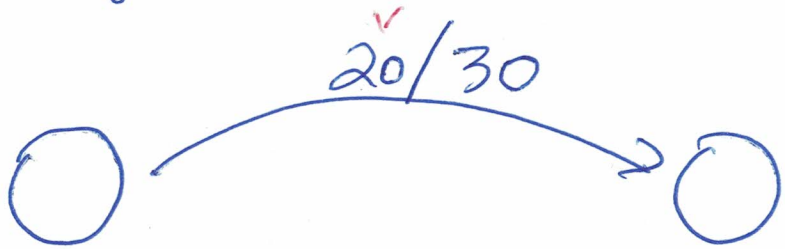
Cpts 515, 9/18/2020

Today: FF-Alg Examples.

When we assign a flow on an edge,

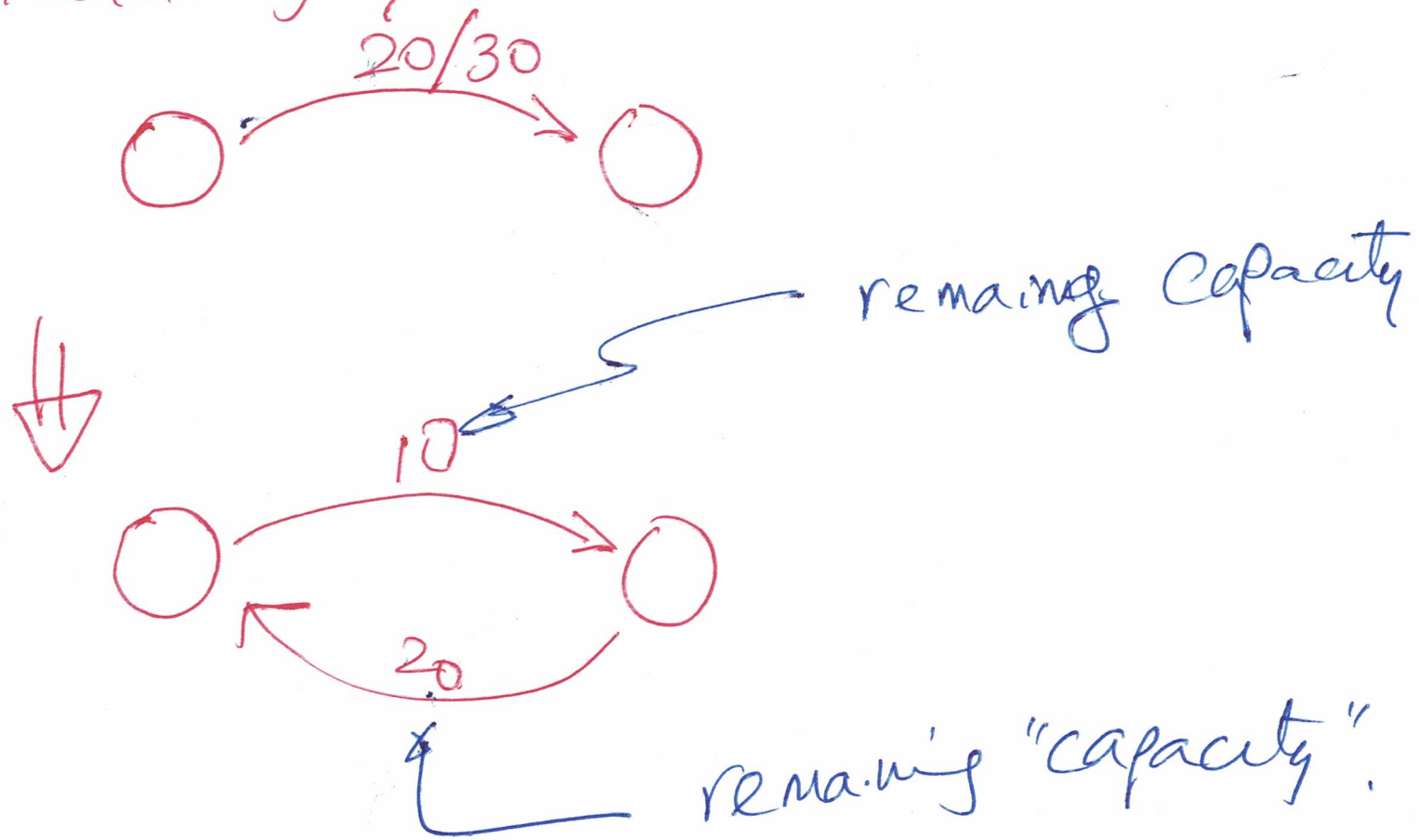


It is impossible to "decrease" the flow (such as the 20) to e.g., 18 (where the actual max is reached!)

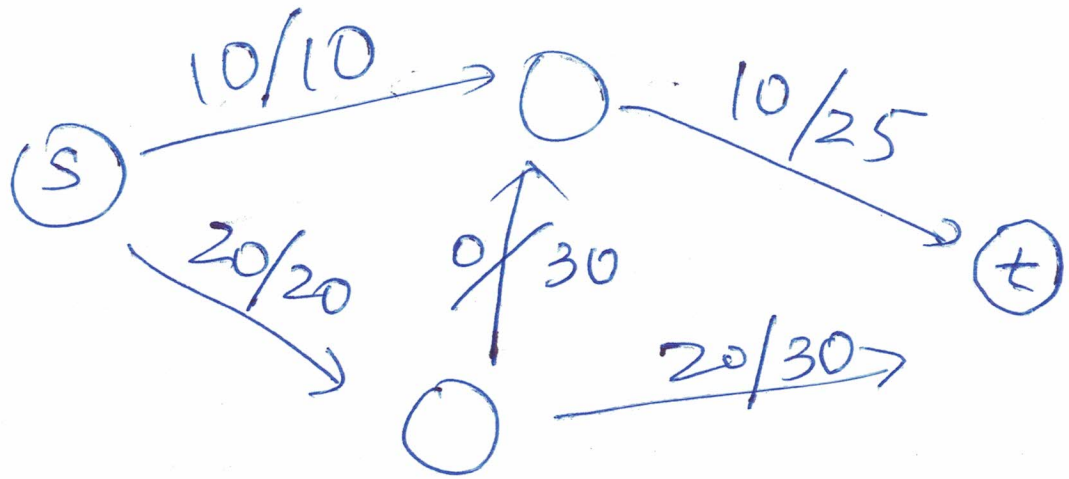
⇒ We need a mechanism (is still greedy) that can bring down 20 in a local

max to 18 is the global max.

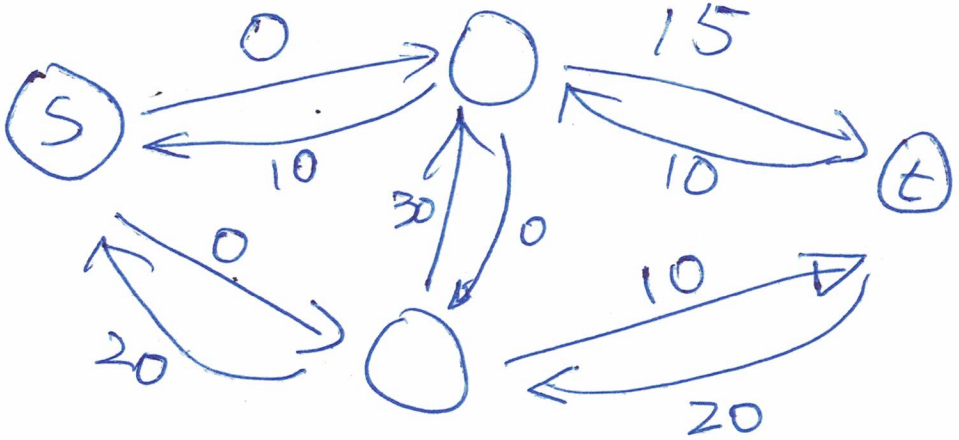
Residual graph:



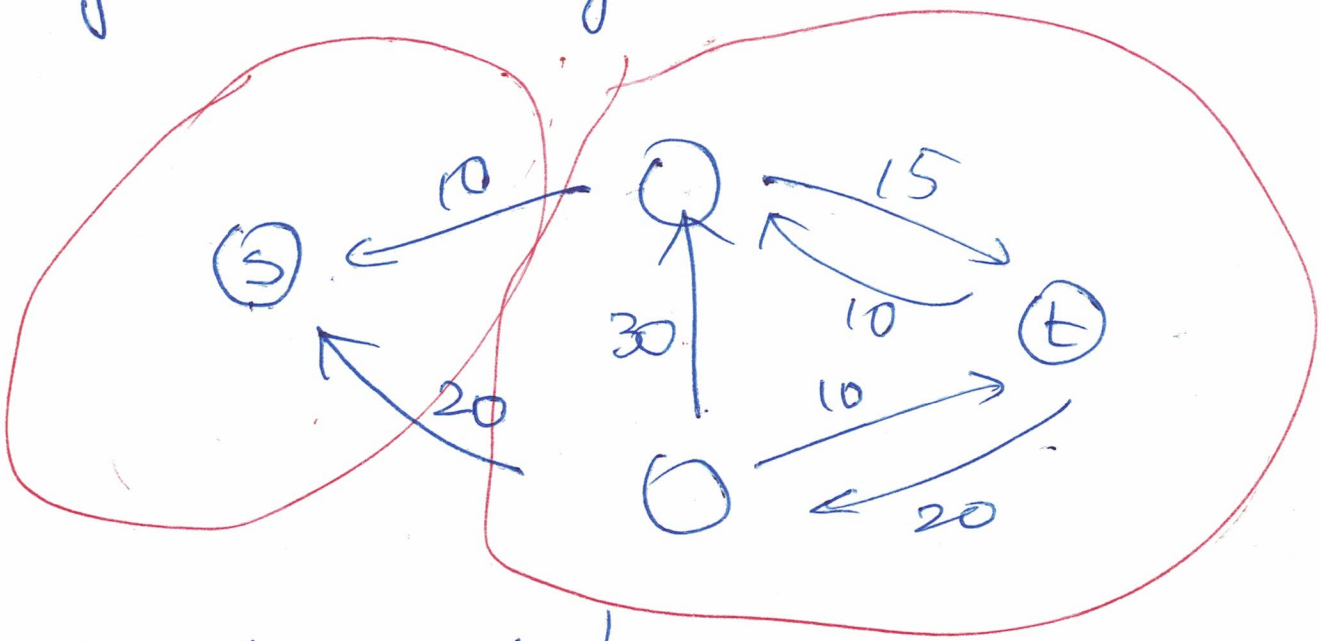
Exaple:



⇒ residual graph



drop all 0-edges.

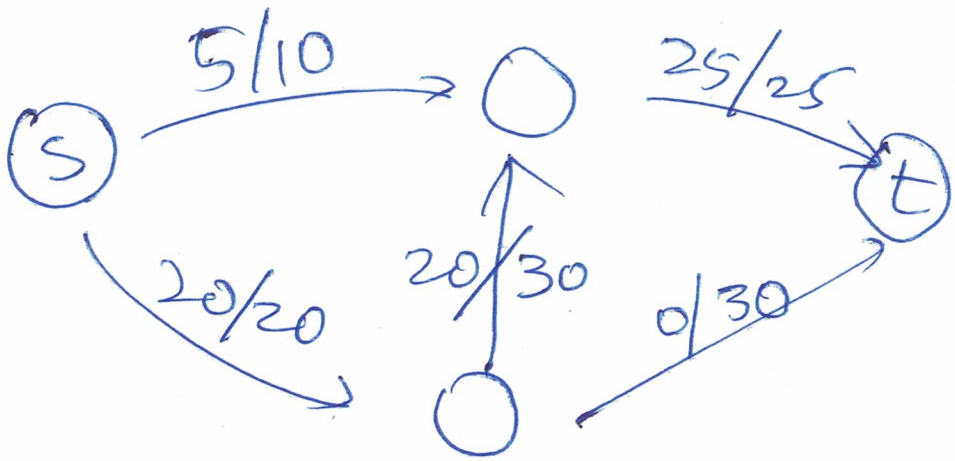


"s" got stuck!

$A = \{s\}$

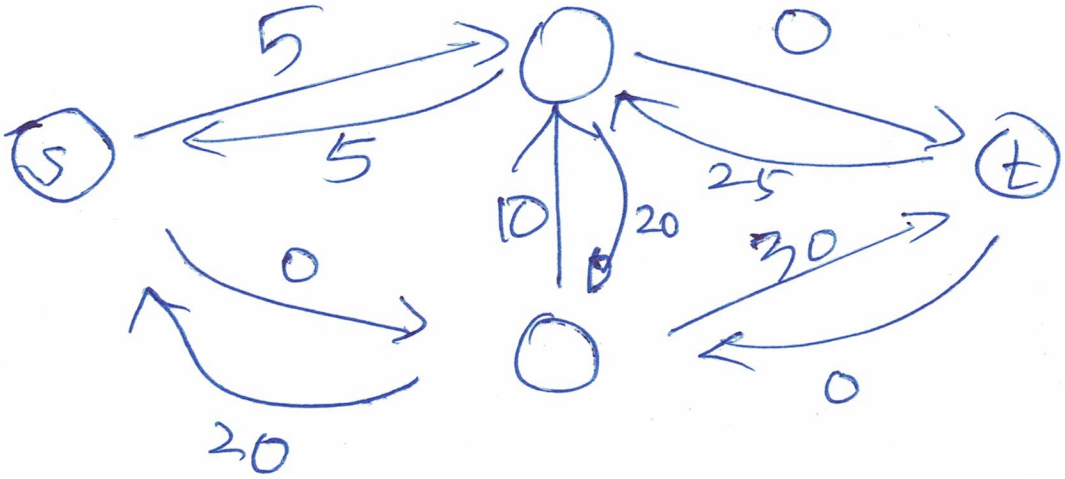
$B =$ the rest three nodes.

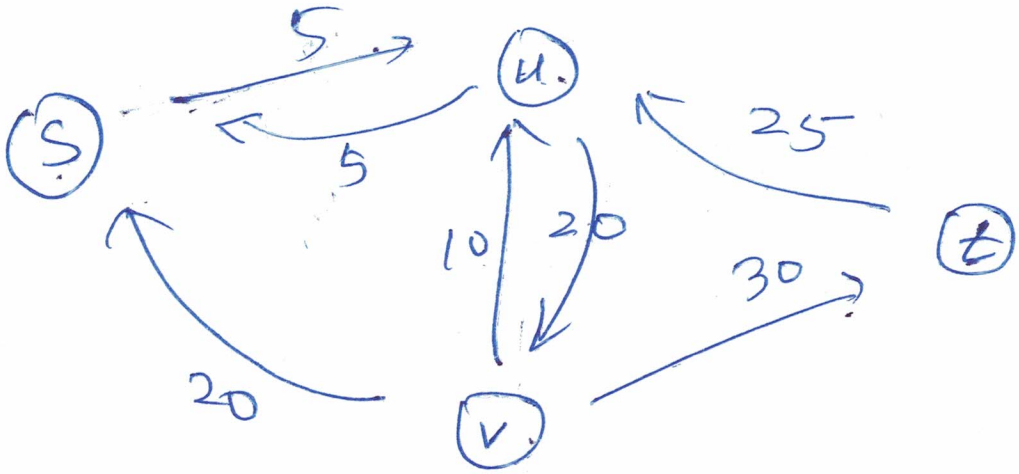
more exple:



is local max.

Residual graph:

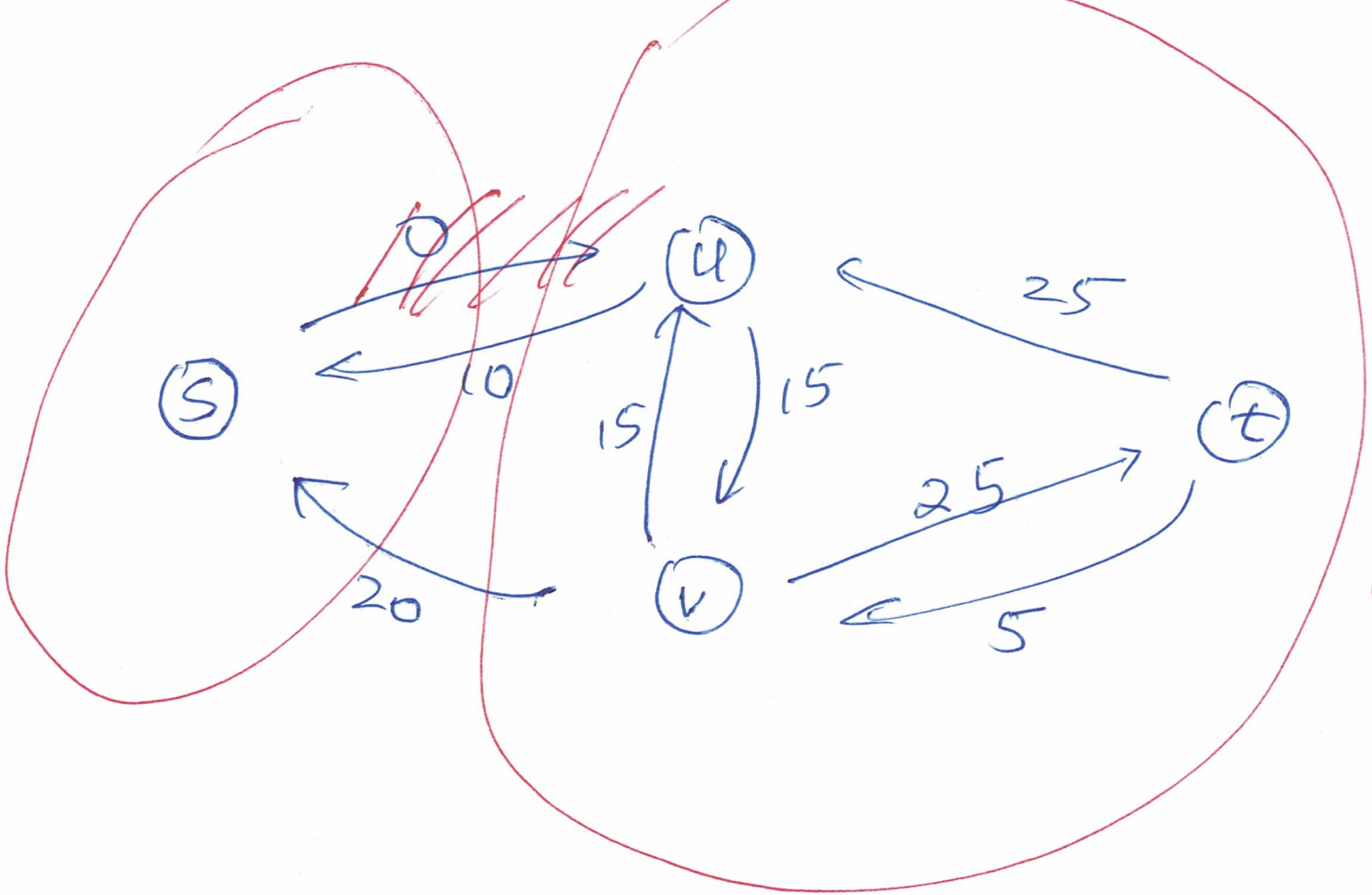




is not gobal max. why?

$$s \xrightarrow{5} u \xrightarrow{20} v \xrightarrow{30} t$$

where we can still put 5 additional flow on the path!



is
global max!

On residual graph,

an augmenting path α is a simple path from s to t such that each edge on the path has positive residual capa.

(Recall: $s \xrightarrow{5} u \xrightarrow{20} v \xrightarrow{30} t$)

FF Alg:

Given G and C, ← capacities on edges.

let $f(e) = 0$, $e \in E$

Repeat

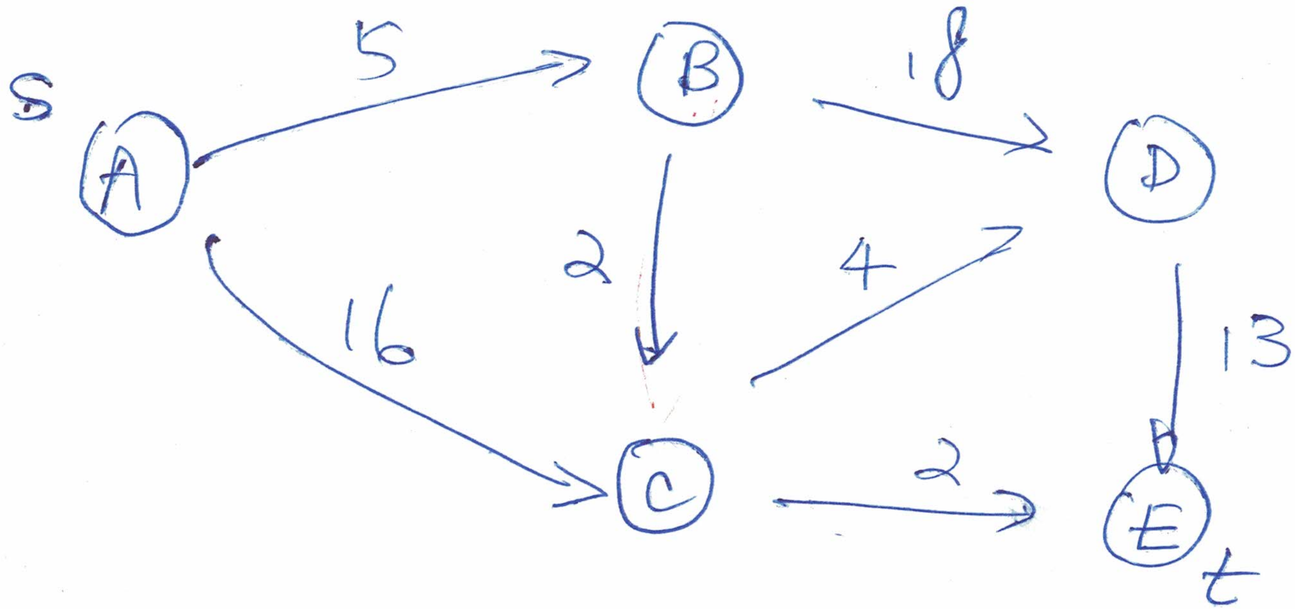
update residual graph ← called G_f under current flow f;

Find an augmenting path in G_f

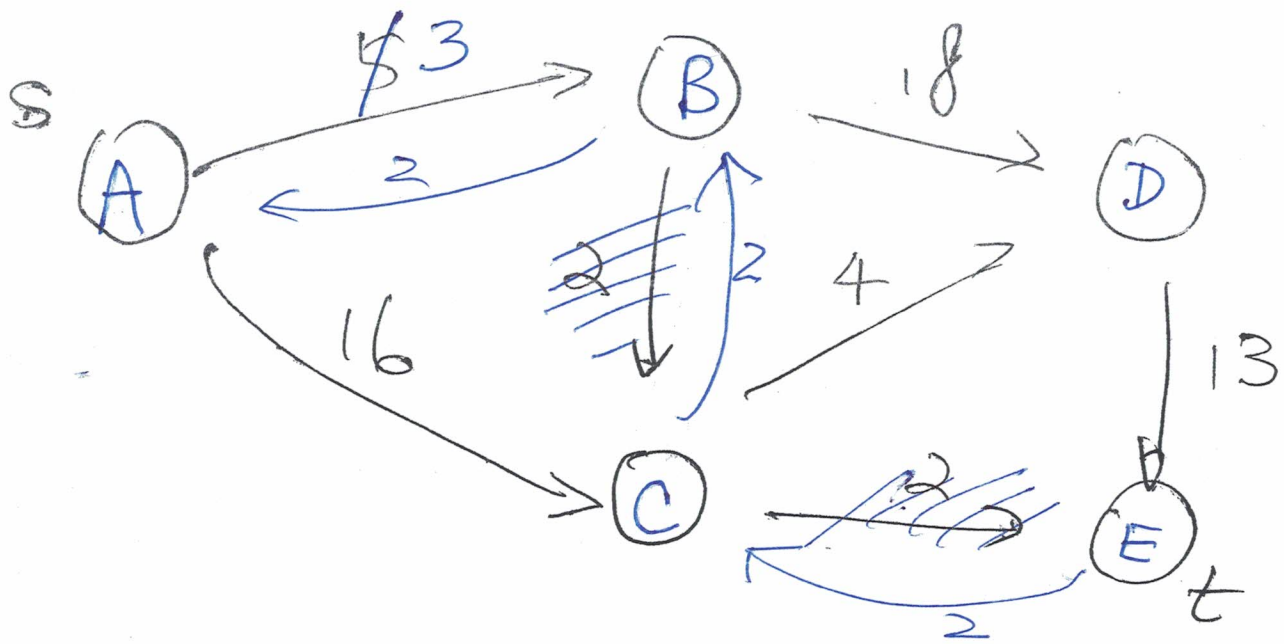
Adjust the flow f (using the path);

Until no augmenting path can be found.

Good Exaple.

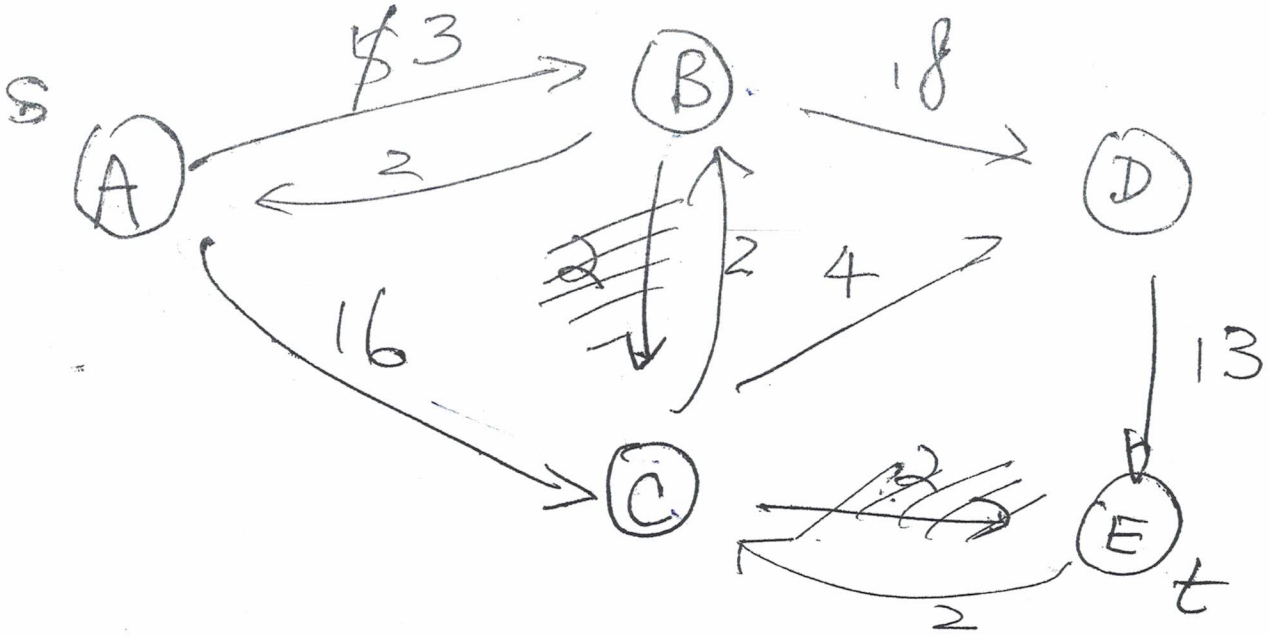


Good Example.



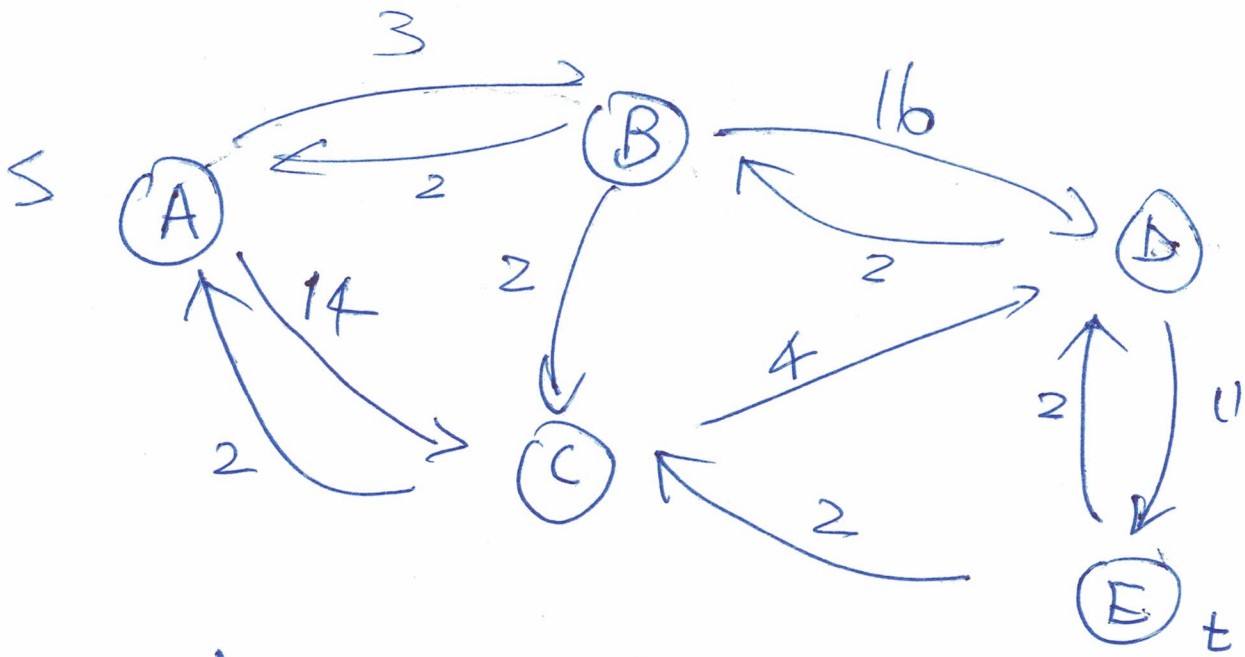
augmenting path: A B C E

Good Example.



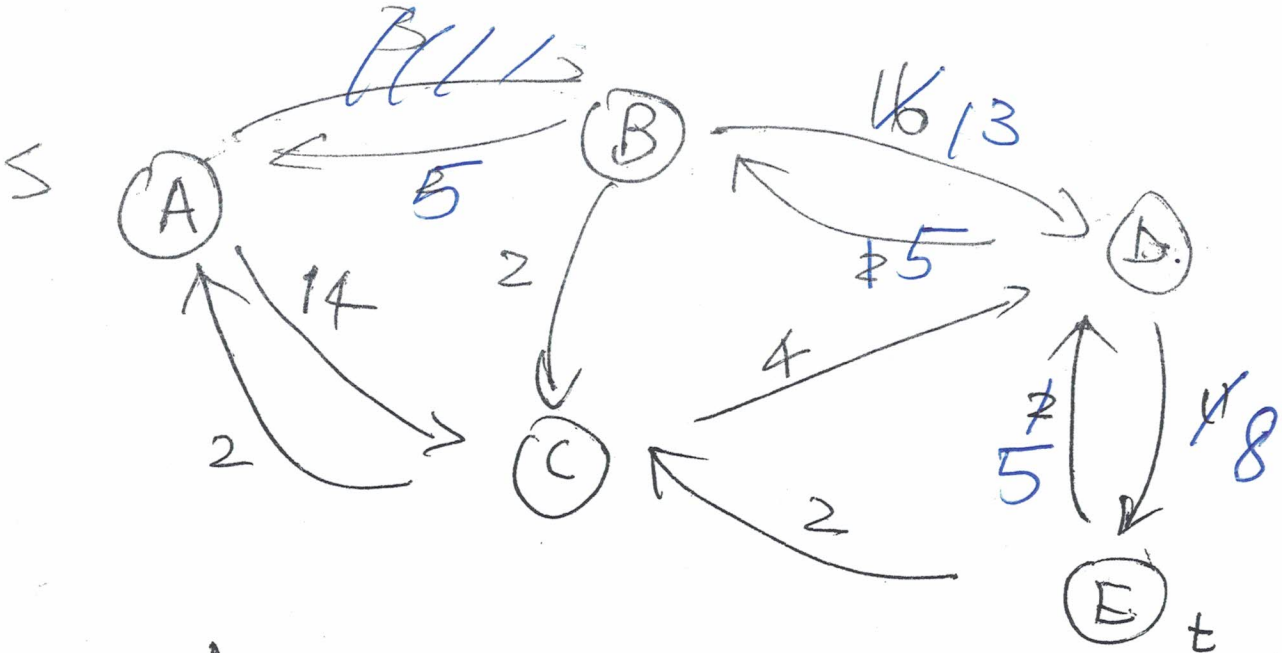
~~augmenting path: A B C E~~

augmenting path: A C B D E



augmenting path ?

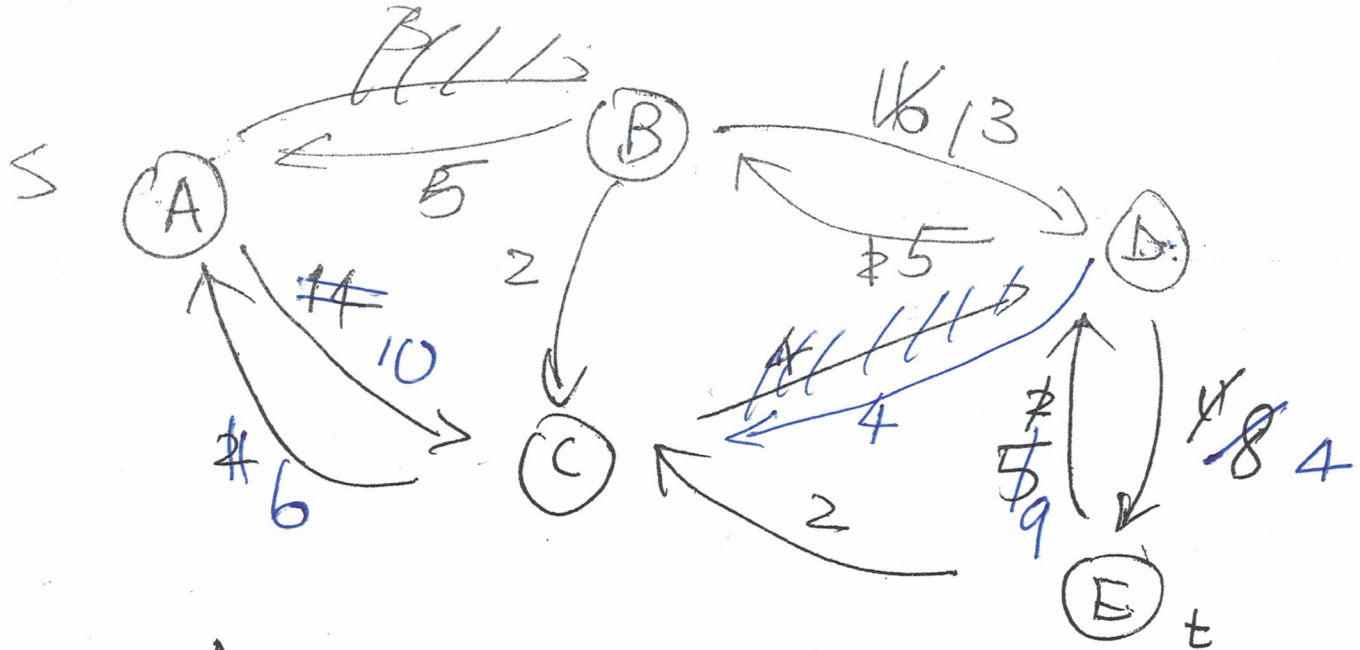
A B D E



~~augmenting path?~~

~~A B D E~~

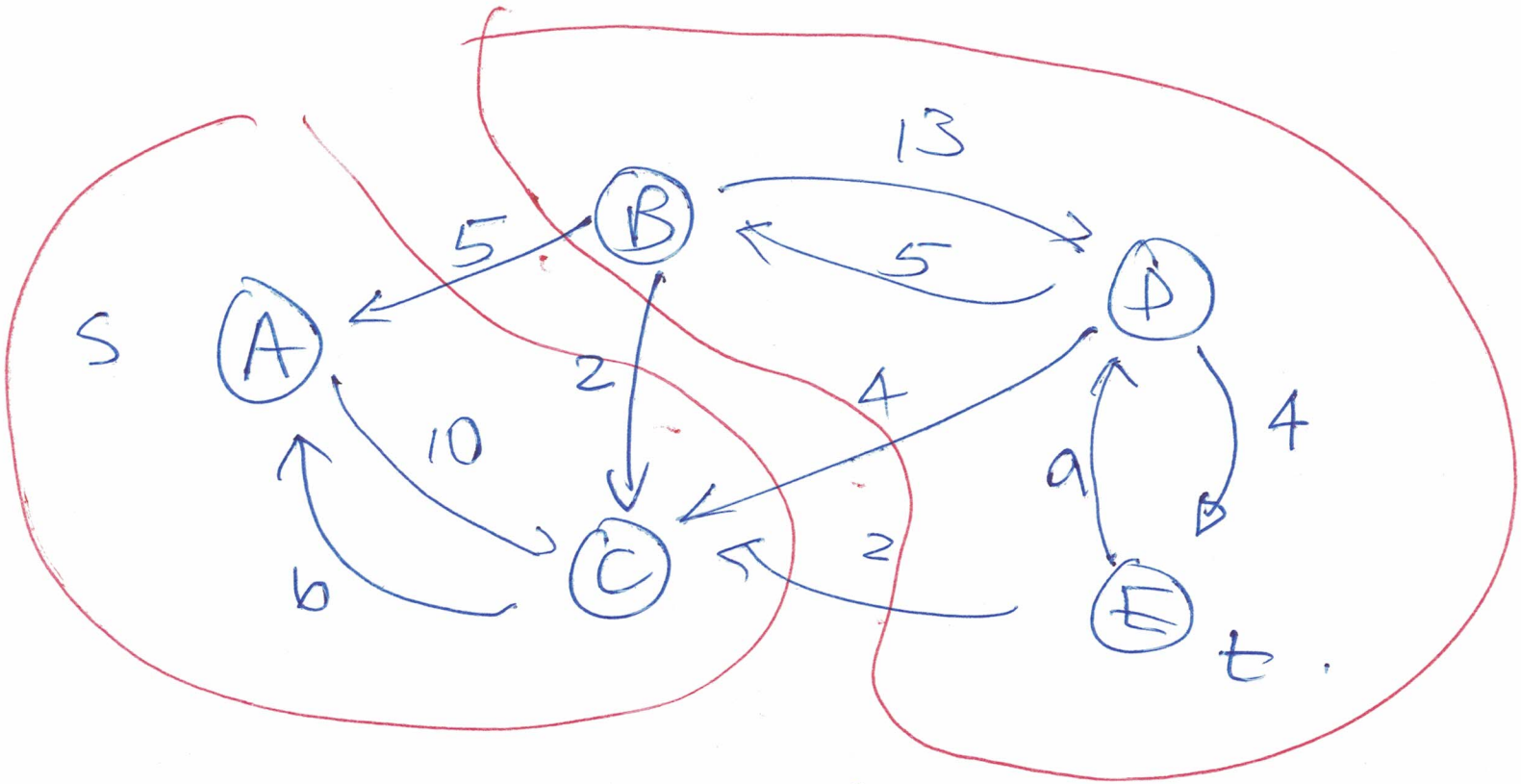
augmenting path? A →(14) C →(4) D →(8) E



~~augmenting path?~~

~~A B D E~~

~~augmenting path?~~ A C D E (14, 4, 8)

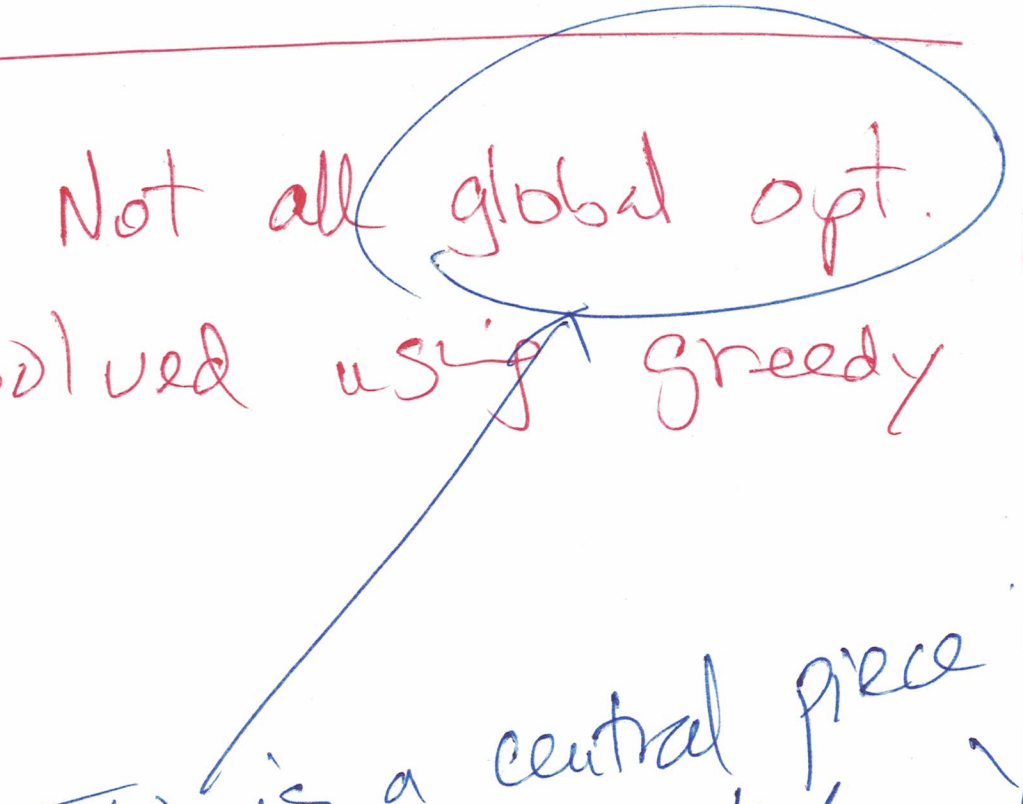


augmenting path (from A to E)?

NO.

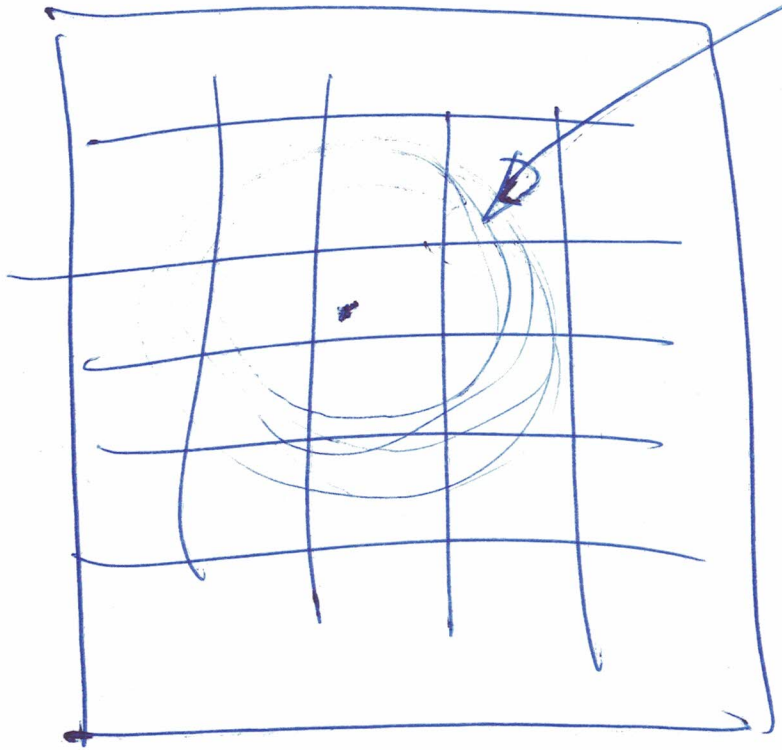
Complexity and implementation details: Read textbook & Online resources.

Remark: Not all global opt. problems can solved using greedy !



HW:

3. HWA.



4. For two programs

P_1, P_2.

Can you define a number to measure the distance (or similarity) between the "meaning" of the programs?

↑ semantics!

Don't work on syntactic similarity.

↑ (source code level).

Two cases:

① pogrums are whiteboxes;

② pogrms are blackboxes.