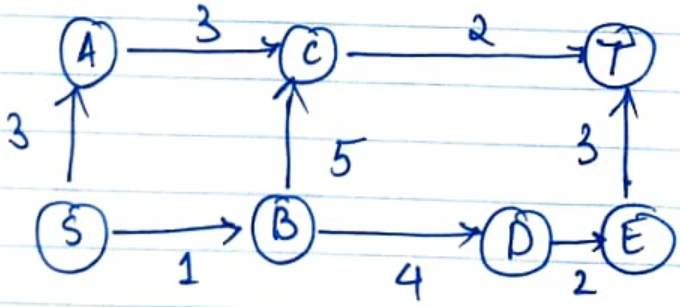
# HomeWork 8

Ques 1. The following graph G has labeled nodes and edges between it. Each edge is labeled with its capacity.

(a) Draw the final residual graph $G_f$ using the Ford-Fulkerson algorithm corresponding to the max flow. Please do NOT show all intermediate steps.

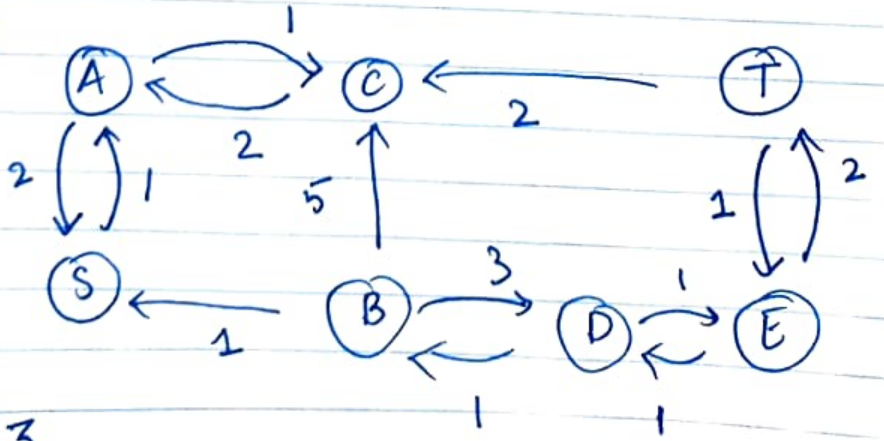
(b) What is the max-flow value?

(c) What is the min-cut?




Ans 1.

(a)




(b) 3

(c) {S, A, C} , {B, D, E, T}

Ques 2. Determine if the following statements are true or false. For each statement, briefly explain your reasoning.

(a) In a flow network, the value of flow from s to T can be higher than the maximum number of edge disjoint paths from s to T. (Edge disjoint paths are paths that do not share any edge)

(b) For a flow network, there always exists a maximum flow that doesn't include a cycle containing positive flow.

(c) If you have non-integer edge capacities, then you cannot have an integer max flow.

(d) Suppose the maximum s-t flow of a graph has value f. Now we increase the capacity of every edge by 1. Then the maximum s-t flow in this modified graph will have a value of at most f+1.

(e) If all edges are multiplied by a positive number k, then the min-cut remains unchanged.

Ans 2. (a) False
(b) False
(c) True
(d) False
(e) True

Ques 3. You are given a flow network with unit-capacity edges. It consists of a directed graph $G=(V,E)$ with source $s$ and sink $t$, and $u_e=1$ for every edge $e$. You are also given a positive integer parameter $k$. The goal is delete $k$ edges so as to reduce the maximum $s$-$t$ flow in $G$ by as much as possible. In other words, you should find a subset $F \subseteq E$ of edges such that $|F|=k$ and the maximum $s$-$t$ flow in the graph $G'=(V, E \setminus F)$ is as small as possible. Give a polynomial-time algorithm to solve this problem and briefly explain its correctness.

Follow up: If the edges have more than unit capacity, will your algorithm produce the smallest possible max-flow value.

Ans 3.

- If max-flow $\leq k$ :
  disconnect $s$ and $e$ after decreasing max-flow to 0 and remove all edges in that $s$-$t$ cut

  else if max-flow $> k$ :
  create new cut with max-flow $-k$ edges and remove $k$ edges from max-flow

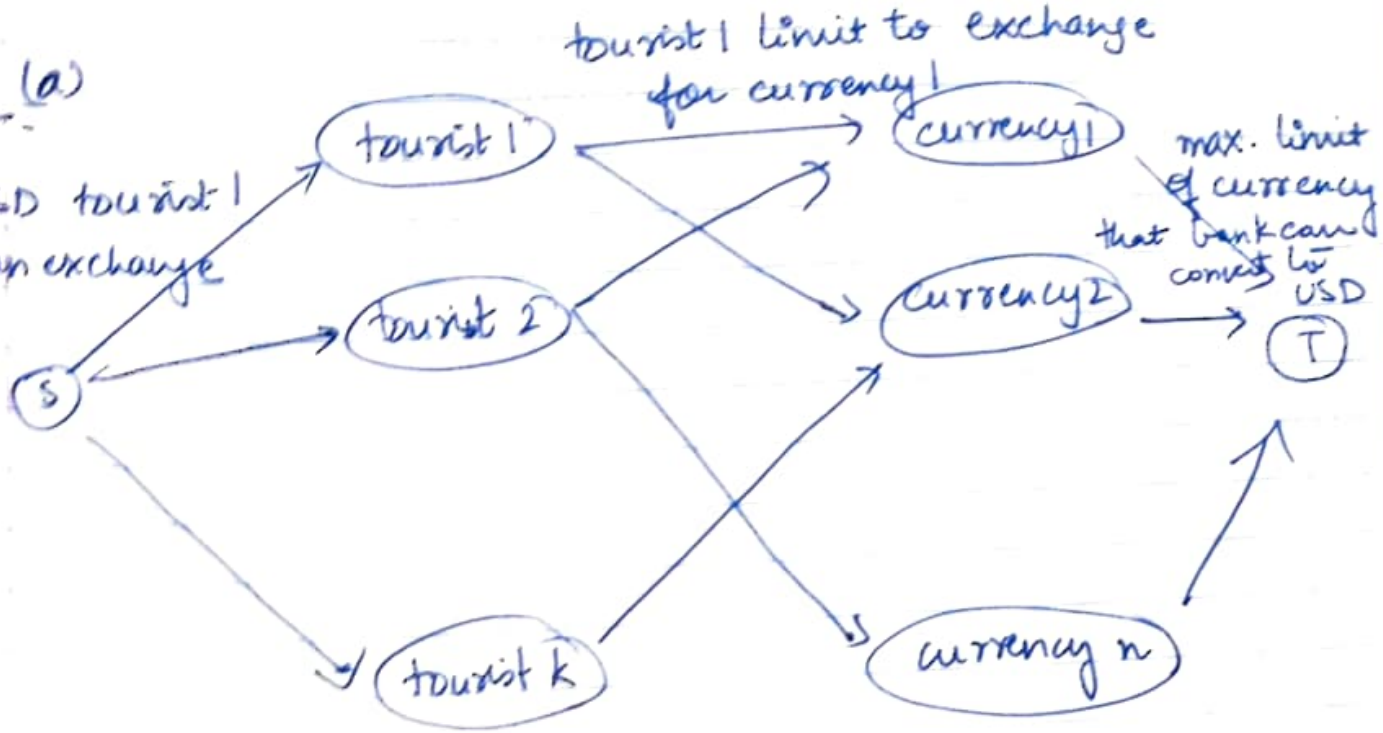
- If the edges have more than unit capacity, removing $k$ edges from min cut in the above way does not guarantee to have minimum possible max flow.

Ques. A tourist group needs to convert their USD into various International currencies. There are $n$ tourists $t_1, t_2, \ldots, t_n$ and $m$ currencies $c_1, c_2, \ldots, c_m$. Each tourist $t_k$ has $F_k$ Dollars to convert. For each currency $c_j$, the bank can convert at most $B_j$ Dollars to $c_j$. Tourist $t_k$ is willing to trade as much as $S_{kj}$ of his Dollars for currency $c_j$. (For example, a tourist with 1000 Dollars might be willing to convert up to 200 of his USD for Rupees, up to 500 of his USD for Japanese Yen and up to 300 of his USD for Euros). Assume that all tourists give their requests to the bank at the same time.

(a) Design an algorithm that the bank can use to satisfy all the requests (if it is possible). To do this, construct and draw a network flow graph, with appropriate source and sink nodes, and edge capacities.

(b) Prove your algorithm is correct by making a claim and proving it in both directions.

Ans 4. (a)




Run Ford-Fulkerson or any Network Flow from S to T.

(b)

Claim - The problem has solution if there is a max flow through the constructed graph and all tourists are able to exchange their specified USD.

Proof -

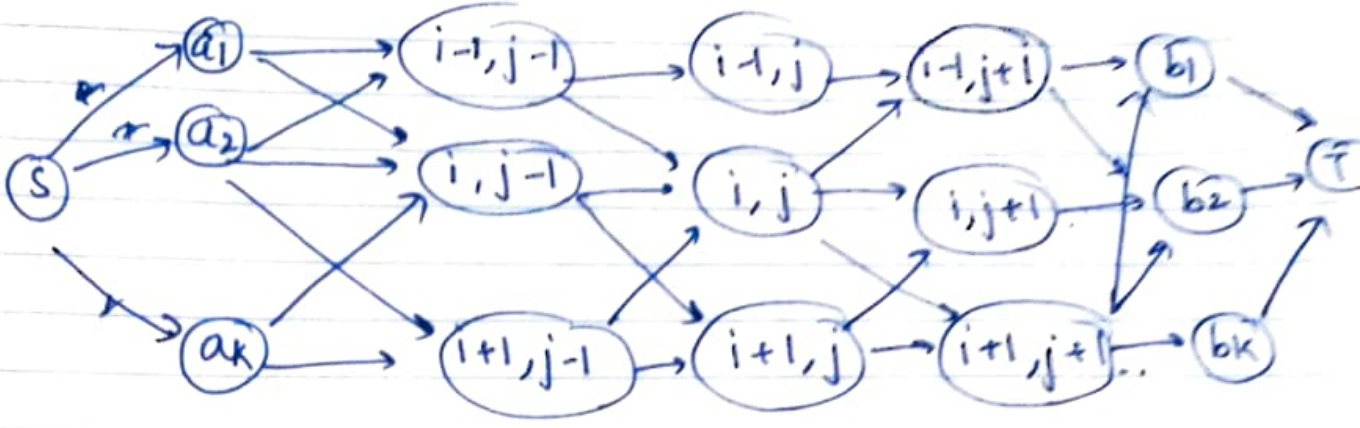
- Let us assume that there is a valid assignment such that the bank can satisfy all the requests of tourists for conversion.
  This means all outgoing edges from source s to each tourist i is saturated and hence we get max flow in the graph from s to T.
- Let us assume a max-flow exists then from the graph it means that all edges are saturated and all tourists were able to convert their currencies.

Ques 5. USC admissions Center needs your help in planning paths for campus tours given to prospective students or interested groups. Let USC campus be modeled as a weighted, directed graph G containing locations V connected by one-way roads E. On a busy day, let K be the number of campus tours do not use the same road. Let the tour have K starting location $A = \{a_1, a_2, \ldots, a_K\} \subseteq V$. From the starting location, the groups are taken by a guide on a path through G to some ending location in $B = \{b_1, b_2, \ldots, b_K\} \subseteq V$. Your goal is to find a path for each group i from the starting location, $a_i$, to any ending location $b_j$ such that no two paths share any edges, and no two groups end on the same location $b_j$.

(a) Design an algorithm to find k paths $a_i \to b_j$ that start and end at different vertices, such that they do not share any edges.

(b) Modify your algorithm to find k paths $a_j \to b_j$ that start and end in different locations, such that they do not share any edges or vertices.

(a)



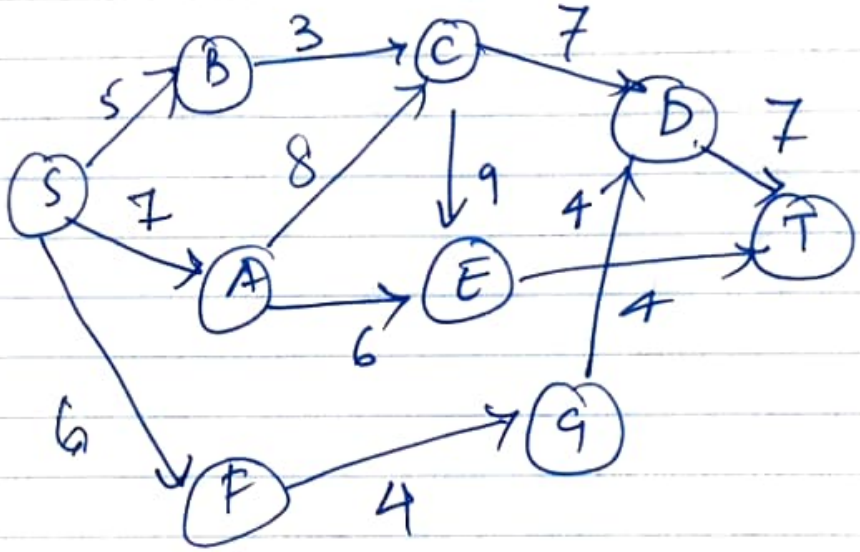

(i,j) denotes location of each building that do not belong to A or B.

For all (i,j) we add directed edges to next nodes (i,j) connected to (i-1, j+1), (i, j+1) and (i+1, j+1)

Add a source node s.

Run Ford Fulkerson algorithm on the above graph to get value of max flow which is k.

Ques 6. Perform two iterations (i.e. two augmentation steps) of the scaled version of the Ford-Fulkerson algorithm on the flow network given below. You need to show the value of $\Delta$ and the augmentation path for each iteration, and the flow $f$ and $G_f(\Delta)$ after each iteration. (Note: iterations may or may not belong to the same scaling phase)



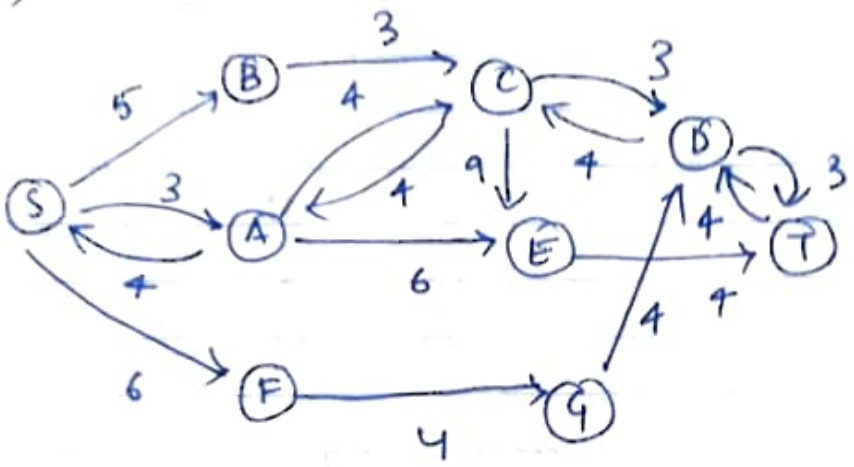

(a) i. Give the value of $\Delta$ and the augmentation path.
ii. Show the flow after the first iteration.
iii. Show the $G_f(\Delta)$ after the first iteration.

(b) i. Give the value of $\Delta$ and the augmentation path.
ii. Show the flow after the second iteration
iii. Show the $G_f(\Delta)$ after the second iteration

(c) Can the choice of augmentation paths in the scaled version of Ford-Fulkerson affect the no. of iteration? Explain why

Ans 6.

1) $\Delta = 4$




2) $\Delta = 2$