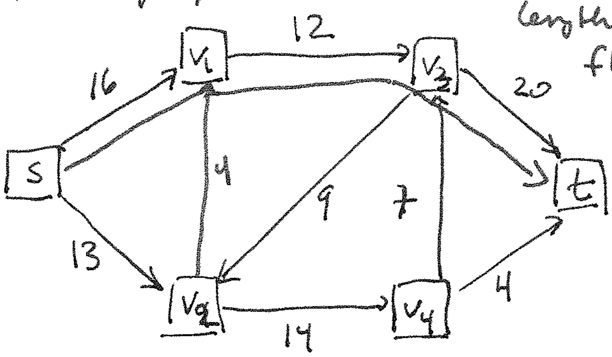
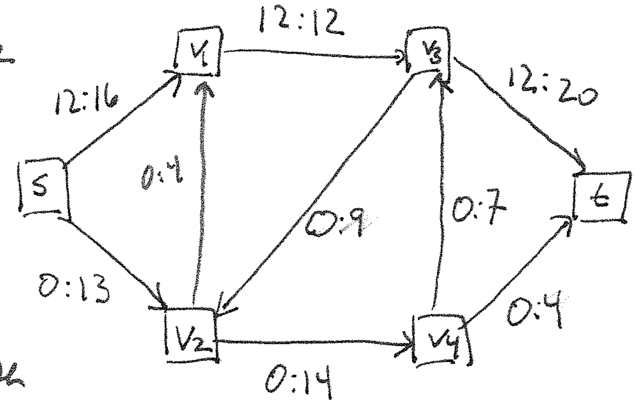
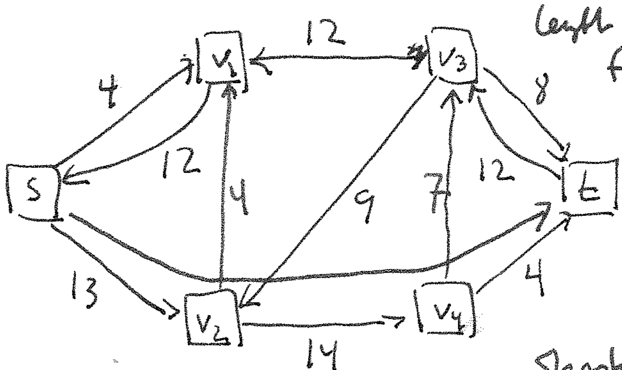
4A Residual graphs are on the left, with augmenting flow marked. Note that the initial residual graph is the original input graph

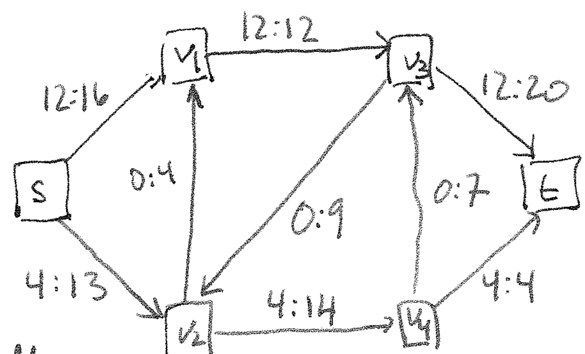
Shortest path length = 3 flow = 12



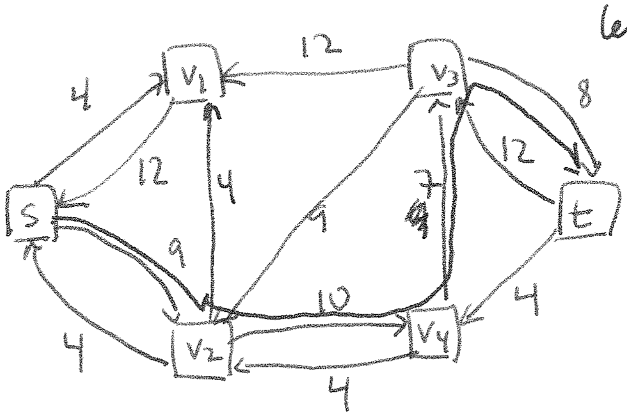


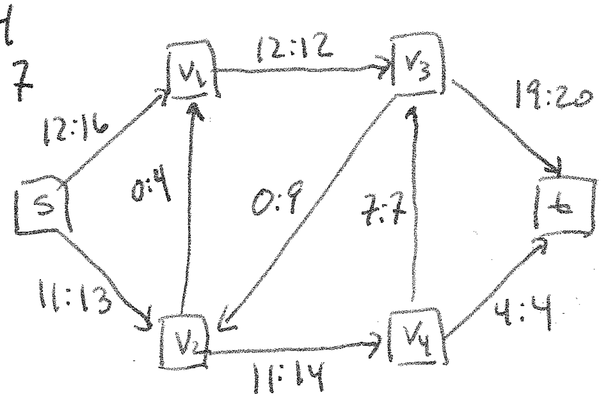
Shortest path length = 3 flow = 4

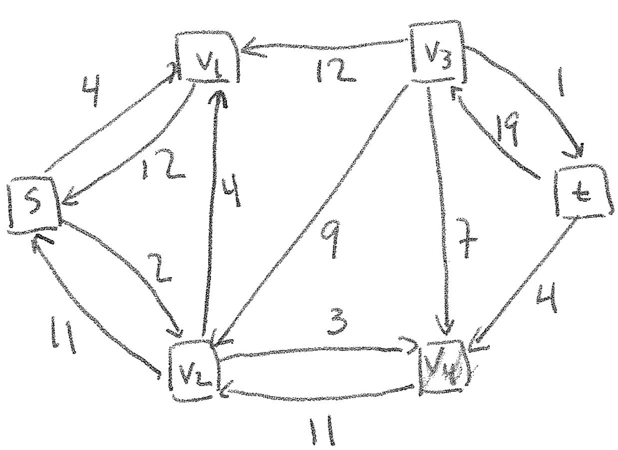




Shortest path length = 4 flow = 7







No augmenting path

Final flow value = 23

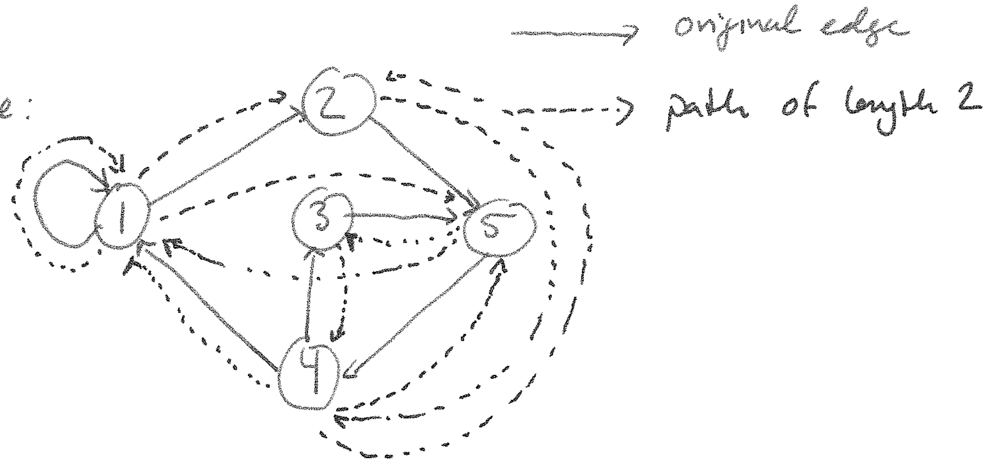
Cut with capacity 23

$S = \{s, v_1, v_2, v_4\}$

$T = \{v_3, t\}$

4B A^2 Gives all paths of length 2 in the graph.

Example:



Adjacency Matrix — times itself — Paths of length 2

$$
\begin{array}{c|ccccc}
 & 1 & 2 & 3 & 4 & 5 \\
\hline
1 & 1 & 1 & 0 & 0 & 0 \\
2 & 0 & 0 & 0 & 0 & 1 \\
3 & 0 & 0 & 0 & 0 & 1 \\
4 & 1 & 0 & 1 & 0 & 0 \\
5 & 0 & 0 & 0 & 1 & 0
\end{array}
\cdot
\begin{pmatrix}
1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 \\
1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0
\end{pmatrix}
=
\begin{pmatrix}
1 & 1 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 & 1 \\
1 & 0 & 1 & 0 & 0
\end{pmatrix}
$$

4C Possible to evaluate in 9 multiplications.

Given x, compute

$x^2 = x \cdot x$	$x^{36} = x^{24} \cdot x^{12}$
$x^4 = x^2 \cdot x^2$	$* \; x^{37} = x^{36} \cdot x$
$* \; x^6 = x^4 \cdot x^2$	$x^{49} = x^{37} \cdot x^{12}$
$x^{12} = x^6 \cdot x^6$	$* \; x^{55} = x^{49} \cdot x^6$
$x^{24} = x^{12} \cdot x^{12}$	

The starred (*) products can be added together to get the value of the polynomial.

4D(a) This direction combines the ideas of 4B and 4C. What we want to compute, given adjacency matrix A for graph G, is

$$A^0 + A^1 + A^2 + \cdots + A^{n-1}$$

(A^0 is the identity matrix)

A^0: paths of length 0; A^1: paths of length 1; A^2: paths of length 2; A^{n-1}: paths of length n-1

Note that $(A^0 + A^1 + \cdots + A^k) \cdot (A^0 + A^1 + \cdots + A^k) =$

$$(A^0 + A^1 + \cdots + A^{2k})$$

So we can start with

$$P^1 = (A^0 + A^1)$$

and do repeated multiplication to get

$$P^2 = P^1 \cdot P^1$$
$$P^4 = P^2 \cdot P^2$$
$$P^8 = P^4 \cdot P^4$$
$$\vdots$$
$$P^{2^k} = P^{2^{k-1}} \cdot P^{2^{k-1}}$$

P^{2^k} has all paths of length $\leq 2^k$. So we need to keep multiplying until $2^k \geq n-1$. For this, $k = \lg n$ will work. So we have $\lg n$ Boolean matrix multiplies, each which takes time $B(n)$, for $O(B(n) \cdot \lg n)$

4 D (b) We can turn Boolean matrix multiply into finding paths in a graph. Given Boolean matrices C_1 and C_2, we will turn them into a graph G where there is a path of length 2 in the graph for every position where $C_1 \cdot C_2$ has a 1.

Assume C_1 and C_2 are $n \times n$.

Graph G has 3 sets of nodes making up N:

$N_1 = v_1, \ldots, v_n$
$N_2 = w_1, \ldots, w_n$ — edges between these 2 represent 1's in C_1

~~$N_2 = x_1, \ldots, x_n$~~
$N_3 = x_1, \ldots, x_n$ — edges between these 2 represent 1's in C_2

So $N = N_1 \cup N_2 \cup N_3$

Let $E_1 = \{(v_i, w_j) \mid C_1[i,j] = 1\}$

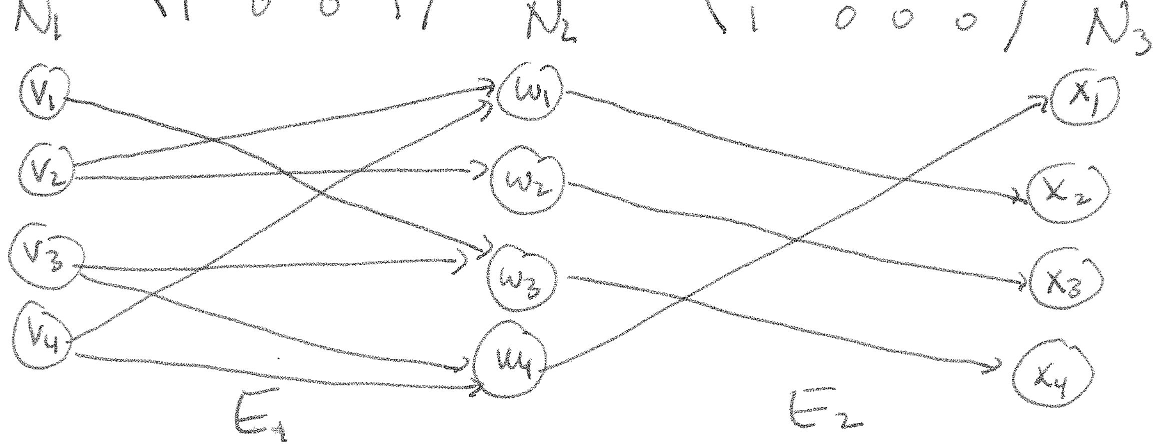
$E_2 = \{(w_j, x_k) \mid C_2[j,k] = 1\}$

And $E = E_1 \cup E_2$

Example:

$$C_1 \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix} \qquad C_2 \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

$G =$



4D (b) Continued

In the transitive closure of G, there will be an edge (v_i, x_k) exactly when $(C_1 \cdot C_2)[i,k] = 1$.

For example, there is (v_1, x_4) in the transitive closure (by $\cancel{x_1} \to w_3 \to x_4$), and $(C_1 \cdot C_2)[1,4] = 1$.

Another way to look at what we are doing is to ~~let A_1 and A_2 be adjacen~~ look at the adjacency matrix of G, which looks like

$$\begin{array}{cc} & 3n \\ 3n & \begin{pmatrix} 0 & C_1 & 0 \\ 0 & 0 & C_2 \\ 0 & 0 & 0 \end{pmatrix} \end{array} \quad \text{whose transitive closure is} \quad \begin{pmatrix} I & C_1 & C_1 \cdot C_2 \\ 0 & I & C_2 \\ 0 & 0 & I \end{pmatrix}$$

Complexity:

- Constructing G from C_1 and C_2 takes $O(3n \cdot 3n) = O(n^2)$ time
- The transitive closure of G takes $T(3n)$ time, which under the reasonable assumption that $T(3n) \le k \cdot T(n)$ for some k, is $O(T(n))$
- Extracting ~~to~~ $C_1 \cdot C_2$ from the transitive closure of G is $O(n^2)$

Assuming $T(n)$ is $\Omega(n^2)$, since we have to read the input, the whole process is $O(n^2) + O(T(n)) + O(n^2)$
$= O(T(n))$.

4 E.

Given $n \times n$ matrices A and B, construct a $2n \times 2n$ matrix $C = \begin{pmatrix} 0 & A \\ B & 0 \end{pmatrix}$

then $C^2 = \begin{pmatrix} 0 & 0 \\ 0 & AB \end{pmatrix}$

If we assume $S(2n) \leq k\, S(n)$ for some k, then $n \times n$ matrix multiply can be done in

$$O(S(2n)) = O(k\,S(n)) = O(S(n))$$ time.