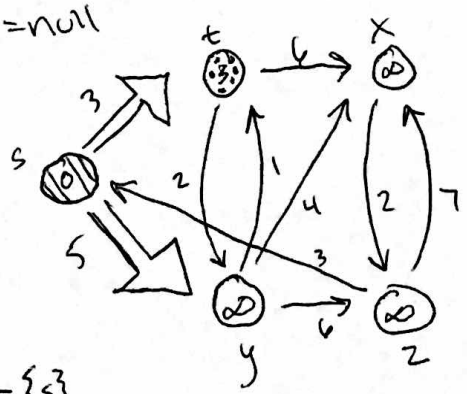
George Katsigiannis
Assignment #4

1) initially $S = \{\}$, In order to comply with Method of figure 24.6, striped circle represents π and dotted circle represents d and Bolded path represents iterations direction.

Start of first $d=0$, $\pi = null$

End of first iteration:

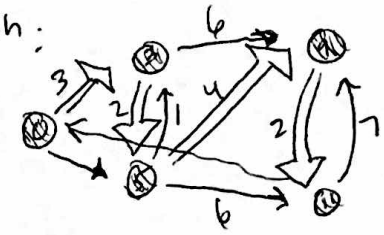


$S = \{s\}$

$\pi[t] = s$

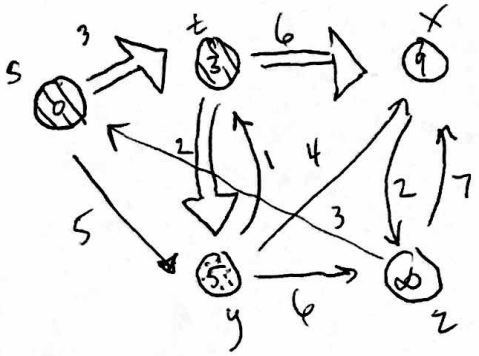
$d = 3$

5th:



$S = \{s,t,x,y,z\}$

2nd:

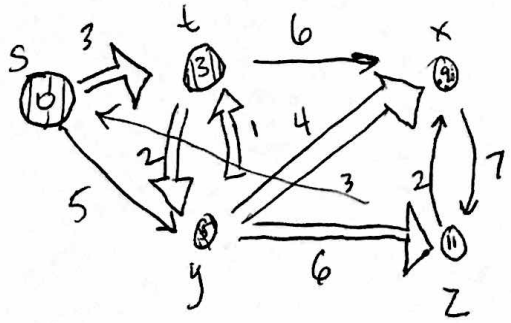


$S = \{s,t\}$

$\pi[y] = t$

$d = 5$

3rd:

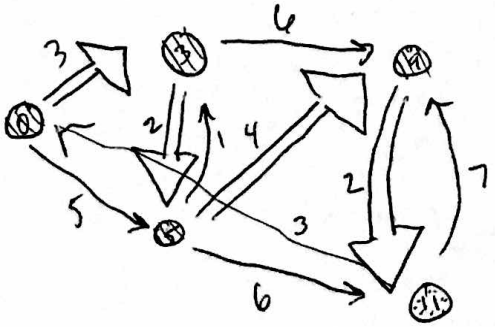


$S = \{s,t,y\}$

$d = 9$

$\pi[x] = y$

4th:

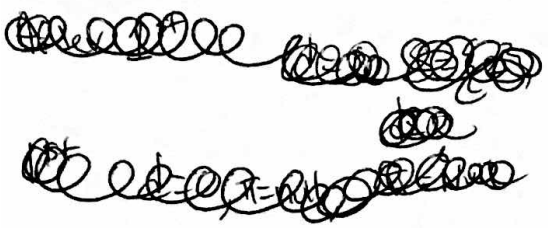


$S = \{s,t,y,x\}$

$d = 11$

$\pi[z] = x$

Now let initial vertex be z



1st: $d=0$, $\pi=$ null, $S=\{z\}$

2: $d=3$, $\pi=z$ $S=\{z,s\}$

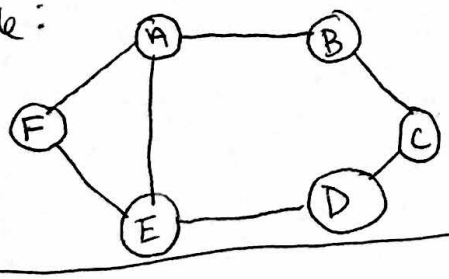
3: $d=6$, $\pi=s$ $S=\{z,s,t\}$

4: $d=7$, $\pi=t$ $S=\{z,s,t,x\}$

5 $d=8$, $\pi=t$ $S=\{z,s,t,x,y\}$

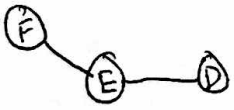
3. No, Pinocchios algorithm does not always produce the maximum set

Counter example:



Scenario 1:

Take away B



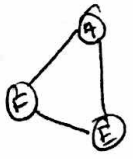
Take away D (F)

Take away F

Set = {B, D, F}

Scenario 2:

Take away C:



Take away F: Empty Graph

$S=\{C,F\}$

Clearly from scenario 1 we see the max set is 3 and not 2, but if you started with C you would not get the max thus proving the greedy algorithm wrong.

4) Let a_i = activity
s_i = start time
f_i = finish time

Least duration from among those compatible with previously selected activities

a_i	1	2	3
s_i	0	3	4
f_i	4	5	7

→ By selecting the least duration the solution you get is a_2, but the optimal solution here is a_1 and a_3.

Overlap fewest remaining

a_i	1	2	3	4	5	6	7	8	9	10	11
s_i	0	1	1	1	2	3	4	5	5	5	6
f_i	2	3	3	3	4	5	6	7	7	7	8
overlaps	3	4	4	4	4	2	4	4	4	4	3

→ This approach yields a_6 and one of a_1 a_2 a_3 a_4 and one of a_8 a_9 a_{10} a_{11} for a total of 3 activities when the optimal solution is a_1 a_5 a_7 a_{11}

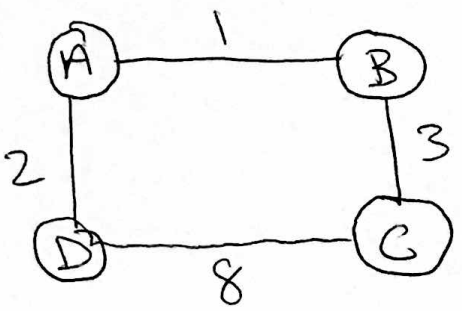
Earliest start time

a_i	1	2	3	4	5	6	7	8	9	10	11
s_i	1	3	0	5	3	5	6	8	8	2	12
f_i	4	5	6	7	8	9	10	11	12	13	14

→ a_1 will be selected and no other activities when is not optimal

5) This will not work.

Counter example:



→ Divide graph into G_1, G_2

$G_1 = \{A, B\}$ $E_1 = \{(A,B)\}$

$G_2 = \{D, C\}$ $E_2 = \{(D,C)\}$

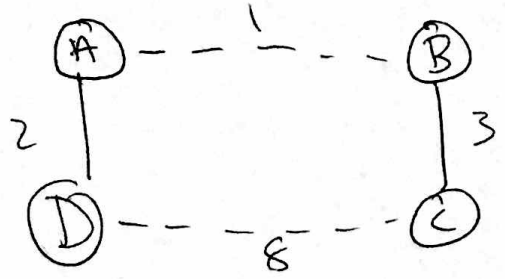
Edges that connect $G_1, G_2 = \{(A,D),(B,C)\}$

Calculate minimum spanning tree of G_1, G_2

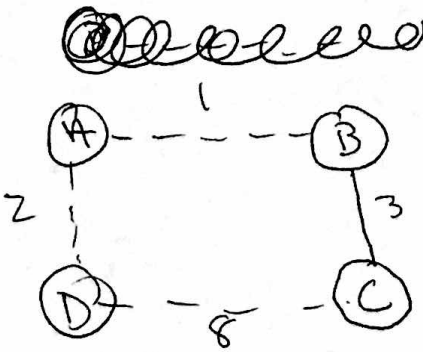
$MST(G_1) = G_1$

$MST(G_2) = G_2$

So



Edge(A,D) is minimum so the MST is



which is not the minimum spanning tree of G_1.