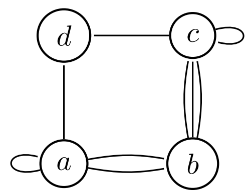
TMA4140
DISKRET MATEMATIKK – DISCRETE MATHEMATICS
NTNU, HØST/FALL2020

SOLUTIONS – EXERCISE SET 11 / ØVING 11

Exercise/Oppgave
1. Section/Sektion 10.3: *17, 19, 23*

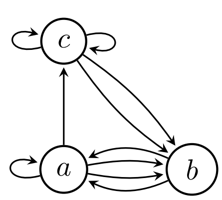
Solution. 17. Recall that, given an adjacency matrix $A = (a_{ij})$ of an undirected graph, we say that there are k edges between the vertices i and j if and only if $a_{ij} = k$. Ordering the vertices of the graph by a, b, c and d, we have:



19. Recall that, for a directed multigraph of b vertices $v_1, \ldots, v_n$, the adjacency matrix $A = (a_{ij})_{i,j=1}^n$ is defined by $a_{ij} = k$ if and only if there are k directed edges from v_i to v_j. The adjacency matrix of the directed graph is the following:

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}.$$

23. According to the definition of the adjacency matrix of a directed multigraph, the corresponding graph is the following:



□

Exercise/Oppgave
2. Section/Sektion 10.4: *26b, 30, 56*

Date: November 8, 2020.

Solution. 26b. Find the number of paths between c and d in the graph in Figure 1 of length 3. By Theorem 2, if A is the adjacency matrix of the graph and if the vertices are ordered as a, b, c, d, e, f, then the entry (3,4) of A^3 is precisely the number of paths between c and d of length 3. Notice that

$$A = \begin{pmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \end{pmatrix}.$$

Making the computations, we see that the entry (3,4) of A^3 is 8. Hence there are 8 paths between c and d of length 3.

30. Let v a vertex of odd degree, and define A the connected component that contains v. Notice that A is also a simple graph. By a consequence of the handshaking lemma, there is an even number of vertices of odd degree in A. Since v has odd degree, there must be a $w \neq v$ in A of odd degree. By definition of connected component, there is a path between v and w, and we conclude.

56. Recall by Theorem 2 that if A is the adjacency matrix of a graph, then the entry (i, j) of A^r is exactly the number of paths of length r from v_i to v_j. Hence, the length of the shortest path from a vertex v to a vertex w is equal to the minimum of r such that the entry (i, j) of A^r is different from 0, where $v_i = v$ and $v_j = w$. □

Exercise/Oppgave

3. Section/Sektion 10.5: *3, 30, 36, 48*

Solution. 3. Notice that the graph has two vertices of degree 3, two vertices of degree 4, and one vertex of degree 6. In particular, by Theorem 1, there is no Euler circuit in the graph since not all the vertices have even degree. On the other hand, there is an Euler path given by $a, b, e, c, e, b, d, e, b, d$.

30. There is no Hamilton circuit on this graph. The reason is because, if we want to go from $\{a, b, c\}$ to $\{d, e, f\}$ or conversely, then we must pass through c and f two times since we must return to the initial subset.

36. Yes. $a, b, c, f, i, h, g, d, e, a$ is a Hamilton circuit.

48. If $n = 2m - 1$, consider two complete graphs K_m where both graphs share one of their points. Each vertex except by the common vertex has degree $(n-1)/2 = m - 1$, and the central vertex has degree $n - 1$. Reasoning in a similar way to 3, this graph has no Hamilton circuit. In the even case $n = 2m$, then each vertex having degree at least $(n-1)/2$ is equivalent that each vertex has degree at least $n/2$, and by Dirac's Theorem, any such graph has a Hamilton circuit. □

Exercise/Oppgave

4. Section/Sektion 10.6: *3, 14, 18*

Solution. 3. A path with starts at a that has no interior vertices other than a must consists of one edge that has a as one of its endpoints. There are two possibilities, b and c. Then the closest vertex from a is c and the shortest path has length 2.

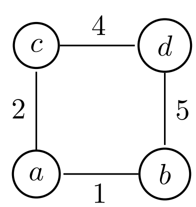
Now, we can find the second closest vertex by examining all the paths that begin with the shortest path from a to a vertex in the set $\{a, c\}$, followed by an edge that has one endpoint in $\{a, c\}$ and its other endpoint not in this set. The possibilities are $\{b, d, e\}$ with paths a, b; a, c, b; a, c, d; and a, c, e. We can see the second closest vertex from a is b and the shortest path is a, b and has length 4.

Now we find the third closest vertex from a, by examining all the paths that begin with the shortest path from a to a vertex in the set $\{a, c, b\}$, followed by an edge that has one endpoint in $\{a, c, b\}$ and its other endpoint not in this set. We can see that the the shortest one is a, c, d and has length 6.

For the fourth closest vertex from a, following the above procedure we find that the corresponding path is a, c, d, e and has length 7. For the fifth closest vertex from a, we have that the shortest path is a, c, d, f and has length 11. For the sixth closest vertex from a, the shortest path is a, c, d, e, g and has length 12. Finally, the seventh closest vertex from a, the corresponding shortest path is a, c, d, e, g, z and has length 16. This allows to conclude that the shortest path from a to z is a, c, d, e, g, z and has length 16.

14. Let G be an undirected graph. We can construct a weighted graph from G just by giving the weight 1 to each edge of G. In this case, a path of length n corresponds to a path $x_0, \ldots, x_n$. Then, the solution shortest path problem in a weighted graph is equivalent to find the path with the least number of edges between two vertices in an undirected graph.

18. The answer is no. Indeed, consider the graph



Clearly the weights of the edges are distinct, and there are two shortest paths from a to d: a, c, d and a, b, d. Both have length 6. □

Exercise/Oppgave

5. Section/Sektion 11.1: *3, 16, 22*

Solution. 3.

(1) Which vertex is the root? a.
(2) Which vertices are internal? a, b, f, c, h, d, j, q and t.
(3) Which vertices are leaves? $e, l, m, n, g, o, p, i, k, r, s$ and u.
(4) Which vertices are children of j? q and r.
(5) Which vertex is the parent of h? c.

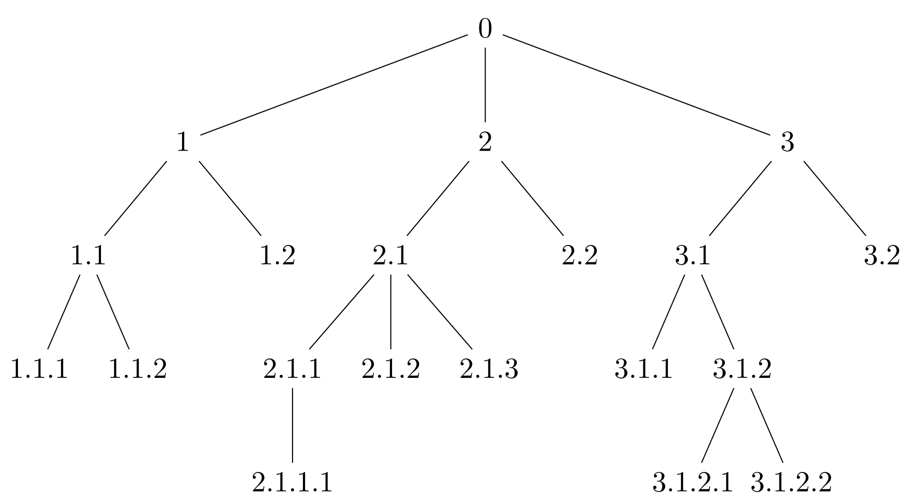
(6) Which vertices are siblings of o? p.
(7) Which vertices are ancestors of m? f, b and a.
(8) Which vertices are descendants of b? e, f, l, m and n.

16. We have that $K_{1,n}$ is a tree for every $n \geq 1$, since obviously it is a graph without circuits. Analogously, we have that $K_{m,1}$ is a tree for every $m \geq 1$. On the other hand, $K_{m,n}$ is not a tree if $n, m \geq 2$. Indeed, if (V_m, V_n) is the bipartition of $K_{m,n}$, take distinct vertices $a, b \in V_m$ and $u, v \in V_n$. Since $K_{m,n}$ is complete bipartite, then we can construct the path a, u, b, v, a an this is a circuit. Hence $K_{m,n}$ is not a tree if $m, n \geq 2$.

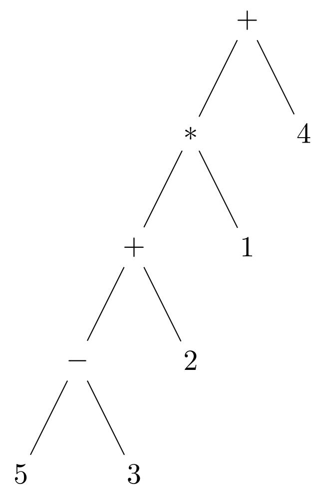
22. We can model the chain using a m-ary tree, with $m = 5$. Since a person sends the letter to other 5 people, or does not send out the letter, we have that every internal vertex of the tree has exactly 5 children, and then our tree is a full m-ary tree. By assumption, 10000 people send out the letter, and then there are $i = 10000$ internal vertices in the tree. By Theorem 3, the tree contains $n = 5(10000) + 1 = 50001$ vertices. This means that, without counting the root since he/she did not received any letter, there are 50000 people who received the letter. The number of people who did not send out the letter is equal to the number of leaves of the tree, and this number is equal to $n - i = 40001$. □

Exercise/Oppgave
6. Section/Sektion 11.3: *6a, 22a, 24a*

Solution. 6a. Yes, the corresponding tree is the following:



22a. Draw the ordered rooted tree corresponding to the arithmetic expression $+ * + - 5\,3\,2\,1\,4$ written in prefix notation. Then write each expression using infix notation. The corresponding tree is the following:



The infix notation for the expression is $5 - 3 + 2 * 1 + 4$.

24a. What is the value of the postfix expression $5\ 2\ 1 - -\ 3\ 1\ 4 + +\ *$? Recall that in order to evaluate an expression from its postfix form, we work from left to right, carrying out operations whenever an operator follows two operands. After an operation is carried out, the result of this operation becomes a new operand. Hence the result is

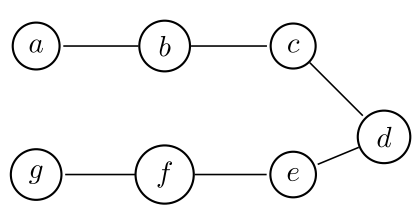
$$\begin{gathered} 5\ \underbrace{2\ 1\ -}_{2-1=1}\ -\ 3\ 1\ 4 + +\ * \\ \underbrace{5\ 1 -}_{5-1=4}\ 3\ 1\ 4 + +\ * \\ 4\ 3\ \underbrace{1\ 4\ +}_{1+4=5}\ +\ * \\ 4\ \underbrace{3\ 5\ +}_{3+5=8}\ * \\ 4\ 8\ * = 32. \end{gathered}$$

Hence the value of the postfix expression is 32. □

Exercise/Oppgave

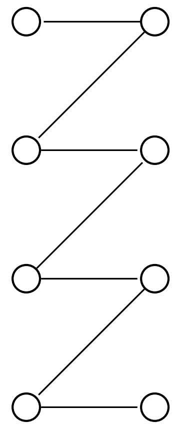
7. Section/Sektion 11.4: *3, 7b, 20*

Solution. 3. Recall that a spanning tree of G is a subgraph of G that is a tree and contains all the vertex of G. A spanning tree for the graph is the following:



Observe that we remove the edges $\{a, d\}, \{d, g\}, \{e, g\}, \{c, e\}, \{b, e\}, \{b, f\}, \{b, g\}$ and $\{a, g\}$ from G.

7b. Find a spanning tree for $K_{4,4}$. Using the definition of complete bipartite graph, a spanning tree for $K_{4,4}$ is given by



20. *Describe the trees produced by breadth–first search and depth–first search of the complete graph K_n, where n is a positive integer.* By definition, given two distinct vertices in K_n, there is an edge between them. The depth–first search algorithm consists in selecting an arbitrary vertex as the root, then form a path starting at this vertex by successively adding vertices and edges, where each new edge is incident with the last vertex i the path and a vertex not already in the path. We will have two possibilities: the path goes through all the vertices of the graph, or does not go. However, in the case of a complete graph, it is always possible to add new vertices and edges to the path of the algorithm depth-first search. Hence the spanning three of K_n found by the depth–first algorithm is a simple path $v_1, v_2, \ldots, v_n$ between the n vertices of K_n.

Now, consider the case of the breadth-first algorithm. This algorithm consists in choosing an arbitrary vertex of the graph as the root. Then, add all the incident vertices to the initial vertex in the graph as the children of the root. Observe that, for the case of K_n, given an arbitrary vertex w, all the remaining vertices are incident to w. Hence the spanning tree produced by the breadth-first algorithm is a corolla: a rooted tree with n vertices and such that $n-1$ of these vertices are leaves. □