in this way is also a tree and has degree sequence

$$(1, d_2, \ldots, d_{n_0}, d_{n_0+1})$$

which is the same as (6.2), so we are finished.

The argument above contains the seeds of a recursive algorithm that accepts an arbitrary list of positive integers $d_j \leq d_{j+1}$ satisfying (6.1) and constructs a tree having the list as its degree sequence: you might try to write this algorithm out in detail. Typically there are *lots* of non-isomorphic trees that have the same degree sequence, as Simon Ge's constructive proof makes clear.

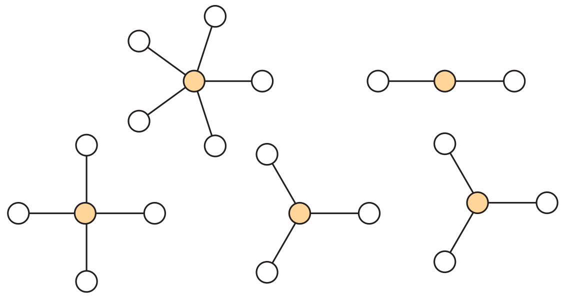


Figure 3: *A forest consisting of star-shaped trees. This is the first step in constructing a tree with degree sequence* $(1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 3, 3, 4, 5)$.

Constructive proof:
Given a sequence $(d_1, d_2, \ldots, d_n)$ that satisfies (6.1), we begin by removing those entries that are equal to one. There are always at least two of these (this is easy to prove using the Handshaking Lemma), but there must also be some entries bigger than one. Let's say that there are $k < n$ entries equal to one, so that $d_1 = d_2 = \cdots = d_k = 1$, but $1 < d_{k+1} \le d_{k+2} \le \cdots \le d_n$. The construction then proceeds as follows:

- Build a forest such as the one illustrated in Figure 3, which consists of $n - k$ star-shaped graphs, one for each $d_j > 1$. One can also think of this as a forest whose $n - k$ connected components are isomorphic to the complete bipartite graphs K_{1,d_j} with $k < j \le n$.
- Now, repeatedly merge two of the trees in the forest as shown in Figure 4. That is, first select two trees (arbitrarily), then choose a leaf in each one and replace the chosen pair of leaves with an edge that joins the trees. The result of this merger is clearly still a tree (persuade yourself that the new edge can't create a cycle) and so, even after the merger, we still have a forest, though with one tree less. And as each merger reduces the number of trees in the forest by one, eventually we must get down to a single tree.

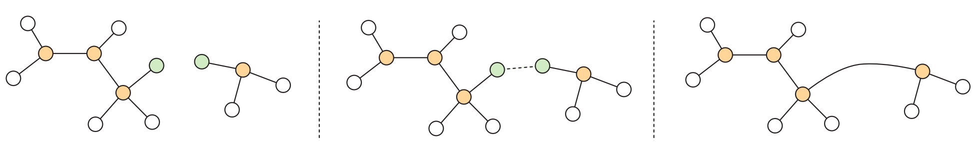


Figure 4: *The panels show, left-to-right, the steps required two merge two trees.*

To see that this construction works, note first that the process of merging trees never changes the degree of a non-leaf vertex. Thus since we started out with $n - k$ vertices having degrees d_{k+1} through d_n, we'll still have vertices with those degrees in the final tree. Thus all we need to do is check that this tree has the right number, k, of leaves.

The original forest of star-shaped graphs has

$$\sum_{j=k+1}^{n} d_j \;=\; \left(\sum_{j=1}^{n} d_j \right) - k \;=\; 2(n-1) - k$$

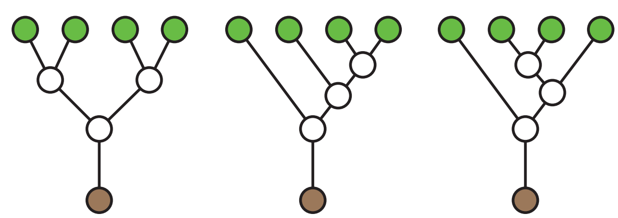
leaves, where I have used Eqn. (6.1) and the fact that $d_1 = \cdots = d_k = 1$. Each merger eliminates two leaves and we do $n-k-1$ of them (one less than $n-k$, which is the number of trees in the forest-of-stars) and so the final tree has

$$\begin{aligned}
\underbrace{2(n-1)-k}_{\text{leaves in forest-of-stars}} - \underbrace{2(n-k-1)}_{\text{eliminated during mergers}} &= (2n-2-k) - (2n-2k-2) \\
&= 2n-2-k-2n+2k+2 \\
&= k,
\end{aligned}$$

which is just the right number for the final tree to have the original sequence, $(d_1,\, d_2, \ldots,\, d_n)$, as its degree sequence.

(7) This problem has to do with *rooted binary trees*. Note that some authors, including Jungnickel, define a rooted tree as an ordinary graph-theoretic tree (that is, a connected, acyclic graph) with a distinguished node, but in this problem I intended the term to mean "a tree with a distinguished *leaf* node."

(a) Here are three trees, the middle and rightmost of which are isomorphic.



(b) One can prove this by induction on the number of leaves. The simplest rooted binary tree has $n = 2$ leaves and one internal node. That is, it has three vertices of degree one (the root and two leaves) and one vertex of degree 3: see Figure 5. This graph has four vertices, so the result is established for the

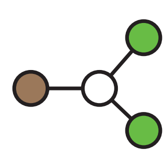


Figure 5: *The simplest rooted binary tree.*

base case. Now suppose it is true for all rooted binary trees with no more than n leaf nodes and consider a tree G with $n+1$ leaves. Note that every leaf node must be connected by its sole edge to an internal node (a node of degree three). Choose one of the leaves—call this vertex v_0—and delete it, along with its edge. The resulting graph, $G' = G \backslash v_0$, is not a rooted binary tree because the node—call it u—that was formerly connected to v_0 now has degree 2.

Call u's remaining neighbours v_1 and v_2. Thus the edge set for G' contains (u, v_1) and (u, v_2). Now make a third graph, G'', by deleting u and its two edges from G' and adding a new edge (v_1, v_2) in their place. An example of the sort of tree surgery we're discussing appears in Figure 6.

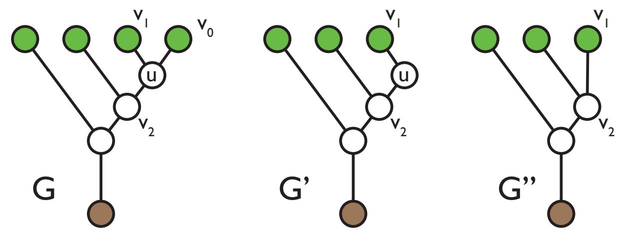


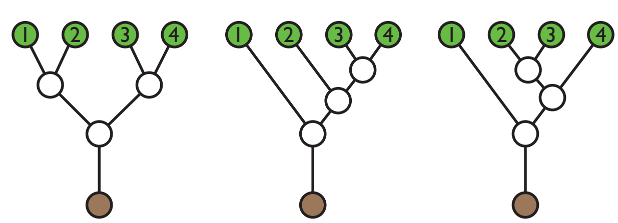
Figure 6: *Removing a leaf from a binary tree.*

G'' is again a binary rooted tree, but it has only n leaves, so the inductive hypothesis applies, and G'' has an even number of nodes. But then G does too, for we deleted exactly two nodes in passing from G to G''.

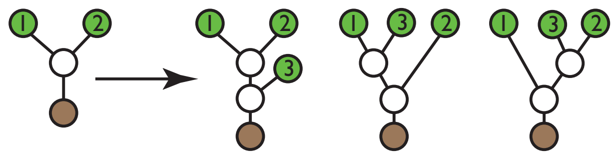
(c) A careful replay of the logic in the previous answer allows one to conclude that a binary rooted tree with n leaves has $2n$ vertices: the n leaves plus $(n-1)$ internal nodes and the root.

(d) A binary rooted tree is a special case of a tree (that is, a connected acyclic graph), so it has one less edge than it has vertices. Thus, in light of the previous answer, a rooted binary tree with n leaves has $2n-1$ edges.

(e) The first thing to do here is to clarify when we should think of two trees as different. If we number the leaf nodes as shown, then the three trees below are all different.



Now imagine that we have a collection of all the distinct trees with n leaves—say there are W_n of them—and imagine trying to construct the collection of distinct trees with $(n+1)$ leaves. We can build them by reversing the construction from part (b) above. To add the $(n+1)$-th leaf node we must break an edge and insert a new internal vertex in its middle. This new internal vertex is also connected to the new leaf. The figure below shows three distinct 3-leaf binary rooted trees derived from a labelled version of the one in Figure 5 by adding the new leaf and accompanying internal node to the various edges of the 2-leaf tree.



A little thought shows that we can insert our new pair of vertices (the internal one and the attached new leaf) on any of the edges of the graph, so each distinct graph with n leaves gives rise to $(2n-1)$ distinct graphs having $(n+1)$ leaves. This leads to the recursion

$$\begin{aligned} W_2 &= 1 \\ W_{n+1} &= (2n-1)W_n. \end{aligned} \tag{7.4}$$

Thus, for example,

$$\begin{aligned} W_5 &= (2 \times 4 - 1)W_4 \\ &= (2 \times 4 - 1)(2 \times 3 - 1)W_3 \\ &= (2 \times 4 - 1)(2 \times 3 - 1)(2 \times 2 - 1)W_2 \\ &= 7 \times 5 \times 3 \times 1 \end{aligned}$$

If you want a tidier form, it's possible to stare at the recursion (7.4) for a while and then conjecture and prove by induction that

$$W_n = \frac{(2n-3)!}{2^{n-2}(n-2)!}.$$

This grows incredibly quickly with increasing n

n	W_n
2	1
3	3
4	15
5	105
6	945
7	10395
8	135135
9	2027025
10	34459425
11	654729075
12	13749310575
13	316234143225
14	7905853580625