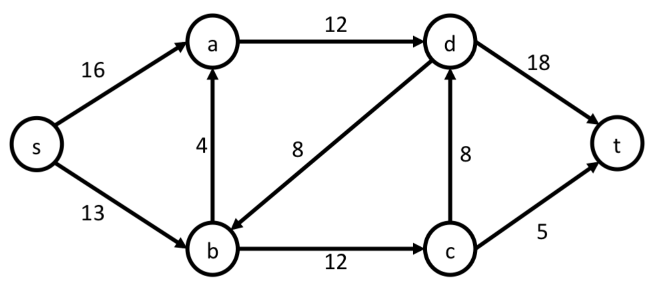
- The instructions are the same as in Homework-0, 1, 2, 3, 4. This is an **optional** homework. Even though we will not grade your submission, we will encourage you to attempt the homework and submit. Note that Network Flow in included in the syllabus for the Final Exam.

There are 5 questions for a total of 100 points.

---

1. (20 points) Consider the network shown in the figure. Consider running the Ford-Fulkerson algorithm on this network.



(a) We start with a zero s-t flow f. The algorithm then finds an augmenting path in G_f. Suppose the augmenting path is $s \to b \to c \to t$. Give the flow f' after augmenting flow along this path.

(b) Show the graph $G_{f'}$. That is, the residual graph with respect to s-t flow f'.

(c) The algorithm then sets f as f' and G_f as $G_{f'}$ and repeats. Suppose the augmenting path chosen in the next iteration of the while loop is $s \to a \to d \to t$. Give f' after augmenting flow along this path and show $G_{f'}$.

(d) Let f be the flow when the algorithm terminates. Give the flow f and draw the residual graph G_f.

(e) Give the value of the flow f when the algorithm terminates. Let A^* be the vertices reachable (using edges of positive weight) from s in G_f and let B^* be the remaining vertices. Give A^* and B^*. Give the capacity of the cut (A^*, B^*).

2. Answer the following:

(a) (1 point) <u>State true or false</u>: For every s-t network graph G, there is a unique s-t cut with minimum capacity.

(b) (4 points) Give reason for your answer to part (a).

(c) (1 point) <u>State true or false</u>: For every s-t network graph G and any edge e in the graph G, increasing the capacity of e increases the value of maximum flow.

(d) (4 points) Give reason for your answer to part (c).

(e) (1 point) <u>State true or false</u>: For every s-t network graph G for which an s-t flow with non-zero value exists, there exists an edge e in the graph such that decreasing the capacity of e decreases the value of maximum flow.

(f) (4 points) Give reason for your answer to part (e).

3. Suppose you are given a bipartite graph (L, R, E), where L denotes the vertices on the left, R denotes the vertices on the right and E denote the set of edges. Furthermore it is given that degree of every vertex is exactly d (you may assume that $d > 0$). We will construct a flow network G using this bipartite graph in the following manner: G has $|L| + |R| + 2$ vertices. There is a vertex corresponding to every vertex in L and R. There is also a source vertex s and a sink vertex t. There are directed edges with weight 1 from s to all vertices in L and directed edges of weight 1 from all vertices in R to t. For each edge $(u, v) \in E$, there is a directed edge from u to v with weight 1 in G.

   (*The figure below shows an example of a bipartite graph and the construction of the network.*)

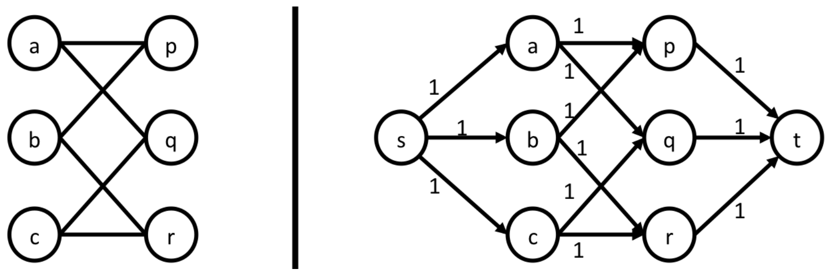


Figure 1: An example bipartite graph (with $d = 2$) and network construction.

   (a) (5 points) Argue that for any such bipartite graph where the degree of every vertex is equal to d, $|L|$ is equal to $|R|$.

   (b) (10 points) Argue that for any given bipartite graph where the degree of every vertex is the same non-zero value d, there is an integer s-t flow (i,e., flow along any edge is an integer) in the corresponding network with value $|L|$.

   (c) (5 points) A matching in a bipartite graph $G = (L, R, E)$ is a subset of edges $S \subseteq E$ such that for every vertex $v \in L \cup R$, v is present as an endpoint of at most one edge in S. A maximum matching is a matching of maximum cardinality. Show that the size of the maximum matching in any bipartite graph $G = (L, R, E)$ is the same as the maximum flow value in the corresponding network graph defined as above.

4. (20 points) There are n stationary mobile-phones $c_1, ..., c_n$ and n stationary mobile-phone towers $t_1, ..., t_n$. The distances between mobile-phones and towers are given to you in an $n \times n$ matrix d, where $d[i, j]$ denotes the distance between phone c_i and tower t_j. It is possible for a mobile-phone c_i to connect to a tower t_j if and only if the distance between c_i and t_j is at most D, where D is the connecting radius. Furthermore, at one time, a mobile-phone can connect to at most one tower and a tower can allow at most one connection. Your goal as a Communications Engineer is to figure out whether all mobile-phones are usable simultaneously. That is, whether it is possible for all mobile-phones to connect simultaneously to distinct towers. Answer the following questions.

   (a) Consider a simple example with 5 mobile-phones and 5 towers. Let the connecting radius be $D = 2$ miles. The distance matrix for this example is as given in Figure 2.

   Prove or disprove: It is possible for all 5 mobile-phones to simultaneously connect to distinct towers for this example.

   (b) Design an algorithm that takes inputs n, D, and the distance matrix d, and outputs "yes" if it is possible for all mobile-phones to simultaneously connect to distinct towers (within the connecting radius D) and "no" otherwise. Analyze the running time of the algorithm and give proof of correctness.

d	1	2	3	4	5
1	1	2	3	4	7
2	4	1	1	5	12
3	5	7	2	1	11
4	4	3	6	1	1
5	1	21	8	9	13

Figure 2: Distance matrix d for part (a) of question 4.

5. (25 points) Town authorities of a certain town are planning for an impending virus outbreak. They want to plan for panic buying and taking cues from some other towns, they know that one of the items that the town may run out of is toilet paper. They have asked your help to figure out whether the toilet paper demand of all n residents can be met. They provide you with the following information:

   - There are n residents $r_1, ..., r_n$, m stores $s_1, ..., s_m$, and p toilet paper suppliers $x_1, ..., x_p$.
   - The demand of each of the residents in terms of the number of rolls required.
   - The list of stores that each of the residents can visit and purchase rolls from. A store cannot put any restriction on the number of rolls a customer can purchase given that those many rolls are available at the store.
   - The number of rolls that supplier x_j can supply to store s_i for all $i \in \{1, ..., m\}$ and $j \in \{1, ..., p\}$.

   The above information is provided in the following data structures:

   - A 1-dimensional integer array $D[1...n]$ of size n, where $D[i]$ is the demand of resident r_i.
   - A 2-dimensional 0/1 array $V[1...n, 1...m]$ of size $n \times m$, where $V[i, j] = 1$ if resident r_i can visit store s_j and $V[i, j] = 0$ otherwise.
   - A 2-dimensional integer array $W[1...m, 1...p]$ of size $m \times p$, where $W[i, j]$ is the number of rolls of toilet paper that the supplier x_j can supply to store s_i.

   Design an algorithm to determine if the demand of all residents can be met. That is, given (n, m, p, D, V, W) as input, your algorithm should output "yes" if it is possible for all residents to obtain the required number of rolls and "no" otherwise. Argue correctness and discuss running time.

   (*For example, consider that the town has two residents, one store and two suppliers. If* $D = [2, 2], V = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$*, and* $W = [2, 2]$*, then the demand can be met. However, if* $D = [2, 2], V = \left[\begin{smallmatrix}1\\1\end{smallmatrix}\right]$*, and* $W = [2, 1]$*, then the demand cannot be met.*)