Thomas E. Armstrong

Sandipan Dey

1. **(Math 685, 15 pts)** Explain how to vary both Prim's Algorithm and Kruskal's algorithm so as to give algorithms for finding maximum weight spanning trees. Be sure to comment upon the importance of stopping when faced with creating a circuit. It maybe helpful to consider replacing weights $w(e)$ of edges e with costs $c(e)$ of the form $M - w(e)$

$\frac{6}{15}$

$\frac{93}{115}$

why is maximum weight tree guaranteed ?

1. ~~Ka~~ Both prim's and kruskal's algorithm find the minimum ~~w~~ weight spaning trees. Prim grows the tree everytime choosing a minimum cost edge that has one endpoint in the tree, other vertex being outside (it respects the 'cut' while building the tree) when kruskal grows the tree ~~return~~ by simply choosing a minimum cost edge that does not have both endpoints in ~~a~~ the same set in the forest and grows the forest (starting with each vertex as ~~an a for~~ a tree initially in the forest)

~~$w(e) = \max_{e \in E[G]} e - w(e)$~~

$M = \max_{e \in E[G]} w(e)$

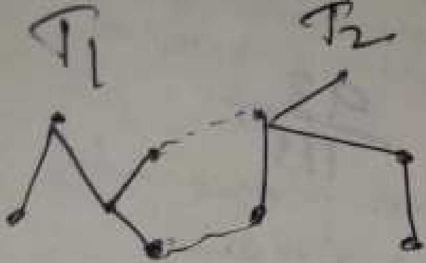
Now, if $\forall e \in E[G]$ if we first change ~~cost~~ $w(e) = M - w(e)$ (simply) negate the weights of the edges both the algorithms will choose the min^m edge (that's max^m negated edge) each time and ultimately build a max^m spaning tree. ~~(since min ...)~~ (since $\min_i(x_i) = -\max_i(x_i)$)

1. $M \leftarrow \max_{e \in E[G]} w(e)$
2. $\forall e \in E[G]$, $w(e) \leftarrow M - w(e)$.
3. Run prim/kruskal to find MST

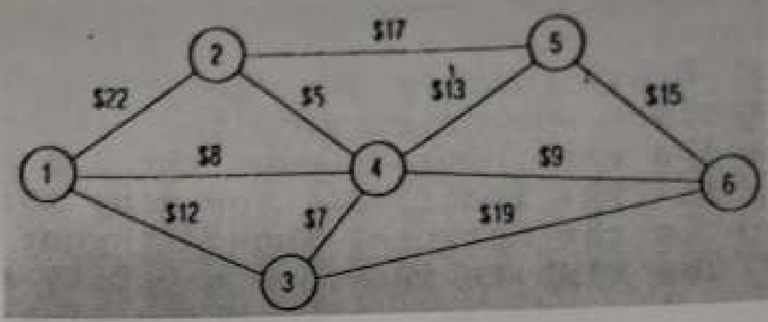
cut for Prim's algo

~~When~~ Prim's algorithm always chooses an edge respecting the cut T and $V[G] - T$ where T is the tree built so far. Hence any ~~negative~~ circuit consiting of negative weight edges can't violate this CUT invariant, i.e., not all edges of the circuit can be chosen for the tree since the last edge will violate the 'CUT' condition. ~~S~~ Hence even if the negative circuits can ~~cost~~ reduce the cost of MST infinitely, they will NOT be chosen

This same is true for kruskal as well, which grows forest by ruling out inclusion of any edge that has both enpoints in the same tree of the forest. So in this case also, even if ~~the~~ a negative circuit can reduce cost of the tree chosen infinitely, it will not be chosen, since the last edge of the circuit will not be added to MST, because if ~~it~~ has both enpoints in the same set.



~~forest~~. forest by kruskal.



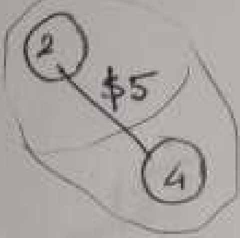
Each node represents the location of a computer terminal, while each branch identifies a potential hookup line between the terminals. Branch values give the costs (in thousands of dollars) for connecting each pair of terminals. The main computer with every other terminal in the network (so there is a path from any node to any other node).

15/20

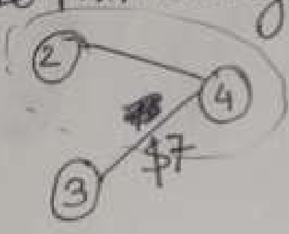
(a) How much will this design cost? Indicate design.

(b) If however a design is an arbitrarily chosen tree, what is the worst cost overrun?

We use Prim's algorithm to find the min^{m} design cost

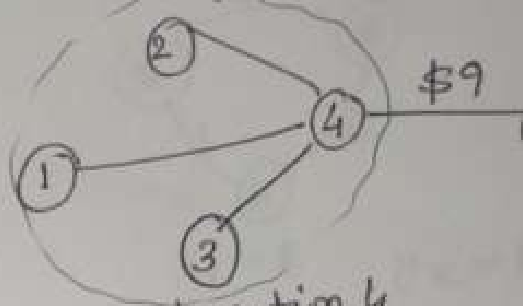


iteration 1

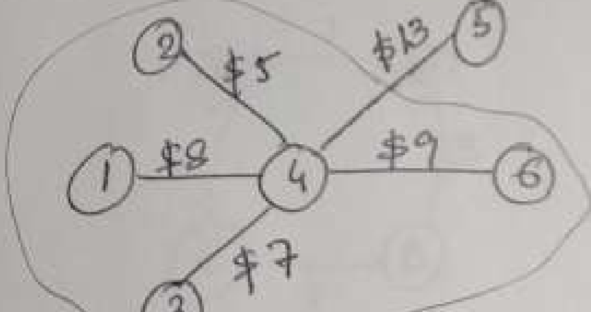


iteration 2

iteration 3



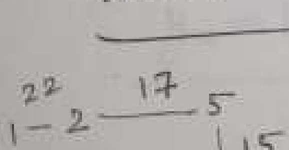
iteration 4



iteration 5

Hence the total cost of the MST is = 5+7+8+9+13
= $42K ✓

Max-cost of a tree ~~...~~ arbitrarily chosen will be = (22) + (19) + (17) + 15 = $75K.

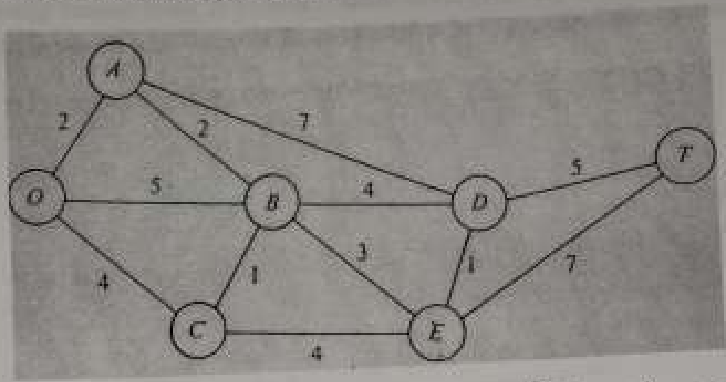


Max cost tree (worst case)

22
19
17
15
73

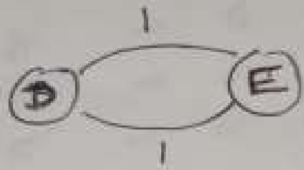
Hence the cost overrun
= 75-42
= $33K

3. **(15 points)** Use the approximate tour construction algorithm for the Traveling Salesman Problem in Seervada Park; choose a starting point judiciously.

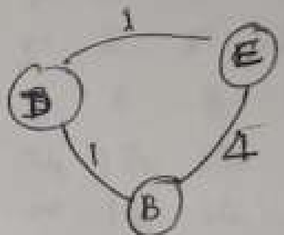


We choose the starting point as D since it has ~~max degree and max number of~~ vertices adjacent with minimal costs.

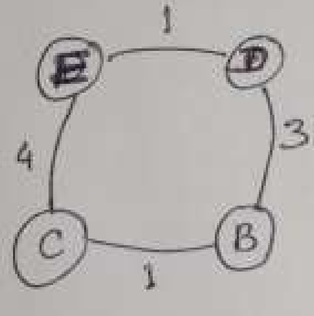
$\frac{15}{15}$



$z_K = E$ is the nearest point from $y_K = D$ on circuit

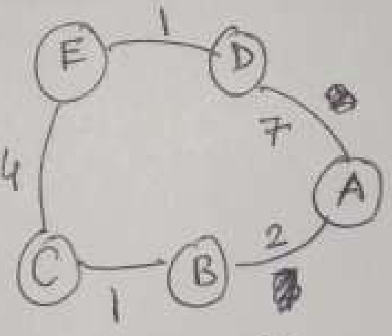


$z_K = B$, $y_K = E$



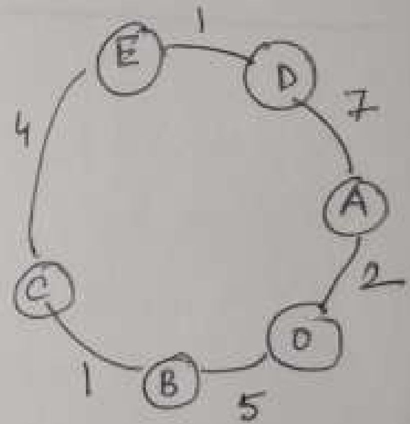
$z_K = C$

$y_K = B$



$z_K = A$

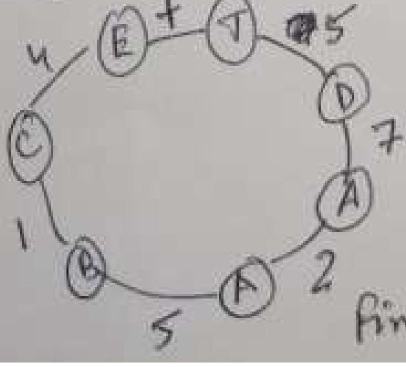
$y_K = B$



$z_K = O$

$y_K = A$.

$OA + OB = 2+5 = 7$



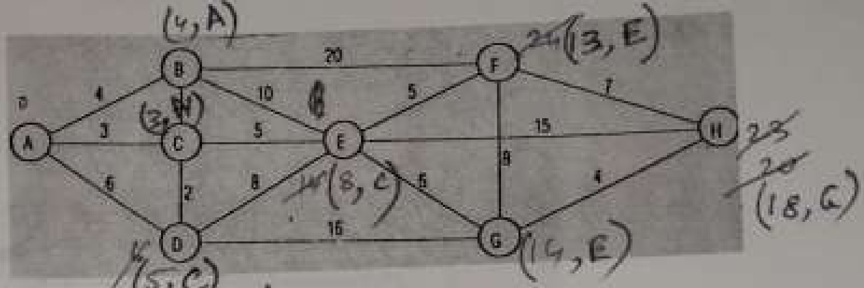
Final Hamilton circuit

4. **(20 points)** The following network shows the costs (in thousands of dollars) of distributing electricity from the production facility A through various transmission points to a large industrial customer H. Find the least costly routes from A to every other node ($A$to B, A to C, etc.). Indicate solution by labeling nodes (c, p) where c is the cost from A and p is the immediate predecessor

~~10~~ 20/20

Run Dijkstra

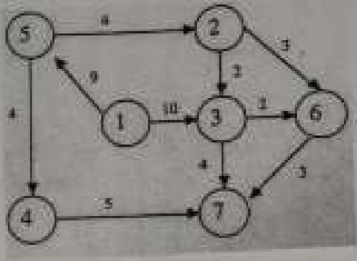
d(A) ← 0
d(B) ← 4
d(C) ←

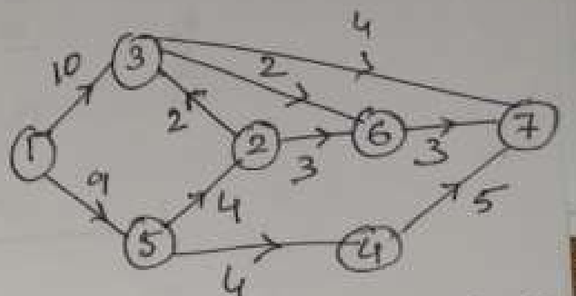


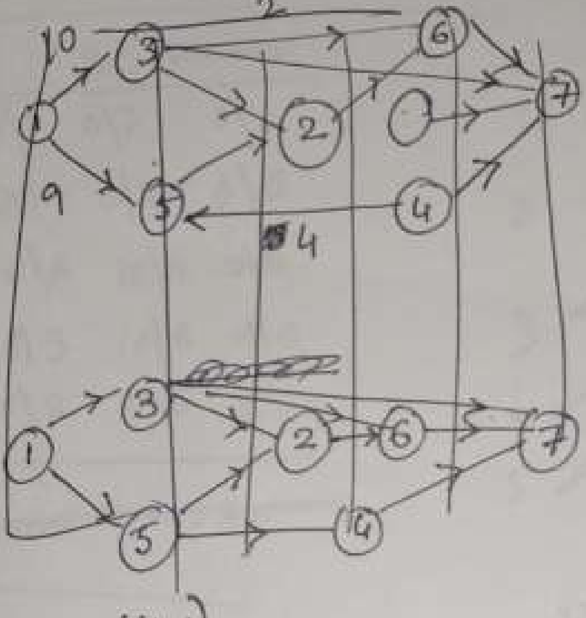
Shortest Distances from A	Iterations 1	2	3	4	5	6	7	8
~~A~~	0	0	0	0	0	0	0	0
~~B~~	∞	4	4	4	4	4	4	4
~~C~~	∞	3	3	3	3	3	3	3
~~D~~	∞	6	∞	5	5	5	5	5
~~E~~	∞	∞	14	8	8	8	8	8
~~F~~	∞	∞	24	24	24	13	13	13
~~G~~	∞	∞	∞	∞	∞	14	14	14
H	∞	∞	∞	∞	∞	23	20	18

5. **(20 points)** The diagram below represents the lubrication system of a machine; the lubricant flows from a source area at node 1, through components 2-6, which require lubrication, and collects at node 7. Edge capacities are maximum allowable flow rates from one position to another. Find the feasible flow that maximizes the total flow of lubricant through the machine. Determine the minimum cut.

20/20







Now let's run the algorithm to find max flow and min-a-z cut.

We start with flows

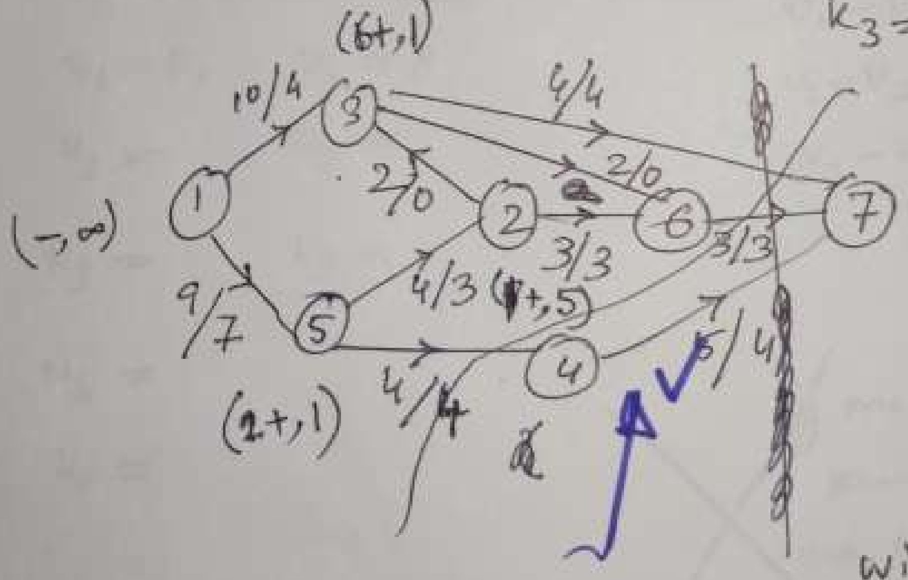
$K_1 = 1-3-7$, $f_{K_1} = 4$.

$K_2 = 1-5-4-7$, $f_{K_2} = 4$.

$K_3 = 1-5-2-6-7$, $f_{K_3} = 3$

by observation.

Now apply algorithm



But we can see that ~~if we~~ we already have ~~cut~~ a saturated cut with $\bar{P} = \{7, 5\}$ 7.

and $K(P, \bar{P}) = 4+3+4$
$= 11$
$= f_{max} =$ maximum flow

6. **(25 points)** For this transportation problem with unbalanced supply and demand

(a) Set up an appropriate transportation simplex tableau.

(b) Determine the initial basic feasible solution using a minimum cost heuristic.

(c) If necessary perform one iteration after testing for optimality before and after.

$\frac{10}{25}$

Stores

		1	2	3	
	1	8	5	7	25
	2	5	∞	3	25
Warehouses	3	3	8	9	20
	4	7	3	6	10
	5	4	8	6	20
		25	30	30	

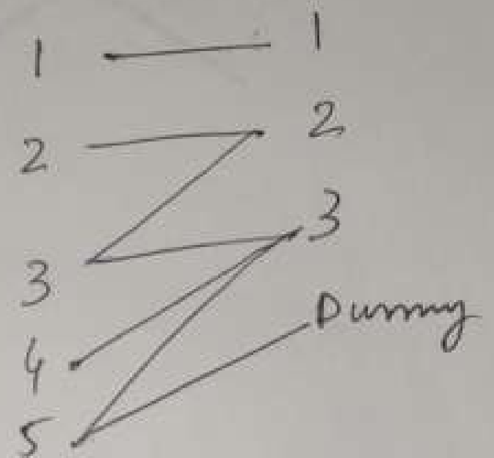
85

50
20
30
100

not a minimum cost Heuristic

(a) By North west corner method.

	1	2	3	Dummy
1	25\8	0\5	0\7	0\0
2	0\5	25\∞	0\3	0\0
3	0\3	5\8	15\9	0\0
4	0\7	0\3	10\6	0\0
5	0\4	0\8	5\6	15\0



(b) ~~By min. cost heuristic.~~

$u_1 = 0, \quad v_1 = 8$

$u_2 = \quad v_2 =$

$u_3 = \quad v_3 =$

$u_4 =$

$u_5 =$

$v_1 - u_1 = 8.$

$v_2 - u_2 =$

$v_3 - u_3 =$

~~if me is +ve decreasing than corresponding cost we can ~~send additional~~ add that edge, find a circuit~~

7. **(BONUS, 20pts)** A cut set in an undirected graph G is a set C of edges whose removal disconnec G but the removal of any proper subset leaves G connected.

3/20

(a) Show that in a flow network, a cut set that separates a from z is an $a-z$ cut.

(b) Can $G \backslash C$ have more than two connectivity components?

(a) cut set $(P, \bar{P})$ where $a \in P$ and $z \in \bar{P}$ separates a from z.

and its a min ~~cut~~ ?? by max flow min-cut theore

(b) $G \backslash C$ can't have more than one connectivity component.

why?.