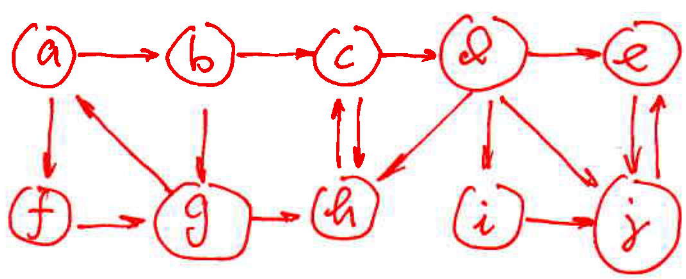
CS5413 - Data Structures and Algorithm Analysis III - Exam #3
(Total 100 points)
(In class, Closed book/notes)
(Justify every step of your answers)

Name (Last name, First name): Key

1. (30 points) Given the following algorithms and graph as described in the adjacency matrix, answer the following questions.

	a	b	c	d	e	f	g	h	i	j
a	0	1	0	0	0	1	0	0	0	0
b	0	0	1	0	0	0	1	0	0	0
c	0	0	0	1	0	0	0	1	0	0
d	0	0	0	0	1	0	0	1	1	1
e	0	0	0	0	0	0	0	0	0	1
f	0	0	0	0	0	0	1	0	0	0
g	1	0	0	0	0	0	0	1	0	0
h	0	0	1	0	0	0	0	0	0	0
i	0	0	0	0	0	0	0	0	0	1
j	0	0	0	0	1	0	0	0	0	0

From adjacency matrix, we have directed graph as following:



(a) Show the execution of the following algorithm. Mark each vertex with its dicover time (d) and finish time (f); (d/f).

```
DFS(G)
for each vertex u ∈ V[G]
        do color[u] ← WHITE
           π[u] ← NIL
time ← 0
for each vertex u ∈ V[G]
        do if color[u] = WHITE
           then DFS-VISIT(u)

DFS-VISIT(u)
color[u] ← GRAY
time ← time+1
d[u] ← time
for each v ∈ Adj[u]
        do if color[v] = WHITE
           then π[v] ← u
                      DFS-VISIT(v)
color[u] ← BLACK
f[u] ← time ← time + 1
```

(a)

Execution step:

Assumution: The nodes has been sorted by node's name in V[G]

actions:

0: whited out all nodes and ~~sort by node's name~~, time = 0

1: greyed out nodes and set times (d)

a → b → c → d → e → j

1 2 3 4 5 6

blacked out nodes and set finish time (f)

j → e

7 8

2: greyed out nodes and set times: h 9 blacked out h 10

3. greyed out: i, blacked out: i 12

4. black out d 11

5. blackout 13 c 14

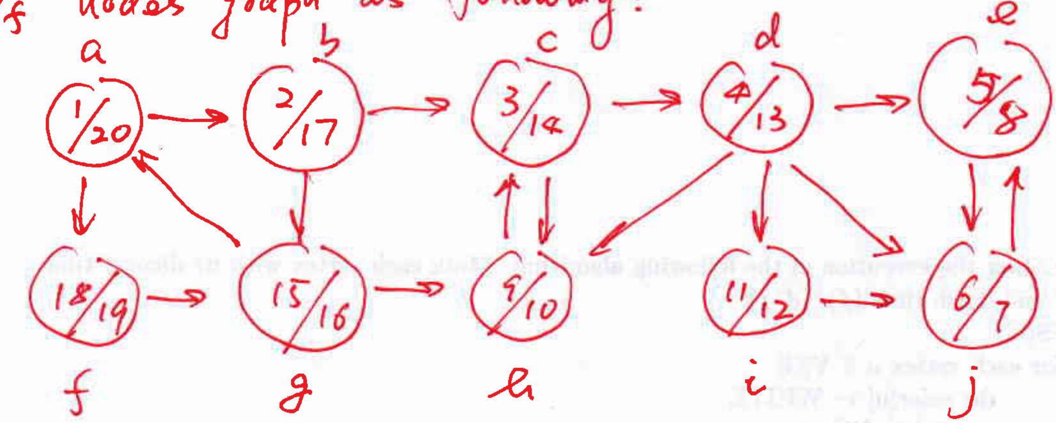
6. greyed out g 15 blacked out g 16

7. black out b

8. greyed out f 17, blacked out f 19

9 Black out a 18 20

d/f nodes graph as following:



(b) Show the execution of the following algorithm.

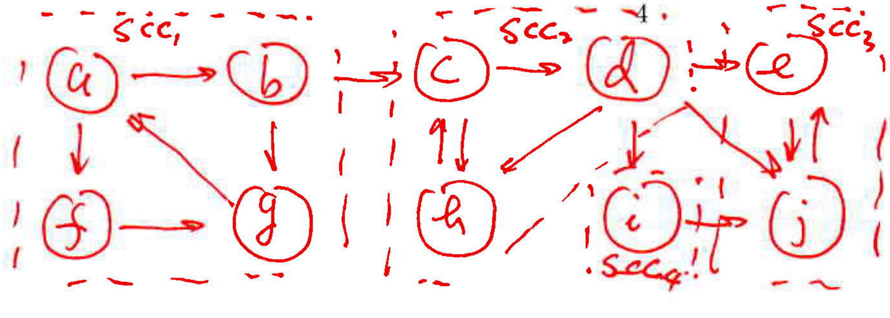
TOPOLOGICAL-SORT(G)
1 call DFS(G) to compute finishing times *f[v]* for each vertex v
2 as each vertex is finished, insert it onto the front of a linked list
3 **return** the linked list of vertices

(c) Given directed graph $G = (V, E)$, a **strongly connected component (SCC)** of G is a maximal set of vertices $C \subseteq V$ such that for all $u, v \in C$, both $u \leadsto^b v$ and $v \leadsto^b u$. Find the SCCs in the graph.

(b) ∵ The new node is always added before the head of the linkedList.
∴ The earlyest node would be the ~~first~~ end node
The last node would be the begining node.
∵ The node insertion order is by the finish time.
∴ The order of nodes is like:

Finish time	20	19	17	16	14	13	12	10	8	7
Linklist nodes:	a →	f →	b →	g →	c →	d →	i →	h →	e →	j

(c) From the directed graph, we can see:
∵ $(a \to b,\ b \to a)$, $(a \to g,\ g \to a)$, $(b \to g,\ g \to b)$, $(a \to f,\ f \to a)$, $(f \to g,\ g \to f)$, $(b \to f,\ f \to b)$
∴ $SCC_1 = \{a, b, f, g\}$
∵ $(c \to d,\ d \to c)$, $(c \to h,\ h \to c)$, $(d \to h,\ h \to d)$
∴ $SCC_2 = \{c, d, h\}$
∵ $(e \to j,\ j \to e)$
∴ $SCC_3 = \{e, j\}$
∴ $SCC_4 = \{i\}$. The SCCs in the graph as following:



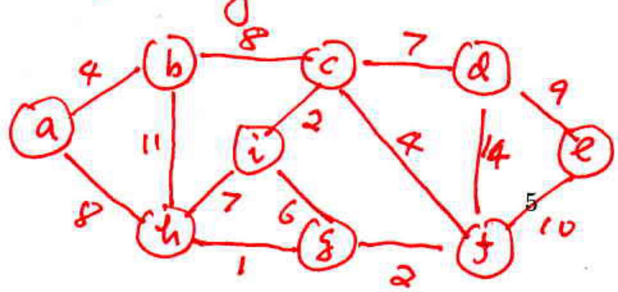
2. (25 points) Given the following graph as described in the adjacency matrix,

	a	b	c	d	e	f	g	h	i
a	0	4	0	0	0	0	0	8	0
b	4	0	8	0	0	0	0	11	0
c	0	8	0	7	0	4	0	0	2
d	0	0	7	0	9	14	0	0	0
e	0	0	0	9	0	10	0	0	0
f	0	0	4	14	10	0	2	0	0
g	0	0	0	0	0	2	0	1	6
h	8	11	0	0	0	0	1	0	7
i	0	0	2	0	0	0	6	7	0

Show the execution of the following algorithm.

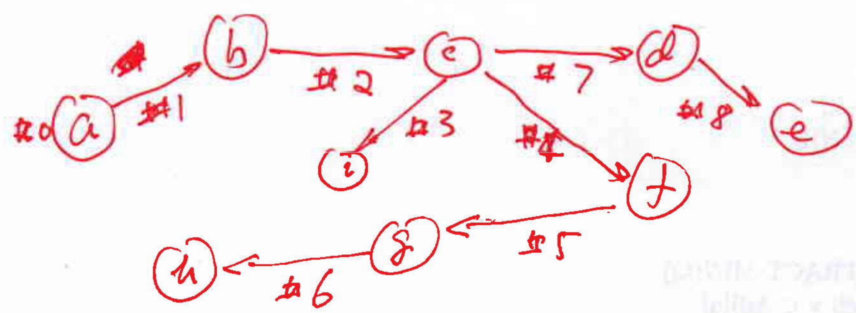
```
MST-PRIM(G,w,r)
for each u ∈ V[G]
        do key[u] ← ∞
           π[u] ← NIL
key[r] ← 0
Q ← V[G]
while Q ≠ ∅
        do u ← EXTRACT-MIN(Q)
           for each v ∈ Adj[u]
               do if v ∈ Q and w(u,v) < key[v]
                     then π[v] ← u
                          key[v] ← w(u,v)
```

From the adjacency matrix, we have the nodes graph with weights as following:



Execution steps	Actions
#0.	choose a node 'a' as starting node, add a to V_A
#1.	find a light edge crossing cut $(V_A, V-V_A) = 4$. add b to V_A.
#2.	the light edge $(V_A, V-V_A) = 8$. here, I choose c and add c to V_A
#3.	the light edge $(V_A, V-V_A) = 2$. add i to V_A
#4.	the light edge $(V_A, V-V_A) = 4$. add f to V_A
#5.	the light edge $(V_A, V-V_A) = 2$, add g to V_A
#6.	the light edge $(V_A, V-V_A) = 1$, add h to V_A
#7.	the light edge $(V_A, V-V_A) = 7$. add d to V_A
#8.	the light edge $(V_A, V-V_A) = 9$. add e to V_A

The execution path as following:

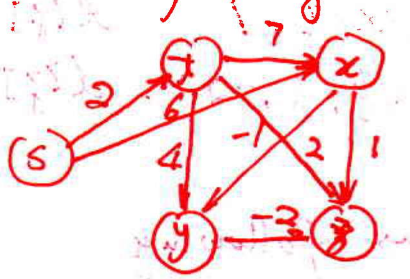


3. (25 points) Given the following graph as described in the adjacency matrix, show the execution of the following algorithm.

	s	t	x	y	z
s	0	2	6	0	0
t	0	0	7	4	2
x	0	0	0	−1	1
y	0	0	0	0	−2
z	0	0	0	0	0

```
a-shortest-path(V, E, w, s)
topologically sort the vertices in V
initialize-single-source(V, s)
for each vertex u, taken in topologically sorted order
        do for each vertex v ∈ Adj[u]
                do RELAX(u, v, w)
```

From the adjacency matrix, we have the directed nodes graph as flowing:



Execution steps:

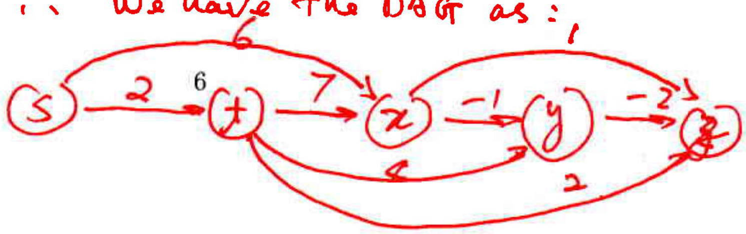
#0

Actions.

use the way of qustion 1(b) to do toplogical sorting.

∵ (z) would be the first reached the finish time. and (y) would be 2nd. and (x) would be 3rd, and so on, the last one is (s).

∴ We have the DAG as:



#1. init-single-source(V, s)

we have d[] as:

d[s]=0. d[t]=∞. d[x]=∞. d[y]=∞. d[z]=∞

#2. RELAX(s, t, 2) when s is u node, t is v node. 2 is weight

we have d[t]=2

#3. ~~RELAX(t, x, ...) and~~ RELAX(s, x, 6)

we have d[x]=6

~~RELAX(t, y, ...) and RELAX(...)~~

~~we have d[y]=~~

#4 RELAX(t, x, 7), d[x] don't change

#5 RELAX(t, y, 4). d[y]=2+4=6

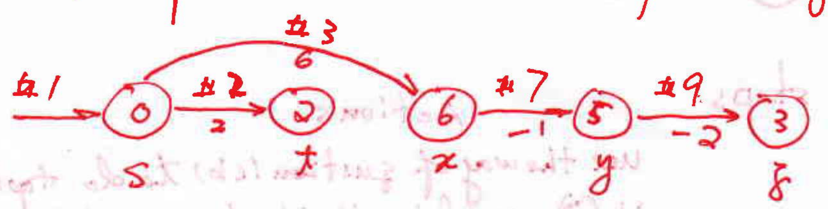
#6 RELAX(t, z, 2), d[z]=2+2=4

#7 RELAX(x, y, -1), d[y]=6-1=5

#8 RELAX(x, z, 1), d[z] no change

#9 RELAX(y, z, -2), d[z]=5-2=3

Effective Execution path with d[] as following



3. (20 points) Given the following graph as described in the adjacency matrix, show the execution of the following algorithm.

	s	x	y	z
s	0	10	5	0
x	0	0	3	0
y	0	4	0	1
z	0	2	0	0

```
DIJKSTRA(G,w,s)
INITIALIZE-SINGLE-SOURCE(G,s)
S ← ∅
Q ← V[G]
while Q ≠ ∅
      do u ← EXTRACT-MIN(Q)
            S ← S ⋃ {u}
            for each vertex v ∈ Adj[u]
                  do RELAX(u,v,w)
```

Init: $d[s]=0$

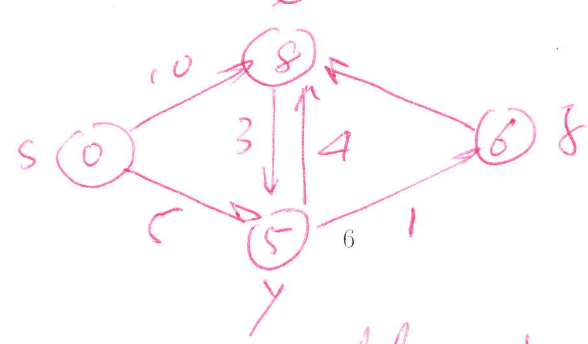
#1. $s \to y$, $d[y]=5$. #2. $s \to x$ $d[x]=10$

#3. $y \to x$ $\because 5+4=9<10$ $\therefore d[x]=9$

#4 $y \to z$ $d[z]=6$. #5 $x \to y$ $\because 9+3=12>5$ $\therefore d[y]=5$

#6 $x \to z$, $\because 9+2=11>6$ $\therefore d[z]=6$

#7 $z \to x$ $\because 6+2=8<9$, $d[x]=8$



Order of adding to S: s, y, z, x