The Maximum Flow Problem :

Problem statement:

- Problem of maximizing the flow of a material through a transportation network.
  Eg: pipeline system, communications or transportation networks.

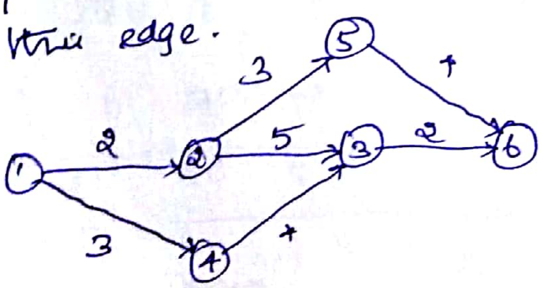
- Formally represented by a connected weighted digraph with n vertices numbered from 1 to n and a set of edges with following properties.

- source:
  It contains exactly one vertex with no entering edges and assumed to be numbered 1

- sink:
  It exactly one vertex with no leaving edges and assumed to be numbered n.

- capacity:
  The weight u_{ij} of each directed edge (i, j) is a positive integer, called the edge capacity.
  - This number represents the upper bound on the amount of the material that can be sent from i to j through a link through this edge.

Example:



Flow:

A flow is an assignment of real numbers x_{ij} to edge (i, j) of a network that satisfy the following:

- Flow - Conservation requirements

$$\sum_{j:(j,i)\in E} x_{ji} = \sum_{j:(i,j)\in E} x_{ij} \quad \text{for } i = 2, 3, \dots n-1$$

The total amount of material entering an intermediate vertex must be equal to the total amount of the material leaving the vertex.

- Capacity constraints:

$$0 \leq x_{ij} \leq u_{ij} \quad \text{for every edge } (i, j) \in E$$

Since no material can be lost or added to by going through intermediate vertices of the network, the total amount of the material leaving the source must end up at the sink.

$$\sum x_{1j} = \sum x_{jn}$$

- The value of the flow is defined as the total outflow from the source = the total inflow into the sink.

- The maximum flow problem is to find a flow of the largest value (maximum flow) for a given network.

- Skew symmetry:

For all $u, v \in f(u, v) = -f(v, u)$ that means reverse edge.

17-10-17

The Ford Fulkerson Method:

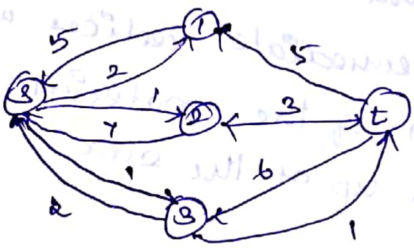
- This algorithm works for solving maximum flow problem.
- Residual network
- Augmenting path.
- Cuts:

Residual Network:

- A flow network $G = (V, E)$ with source s and sink t.
- Let every edge u, v is having a pair flow/capacity then the representation of a graph with residual capacity is called residual network.

$$c_f(u,v) = c(u,v) - f(u,v)$$

Eg:



$c_f(u,v) = c(u,v) - f(u,v)$

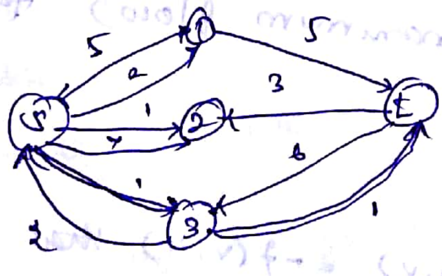
$= 7 - 5$

$= 2$

augmenting path:

- An augmenting path is a simple path p from s to t in residual network G_f.
- This is the path which never violates the capacity constraints.

Eg:



- The graph is in maximum flow if there is no augmenting path.

17-10-17 1.

Cuts in flow Networks:

A cut $C = (S, T)$ is a partition of V of a graph $G = (V, E)$ into two subsets S and T. The cut set of a cut $C = (S, T)$ is the set of edges that have one endpoint in S and the other endpoint in T.

- If s and t are specified vertices of the graph G, then an s-t cut is a cut in which s belongs to the set S and t belongs to the set T.

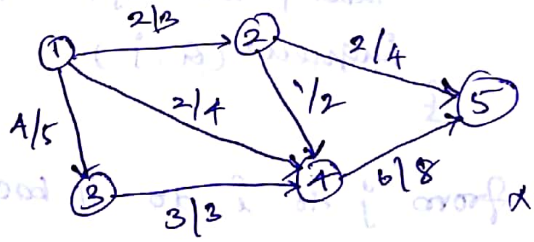
Minimum cut:

A cut is minimum if the size or the weight of the cut is not larger than the size of any other cut.

$X = \{1, 2, 3, 4\}$ $\bar{X} = \{5, 6\}$

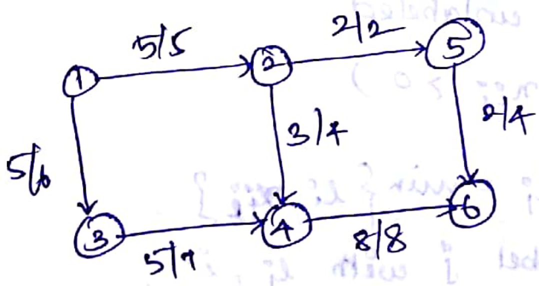
$C = \{(2,5)\ (4,6)\} = \underline{10}$

Eg:



$X = \{1\}$ $\bar{X} = \{2, 3, 4, 5, 6\}$

$C = \{(1,2)\ (1,3)\} = 11$



$X = \{1, 2\}$ $\bar{X} = \{3, 4, 5, 6\}$

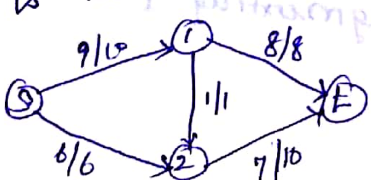
$C = \{(1,3)\ (2,4)\ (2,5)\} = 12$

$X = \{1, 3\}$ $\bar{X} = \{2, 4, 5, 6\}$

$C = \{(1,2)\ (3,4)\} = 13$

Maximum cut:

A cut is maximum if the size of the cut is not smaller than the size of any other cut



$\bar{X} = \{1, 2, 5\}$ $X = \{3, 4, 6\}$

$C = \{(1,3)\ (2,4), (5,6), (4,6)\}$

$C = \{(5,6)\ (4,t)\} = \cancel{12}\ 22$

12-10-17

Algorithm ShortestAugmentingPath (G)

// i/p : A N/w with single source 1, single sink n, & +ve integer capacity u_{ij} on its edges (i, j)

// o/p : A maximum flow x.

{

assign $x_{ij} = 0$ to every edge (i, j) in the N/w

label the source with ∞, – and add the source to the empty queue Q.

while not empty(Q) do

{

i ← Front(Q) ; Dequeue(Q)

for every edge from i to j do // forward edges

if j is unlabeled

{

$r_{ij} = u_{ij} - x_{ij}$;

if ($r_{ij} > 0$)

{ $l_j = \min\{l_i, r_{ij}\}$;

label j with l_j, i+ ;

Enqueue (Q, j)

} }

for every edge from j to i do // backward edges

if j is unlabeled

{ if ($x_{ji} > 0$)

{ $l_j = \min\{l_i, x_{ji}\}$;

label j with l_j, i– ;

Enqueue (Q, j)

} }

if the sink has been labeled.

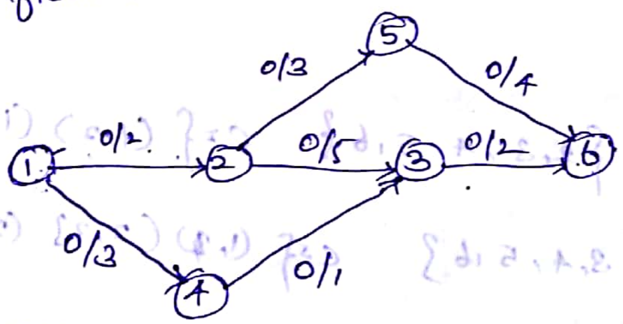
// augment along the augmenting path found.

17-10-17

$j = n$ // start at sink and move backwards using second labels.

while ($j \neq 1$) // the source hasn't been reached
{
  if the second label of vertex j is i^+
    $x_{ij} = x_{ij} + l_n$
  else // the second label of vertex j is i^-
    $x_{ji} = x_{ji} - l_n$
  $j = i$;
  i = the vertex indicated by i's second label
}
Erase all vertex labels except the ~~...~~ source
reinitialize Q with the source
}
return (x) // the current flow is maximum
}

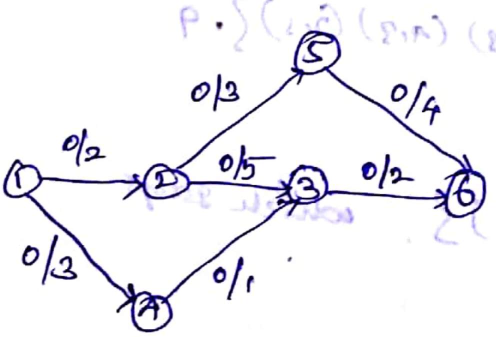
① Apply the shortest augmenting path algorithm to find a maximum flow and a minimum cut in the following N/w

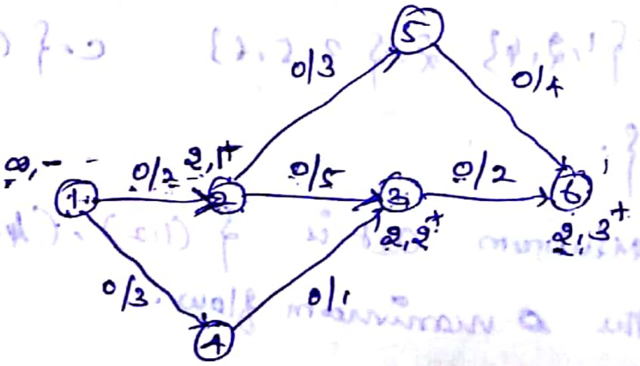


$X = \{1,2,3\}, \bar{X} = \{4,5,6\}$

$= 3+3+2 = 8$

step 1: consider the augmenting path 1-2-3-6

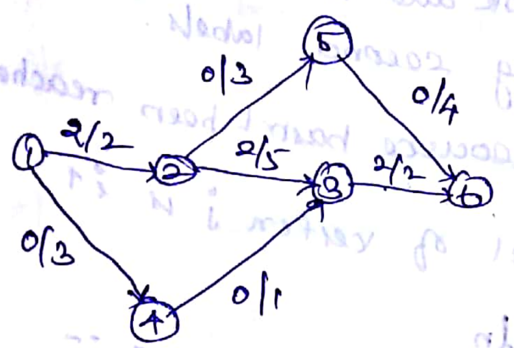


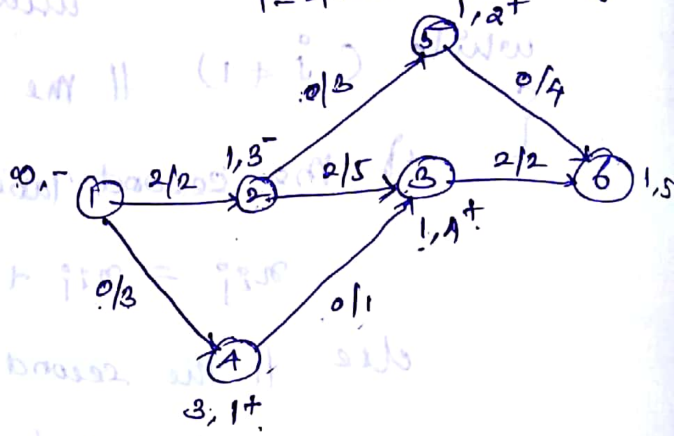


17-10-17

Step 2:

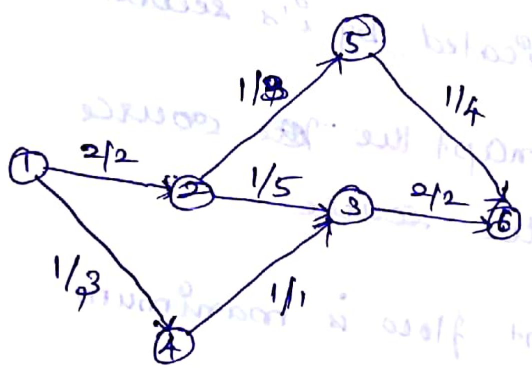
consider the augmenting path 1-4-3-2-5-6

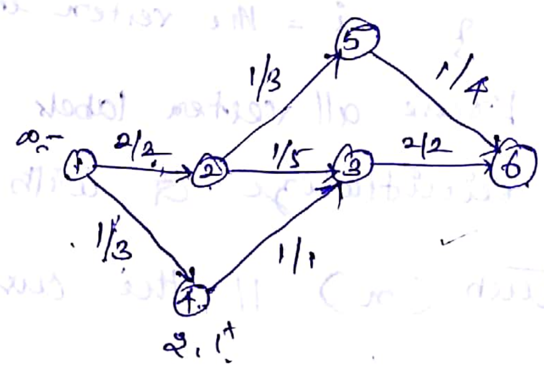




Step 3:

Consider the 1-4-6





No augmenting path.

Hence the current flow is maximal.

Minimum cut:

$X = \{1\}$ $\bar{X} = \{2,3,4,5,6\}$ $C = \{(1,2)\ (1,4)\} = 5$

$X = \{1,2\}$ $\bar{X} = \{3,4,5,6\}$ $C = \{(1,4)\ (2,3)\ (2,5)\} = 11$

$X = \{1,4\}$ $\bar{X} = \{2,3,5,6\}$ $C = \{(1,2), (1,4)\} = 3$

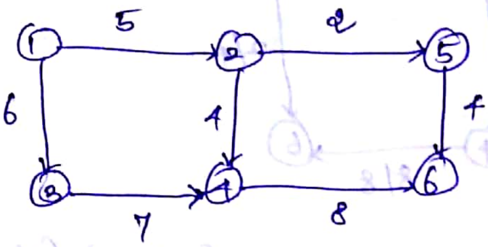
$X = \{1,2,4\}$ $\bar{X} = \{3,5,6\}$ $C = \{(2,3)\ (4,3)\ (2,5)\} = 9$

$X = \{$

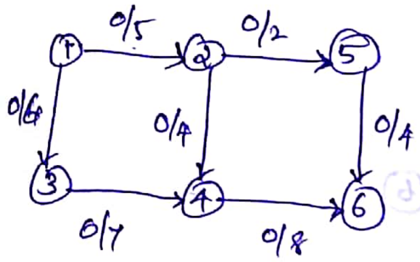
Minimum cut is $\{(1,2), (4,3)\}$ which says the maximum flow.

17-10-17

② Apply the shortest-augmenting path algorithm to find a maximum flow and a minimum cut in the following networks.



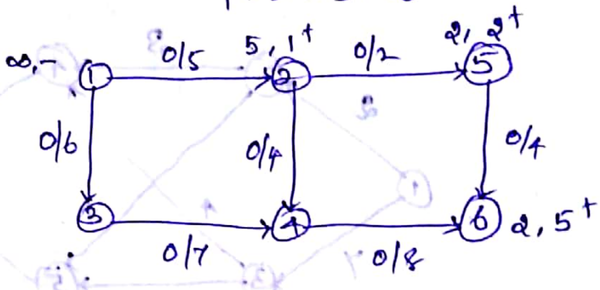
Step 1:



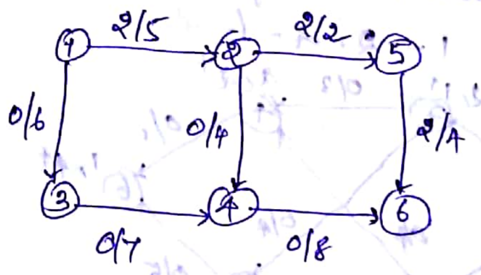
⇒

consider the augmenting path.

1-2-5-6



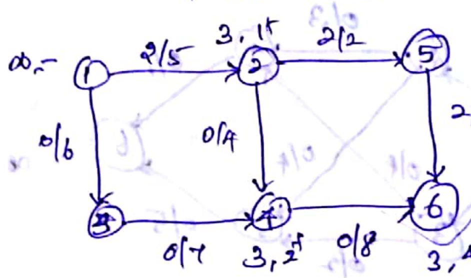
Step 2:



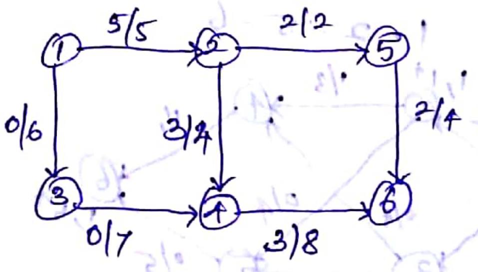
⇒

Consider the augmenting path

1-2-4-6



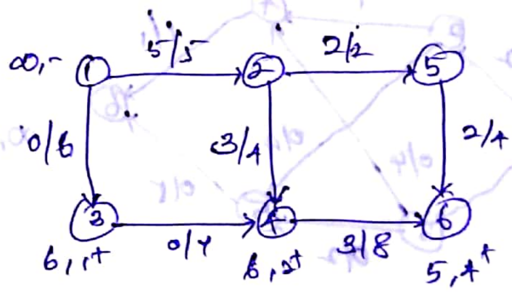
Step 3:



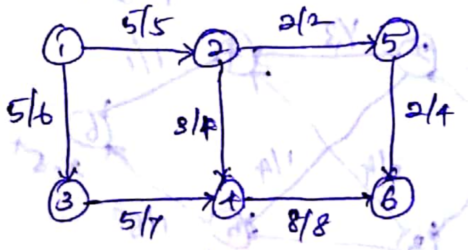
⇒

Consider the augmenting path.

1-3-4-6

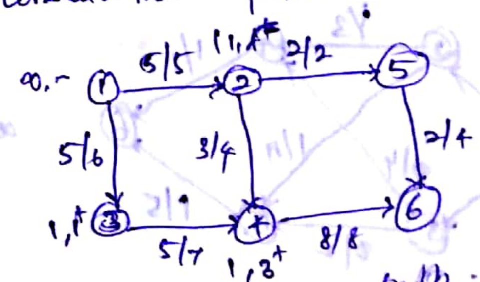


Step 4:



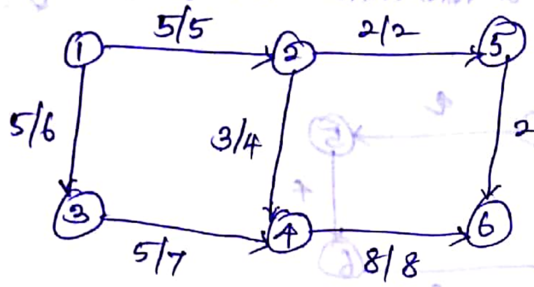
⇒

consider the path:



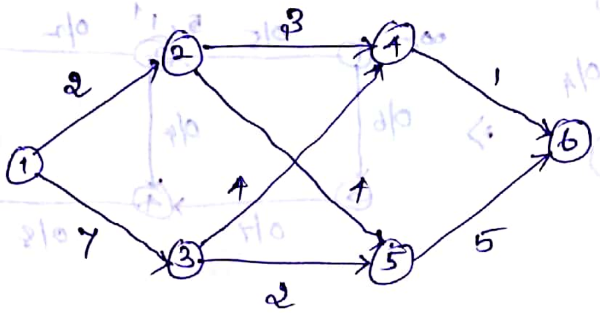
No augmenting path.

The graph has maximum flow.

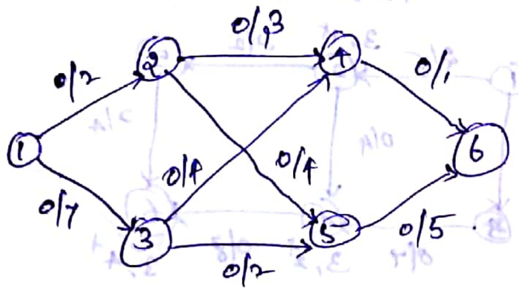


- The minimum cut is $C = \{(2,5)\ (4,6)\}$

(3)

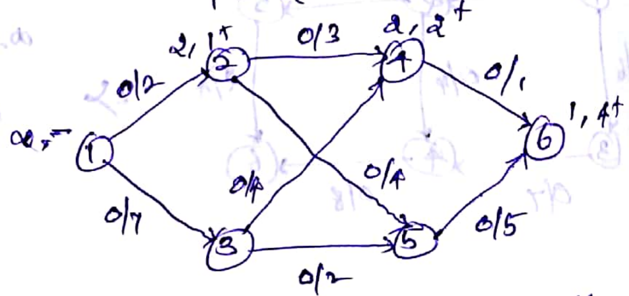


Step 1:

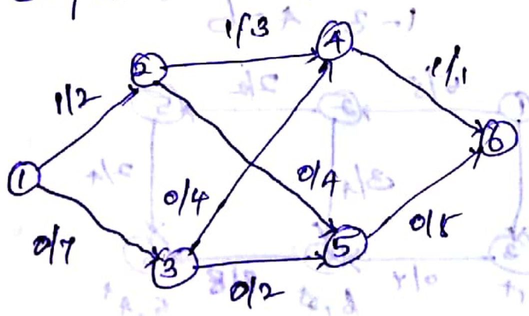


Consider the augmenting path.

1 - 2 - 4 - 6

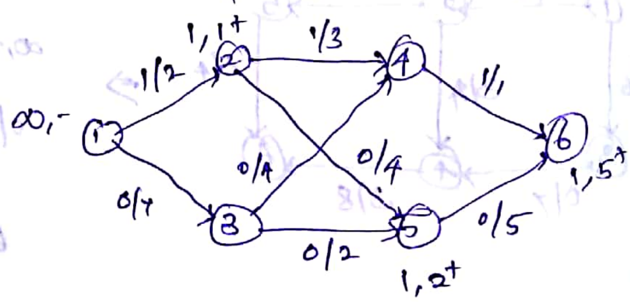


Step 2:

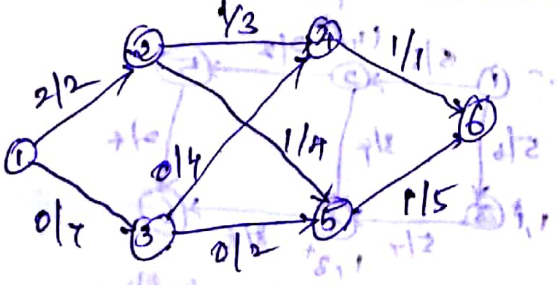


Consider the augmenting path

1 - 2 - 5 - 6

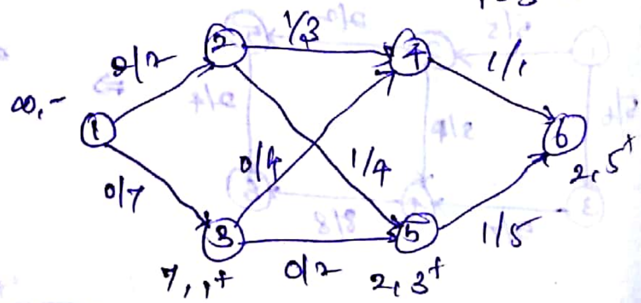


Step 3:



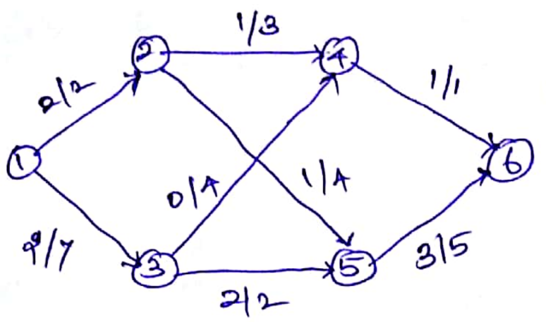
Consider the aug path.

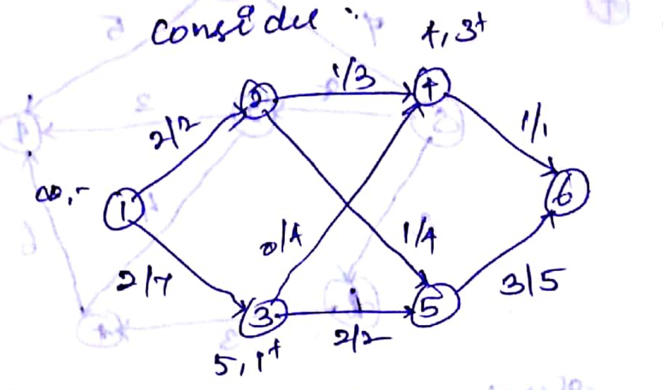
1 - 3 - 5 - 6



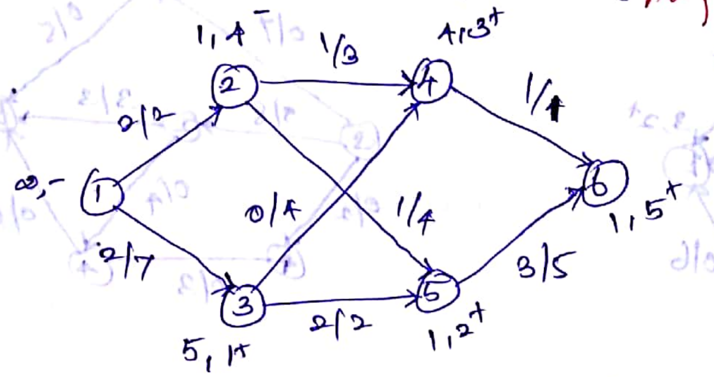
step 4:

Consider

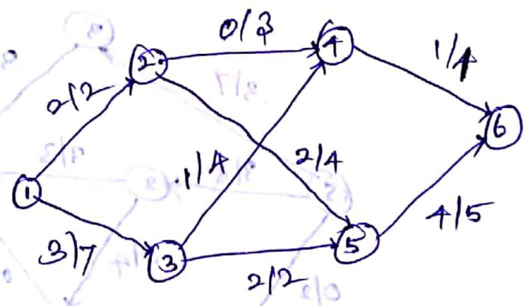


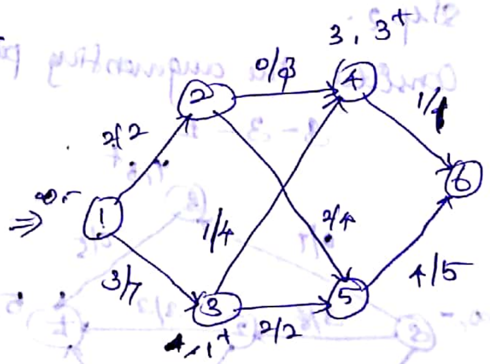


✓ ~~There is no augmenting path~~ Aug. path 1-3-4-2-5-6



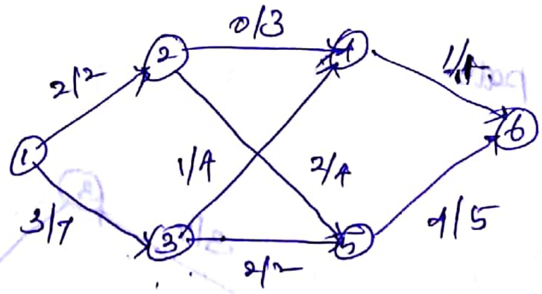
✓ ~~The minimum cut of~~





✓ There is No augmenting path

✓



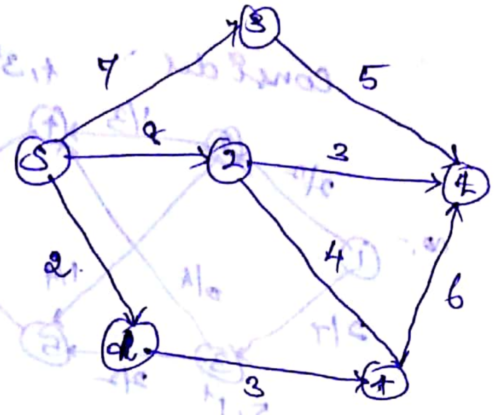
– The graph has maximum flow.

✓ The minimum cut = {(1,2), (4,6) (3,5)}

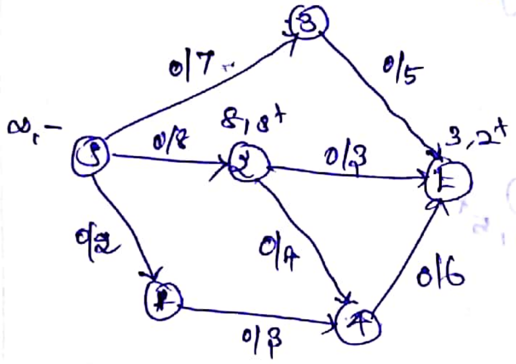
– The set of vertices that can be reached from source are 3, 4 $X = \{1, 3, 4\}$ $\bar{X} = \{2, 5, 6\}$

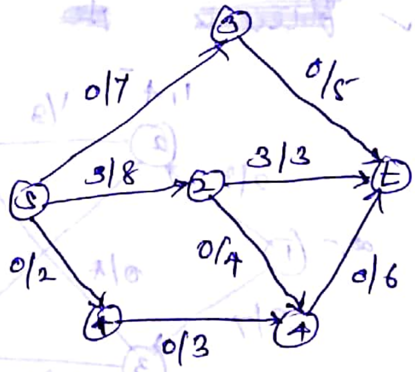
minimum cut = {(1,2) (3,5) (4,6)} = 2+2+1 = 5

④

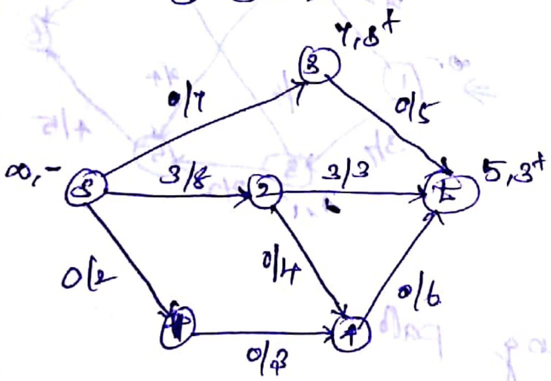


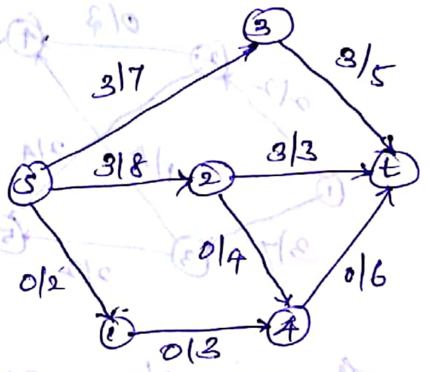
Step 1: Initially flow on each edge is 0:
Consider the augmenting path s-2-t



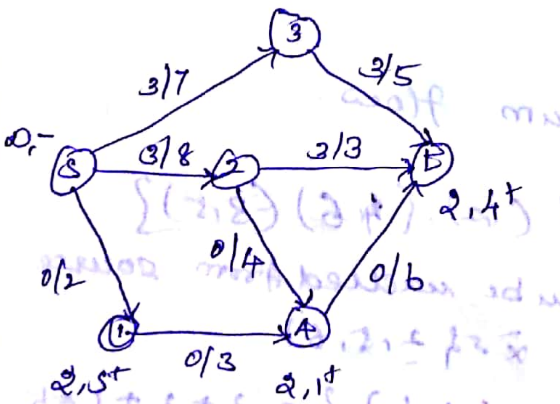


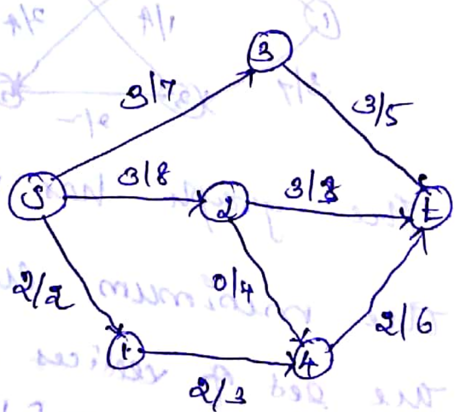
Step 2:
Consider the augmenting path
s-3-t





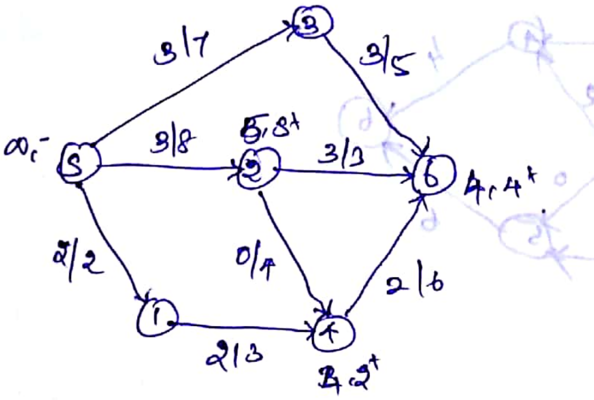
Step 3.
Consider the augmenting path
s - 1 - 4 - 6

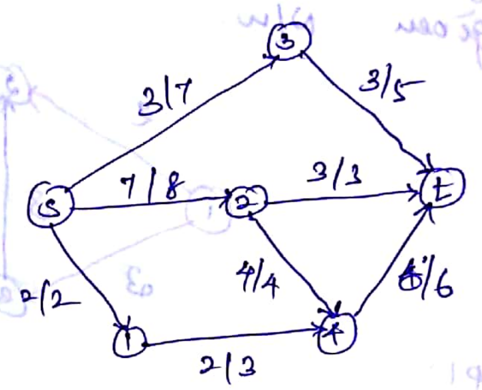




Step 4:

Consider the augmenting path- s-2-4-6

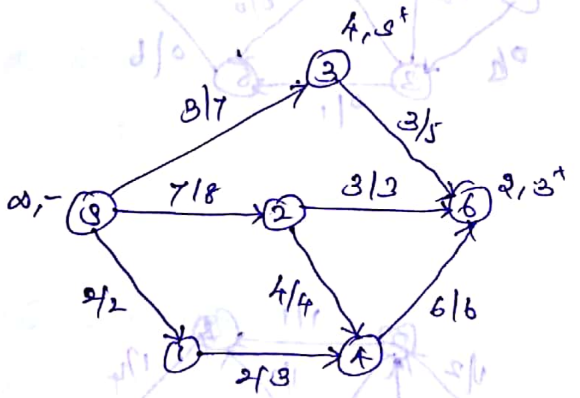


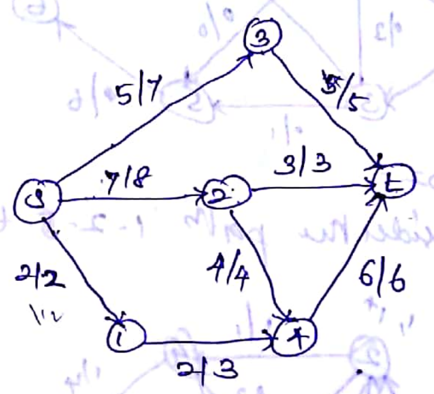


Steps:

Consider the augmenting path

s - 3 - t





- There is no augmenting path. S, 3
- Hence graph has maximum flow.
- The minimum cut c = {(3,t) (2,t), (4,t)}

Max

(S, 3, 2) (1, 3, 4, t)

(S, 1) (S, 2)

Algorithm Analysis:

- time complexity $O(EF^*)$

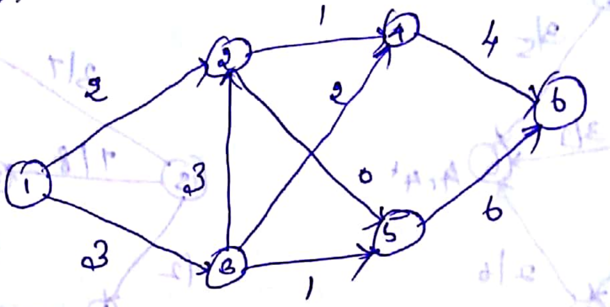
- The set of vertices that can be reached from source but does not include sink are: 2, 3,

$\bar{X} = \{s, 2, 3\}$ $X = \{1, 4, t\}$

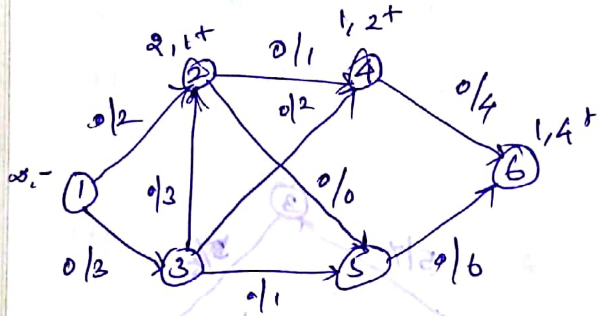
cut = {(s,1) (2,4) (2,t) (3,t)} = 2+4+3+5

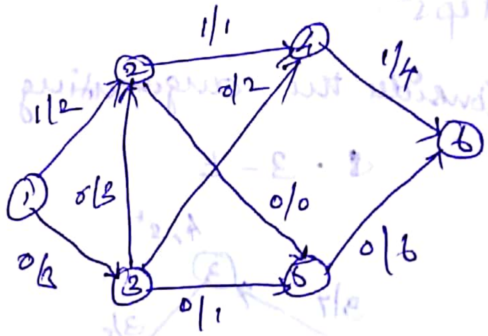
= 14

(5) Illustrate pictorially the Ford-Fulkerson method by showing the flow augmenting paths for the given N/w.

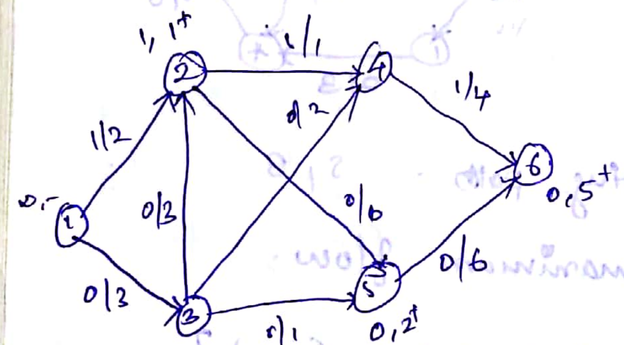


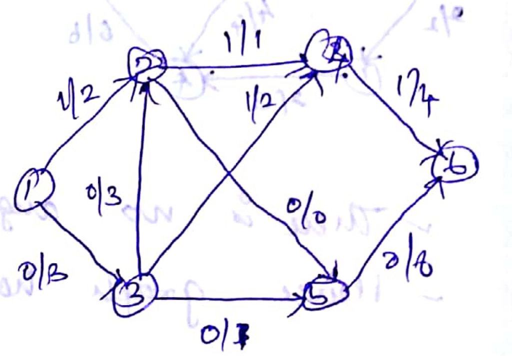
Step 1:
Consider the path.
1 - 2 - ~~3~~ - 4 - ~~5~~ - 6



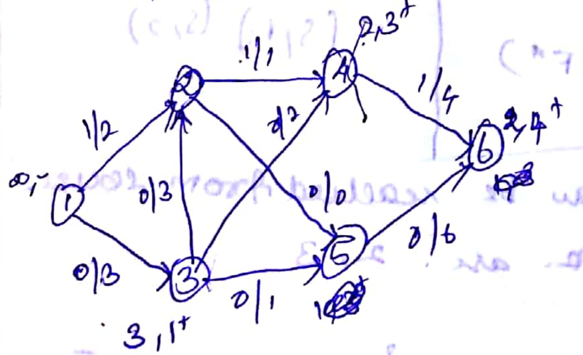


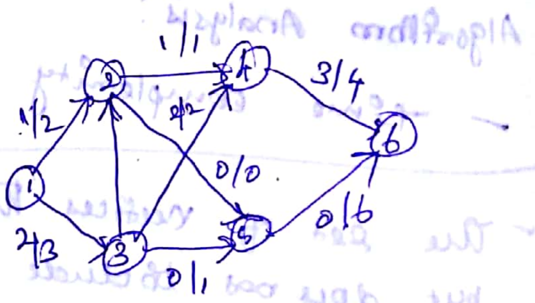
Step 2:
Consider the path 1-2-5-6





Step 3:
Consider the path.
1-3-4-6



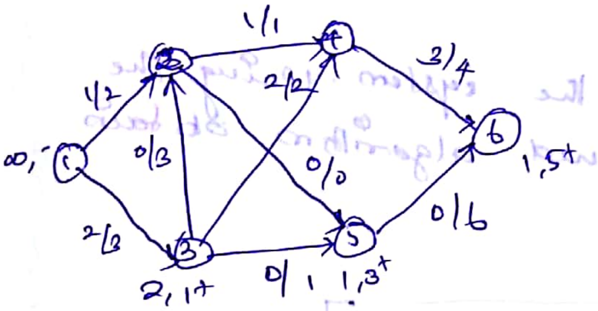


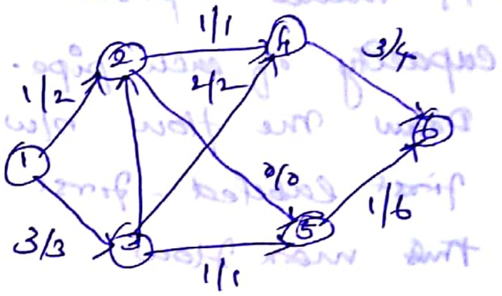
Scanned by CamScanner

step 4:

Consider the path.

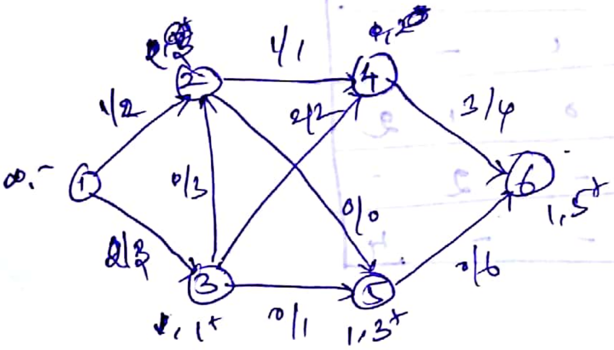
1-3-5-6

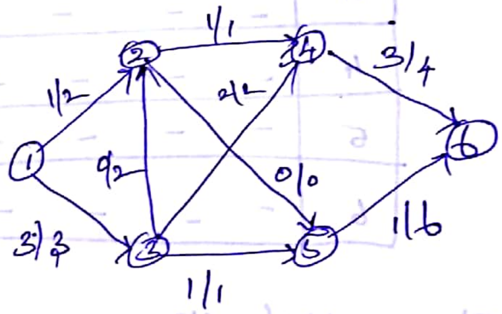




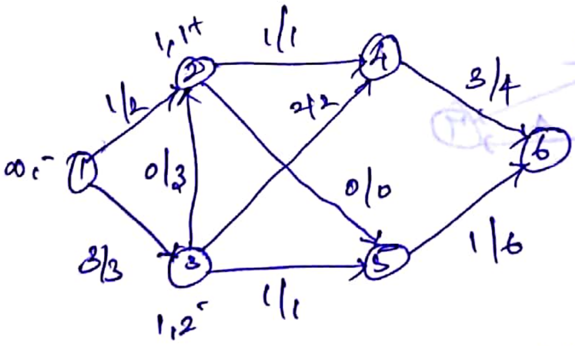
Consider the path.

1-2-3-5-6





Consider the path. 1 2-5-6



- There is no augmenting path.
- Hence the graph has maximum flow.
- the minimum cut is C = {(2,4) (2,5) (3,5)}
- The set of vertices reachable from source are.

2, 4 x = {1,2,3,} $\bar{x}$ = {3,4,5,6}

cut = {(2,4) (2,5) (3,4) (3,5)} = 4

which is max flow of N/W as well.