CMPUT 204 (B1) – Assignment #4

DUE DATE: August 16, 2000

Solutions

---

1. *First Order Nonhomogeneous Linear Difference Equations*

   Consider the following 1^{st} order nonhomogeneous linear difference equation

$$\begin{aligned} a_n &= c_n a_{n-1} + d_n, \quad n \geq 1 \\ a_0 &= d_0 \end{aligned}$$

   where the coefficients c_n and d_n may both depend on n.

   (a) Prove by induction that a particular solution to this problem is given by

$$a_n = \sum_{j=0}^{n} \left(\prod_{i=j+1}^{n} c_i \right) d_j$$

   for $n \geq 0$. Here we interpret the empty product as $\prod_{i=k}^{n} c_i = 1$ for $k > n$.

   (b) In the last assignment, we saw that the average number of comparisons done by the function `selection` in finding the k^{th} smallest entry in a list with n entries satisfied the discrete initial value problem

$$\begin{aligned} a_n &= \frac{n a_0 + 2\left[a_1 + 2a_2 + 3a_3 + \cdots + (n-1)a_{n-1}\right]}{n^2} + n - 1, \quad n \geq 1 \\ a_0 &= 0. \end{aligned}$$

   Use the result in part (a) to find the exact solution to this discrete initial value problem.

   SOLUTION:

   (a) To see where this formula came from, we can write out the first few terms of the sequence $\{a_n\}$.

$$\begin{aligned} a_0 &= d_0 \\ a_1 &= c_1 a_0 + d_1 = c_1 d_0 + d_0 \\ a_2 &= c_2 a_1 + d_2 = c_2 c_1 d_0 + c_2 d_1 + d_2 \\ a_3 &= c_3 a_2 + d_3 = c_3 c_2 c_1 d_0 + c_3 c_2 d_1 + c_3 d_2 + d_3 \end{aligned}$$

   and the pattern that emerges is

$$\begin{aligned} a_n &= \left(\prod_{i=1}^{n} c_i\right) d_0 + \left(\prod_{i=2}^{n} c_i\right) d_1 + \cdots + \left(\prod_{i=n}^{n} c_i\right) d_{n-1} + d_n \\ &= \left(\prod_{i=1}^{n} c_i\right) d_0 + \sum_{j=1}^{n} \left(\prod_{i=j+1}^{n} c_i \right) d_j \\ &= \sum_{j=0}^{n} \left(\prod_{i=j+1}^{n} c_i \right) d_j, \end{aligned}$$

   where we used the fact that the empty product is $\prod_{i=n+1}^{n} c_i = 1$.

Now we prove this is the solution to the initial value problem using the principle of mathematical induction. Suppose that a_n is a solution to the initial value problem, for $n = 0$, we have

$$\sum_{j=0}^{0}\left(\prod_{i=j+1}^{0} c_i\right) d_j = \left(\prod_{i=0+1}^{0} c_i\right) d_0 = d_0,$$

and since $a_0 = d_0$, then the result is true for $n = 0$. Now suppose that

$$a_n = \sum_{j=0}^{n}\left(\prod_{i=j+1}^{n} c_i\right) d_j$$

for some $n \geq 0$, then from the difference equation, we have

$$\begin{aligned}
a_{n+1} &= c_{n+1}a_n + d_{n+1} \\
&= c_{n+1}\left[\sum_{j=0}^{n}\left(\prod_{i=j+1}^{n} c_i\right) d_j\right] + d_{n+1} \\
&= \sum_{j=0}^{n}\left(\prod_{i=j+1}^{n+1} c_i\right) d_j + 1 \cdot d_{n+1} \\
&= \sum_{j=0}^{n}\left(\prod_{i=j+1}^{n+1} c_i\right) d_j + \left(\prod_{n+2}^{n+1} c_i\right) d_{n+1} \\
&= \sum_{j=0}^{n+1}\left(\prod_{i=j+1}^{n+1} c_i\right) d_j
\end{aligned}$$

and so the result is also true for $n+1$. Thus, by the principle of mathematical induction, the result is true for all $n \geq 0$.

(b) In order to find the solution to the initial value problem

$$\begin{aligned}
a_n &= \frac{na_0 + 2\left[a_1 + 2a_2 + 3a_3 + \cdots + (n-1)a_{n-1}\right]}{n^2} + n - 1, \quad n \geq 1 \\
a_0 &= 0,
\end{aligned}$$

we multiply the difference equation by n^2, and then replace n by $n-1$ to get the two equations

$$\begin{aligned}
n^2 a_n &= 2\sum_{k=1}^{n-1} k a_k + n^2(n-1) \\
(n-1)^2 a_{n-1} &= 2\sum_{k=1}^{n-2} k a_k + (n-1)^2(n-2).
\end{aligned}$$

Subtracting the second from the first and simplifying, we obtain the following discrete initial value problem for a_n,

$$\begin{aligned}
a_n &= \left(\frac{n^2-1}{n^2}\right) a_{n-1} + \frac{(n-1)(3n-2)}{n^2}, \quad n \geq 1 \\
a_0 &= 0.
\end{aligned}$$

Now we put $c_j = \dfrac{j^2-1}{j^2}$ and $d_j = \dfrac{(j-1)(3j-2)}{j^2}$ for all $j \geq 1$, and let $d_0 = 0$, then from part (a) we have

$$a_n = \sum_{j=1}^{n} \left(\prod_{i=j+1}^{n} c_i \right) d_j$$

for $n \geq 1$.

For $j \geq 1$, we have

$$\prod_{i=j+1}^{n} \left(\frac{i^2-1}{i^2} \right) = \prod_{i=j+1}^{n} \left(\frac{i-1}{i} \right) \prod_{i=j+1}^{n} \left(\frac{i+1}{i} \right) = \frac{j}{n} \cdot \frac{n+1}{j+1} = \frac{n+1}{n} \cdot \frac{j}{j+1},$$

and

$$d_j = \frac{(j-1)(3j-2)}{j^2},$$

so that

$$a_n = \sum_{j=1}^{n} \left(\frac{n+1}{n} \right) \cdot \frac{j}{j+1} \cdot \frac{(j-1)(3j-2)}{j^2},$$

that is,

$$a_n = \frac{n+1}{n} \sum_{j=1}^{n} \frac{(j-1)(3j-2)}{j(j+1)} = \frac{n+1}{n} \sum_{j=1}^{n} \left[3 - \frac{10}{j+1} + \frac{2}{j} \right].$$

Therefore,

$$a_n = 3n + 13 - \frac{8(n+1)}{n} H_{n+1} + \frac{8}{n}$$

for $n \geq 1$.

2. *Unicyclic Graphs*

   If $G = (V, E)$ is an undirected graph with vertex set V and edge set E, and if $p = |V|$ is the number of vertices and $q = |E|$ is the number of edges, we say that G is *unicylic* if and only if G is connected and G contains exactly one cycle.

   Show that the following statements are equivalent, that is, either they are all true or they are all false.

   (a) G is unicyclic.

   (b) G is connected and $p = q$.

   (c) G has exactly one cycle and $p = q$.

   SOLUTION: We will show that (a) implies (b), (b) implies (c), and finally that (c) implies (a).

   (a) implies (b). Suppose that G is a unicylic graph with p vertices and q edges, then G is connected and has exactly one cycle C. If we remove an edge e from the cycle C, then the reulting graph $G - e$ has p vertices and $q - 1$ edges. The graph $G - e$ is still connected and has no cycles, that is, $G - e$ is a tree. Therefore, $p = (q - 1) + 1 = q$, and (b) is true.

   (b) implies (c). Let G be a graph with p vertices and q edges, and suppose that G is connected and $p = q$. If G had no cycles, then G would be a tree and we would have $p = q+1$, which is a contradiction. Thus, G must contain at least one cycle C, and if we remove an edge e from the cycle, the graph $G - e$ is still connected and has p vertices and $q - 1$ edges, and since $p = q$, then $p = (q - 1) + 1$, so that $G - e$ is a tree. Thus, the original graph G has exactly one cycle and $p = q$.

   (c) implies (a). Let G be a graph with p vertices and q edges, and suppose that G has exactly one cycle C and $p = q$. If we remove an edge e from the cycle, then $G - e$ has no cycles, and $p = (q - 1) + 1$, so that $G - e$ is a tree and hence is connected. Now add the edge e back to the graph, the resulting graph G is also connected and has exactly one cycle, that is, G is unicyclic.

3. *Connected Graphs and Spanning Trees*

   Let $G = (V, E)$ be an undirected graph with $p = |V|$ vertices and $q = |E|$ edges.

   (a) Show that G is connected if and only if G has a spanning tree.

   (b) If G is connected, what is the smallest number γ of edges that must be removed from G in order that no cycles remain? Express γ in terms of p and q.

   Note: The number γ is called the **cycle rank** or the **cyclomatic number** of the graph G.

   (c) Show that if

   $$q > \frac{(p-1)(p-2)}{2},$$

   then G is connected.

   (d) If G is connected, show that the height of any DFS spanning tree for G starting from a given root v must be at least as large as the height of a BFS spanning tree for G with root at v.

   (e) If G is connected, by starting with different roots, it might be possible for a DFS spanning tree for G to have a smaller height than a BFS spanning tree for G. Find a graph G in which this can happen.

   SOLUTION:

   (a) If G has a spanning tree T, then T is an acyclic connected subgraph of G which contains all the vertices of G. If v and w are any two distinct vertices in G, then since v and w are in T and T is connected, there is a path from v to w in T. But this means that there is a path from v to w in G, since T is a subgraph of G. Thus, given any two vertices in G, there is a path in G joining these two vertices, so G is connected.

   Conversely, suppose that G is connected, but G is not a tree, then G must contain a cycle. If we remove an edge from this cycle, then the resulting subgraph G_1 is still connected and has the same vertices as G. If G_1 is not a tree, then it must contain a cycle. If we remove an edge from this cycle, then the resulting subgraph G_2 is still connected and has the same vertices as G. Since there are only finitely many edges in the graph, this process must eventually stop, say with a subgraph G_k of G which is still connected and has the same vertices as G, but this last subgraph G_k, has *no* cycles, and is therefore a spanning tree for G.

(b) If G is connected, and we remove edges from the cycles of G until we reach a spanning tree $T = (V, E_T)$ for G, then the number of edges E_T in T must satisfy $E_T = p - 1$. Therefore, we must remove at least

$$\gamma = q - |E_T| = q - p + 1$$

edges from G in order that no cycles remain. If we remove less than γ, then $|E_T| > p - 1$, and T would contain a cycle; if we remove more than γ, then $|E_T| < p - 1$, and T would not be connected.

(c) Suppose that G is not connected, then there exists an integer k, with $1 \leq k \leq p - 1$, such that G has a connected component with exactly k vertices. The largest number of edges that G could possibly have is

$$\binom{k}{2} + \binom{p-k}{2} = \tfrac{1}{2}\{k(k-1) + (p-k)(p-k-1)\} = \tfrac{1}{2}\{2k^2 - 2pk + p(p-1)\}.$$

As a function of k, the quadratic function $f(k) = \frac{1}{2}\{2k^2 - 2pk + p(p-1)\}$ has a minimum at $k = \frac{1}{2}p$ and a maximum at $k = 1$ and $k = p - 1$. Therefore, if G is not connected, the number of edges cannot be larger that the maximum of this quadratic funtion, that is, the number of edges m cannot be larger than $\dfrac{(p-1)(p-2)}{2}$.

Therefore, if

$$q > \frac{(p-1)(p-2)}{2},$$

then G is connected.

(d) First we show that in a connected undirected graph G, the length of a shortest path in G between vertices v and w equals the level number in any BFS spanning tree for G which is rooted at v. The proof is by induction on the length of a shortest path.

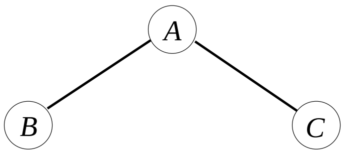
Let $G = (V, E)$ be a connected undirected graph and let $v, w \in V$, and let $T = (V, E_T)$ be a BFS spanning tree for G rooted at v. If a shortest path in G from v to w has length 0, then $v = w$, and in the BFS tree rooted at v, the level number of w is also 0. If the length of a shortest path in G from v to w has length 1, then w is adjacent to v, so that w is at level 1 in the BFS spanning tree T.

Now suppose that the result is true for all shortest paths of length ℓ. If the length of a shortest path in G from v to w is $\ell + 1$, let w' be the next–to–last vertex in a shortest path in G from v to w. By the inductive hypothesis, w' is at level ℓ in the BFS spanning tree T. If w is at a level less than or equal to ℓ in T, then clearly the length of a shortest path in G from v to w would also be less than or equal to ℓ. Therefore, w has not yet been added to the BFS spanning tree T after the vertices at level ℓ have been added. Since w is adjacent to w', it will be added at level $\ell + 1$ (although the edge connecting w' and w is not necessarily added). This completes the proof by induction.

Now let $v \in V$ and let $D = (V, E_D)$ be a DFS spanning tree for G rooted at v, with height d. Similarly, let $B = (V, E_B)$ be a BFS spanning tree for G rooted at v, with height b, and suppose that $d < b$. Let w be any vertex in B at level b, if $L(w)$ is the length of the path from the root v to w in the DFS spanning tree D, then $L(w) \leq d < b$, which contradicts the fact that b is the length of a shortest path in G from v to w. Therefore we must have $b \leq d$ if the DFS and BFS trees are both rooted at v.

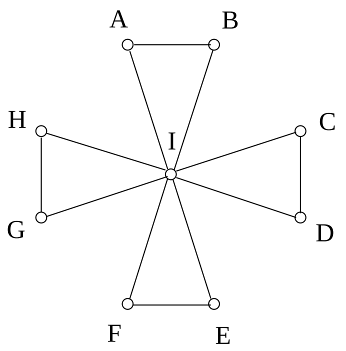
(e) In the graph G below, the height of a DFS spanning tree rooted at A is 1, while the height of a BFS spanning tree rooted at B is 2.

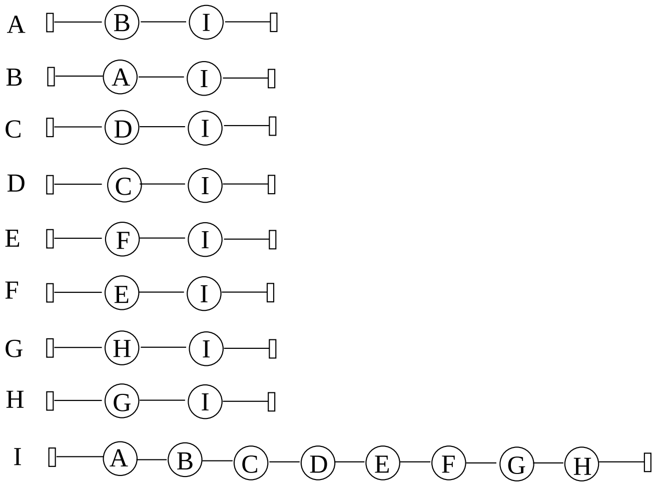
G :



4. *Depth First Search and Breadth First Search Trees*

Consider the graph G below with the adjacency lists shown.



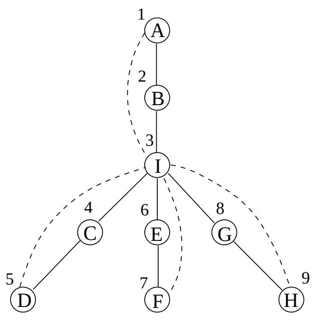


(a) Draw the DFS tree for this graph when the depth first search algorithm is executed with this adjacency list structure as input. Show the back edges also as dotted lines.

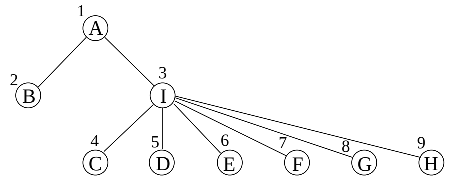
(b) Draw the BFS tree for this graph when the breadth first search algorithm is executed with this adjacency list structure. Also, show the contents of the queue for the queue-based implementation of the algorithm.

SOLUTION:

(a) The depth first search tree is shown below, the dfs numbers, that is, the order in which the nodes are visited by the DFS algorithm, are also shown.



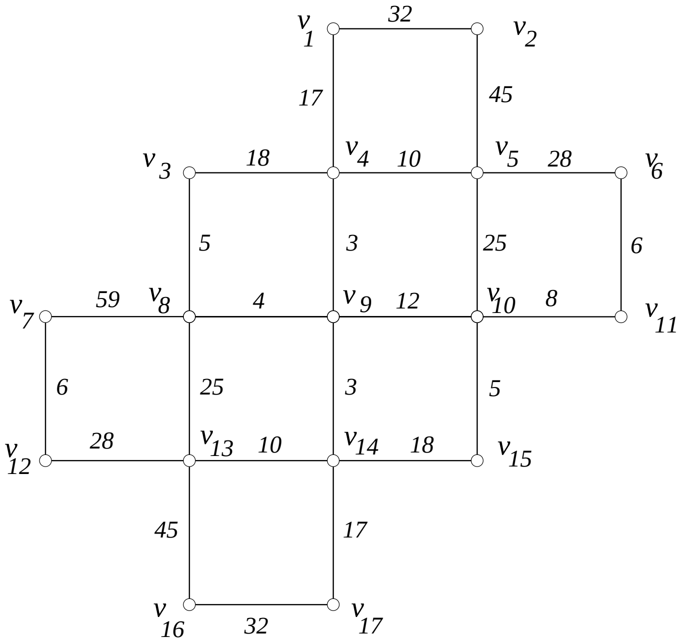
(b) The breadth first search tree is shown below, the contents of the queue as the BFS algorithm is executed are also shown.



A								
	B	I						
		I						
			C	D	E	F	G	H
				D	E	F	G	H
					E	F	G	H
						F	G	H
							G	H
								H

5. *Kruskal's Algorithm for Minimal Spanning Trees*

Use Kruskal's algorithm to find a minimal spanning tree for the weighted graph shown below.



What is the weight $W(T)$ of a minimal spanning tree?

Solution: A minimal spanning tree generated by Kruskal's algorithm is shown below and has weight $W(T) = 198$.