Ford Fulkerson Algorithame for Max Flow.

⇒

Find the maximum flow through the given network using ford fulkerson algorithame.

1, The ford fulkerson method is used for solving maximum flow problem.

2, Basic terms:

- source
- sink
- capicity and bottle neck capicity.
- flow
- Augmenting path.
- Residual capicity.

1, The source vertex has all outward edge, no inward edge.

2, Sink will have all inward edge, no outward edge.

Note: in Given graph v_1 is the source vertex.
while v_7 have an sink edge.

3, capicity :- weight of flow from one vertex to another.
while Bottle neck capicity ~~of the~~ we find for the capicity of the path is the minimum capicity of the ~~path~~ any edge on the path.

ex:- like in given graph 1 to 2 to it's weight is capicity same 2 to 5 weight is known as capicity.

→ Note we use Bottle kneck capicity for the path. like example we have chosen source n to n7 as a path.

so 1 to 2 and 2 to 5 and 5 to 7 vertex on that path 7, 5, 10 weight which will the minimum is known as bottle kneck capicity.

and 7, 5, and 10 so 5 is minimun known as bottle kneck capicity.

flow capicity

4, flow. 4/5

5, Augmenting path when we are finding capicity from 1 to 2, 2 to 5 and 5 to 7 our Bottle kneck capicity. so that whole path is called Augmenting path.

1 to 2, 2 to 5 and 5 to 7 is known as augmenting Path.

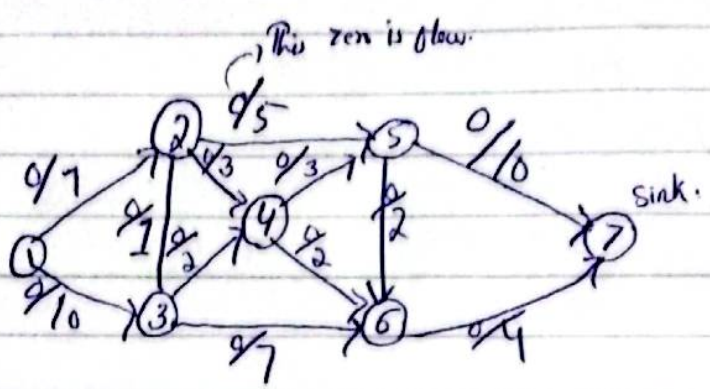
6, Residual capicity: Every edge of a residual graph has a value called residual capicity. which is eaual to orginal capicity minus current flow.

so, Residual capicity = Original capicity - current flow.

ex: 4/5 (flow / capacity)

5 − 4 = 1 → This will become residual capacity

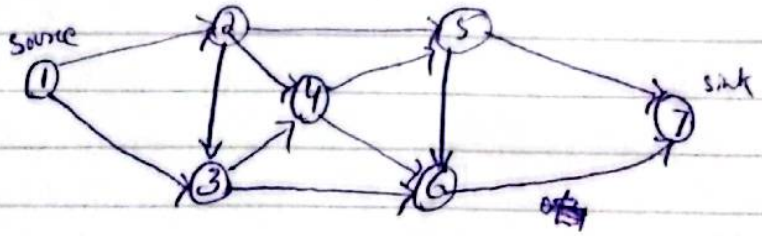
Let's take an example and calculate it:



step1: we have to calculate the flow and initial we have to make 0 (zero) to all.

step2: create another graph.

Solution



after making initially to zero Now take or select any path randomly. source to sink.

For solving it we have to make an table.

so we have select as an Augmenting path as a 1 − 2 − 5 − 7 so for that path we have to find Bottle neck capacity.

Like this:

Augmenting Path	Bottleneck capacity:
1 → 2 → 5 → 7	

so for given path 1 → 2 → 5 → 7 Bottleneck capacity will remain 7 or 5 and 10 weight 5 is the minimum, Bottleneck capacity.