Exercise:

1. Explain the efficiency of BST on the following operations.

	average	worst
① min	$\Theta(h)=\Theta(\lg n)$	$\Theta(n)$
② max	$\Theta(h)=\Theta(\lg n)$	$\Theta(n)$
③ search	$\Theta(h)=\Theta(\lg n)$	$\Theta(n)$

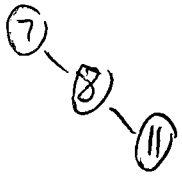
2. build a BST for the following data

7 8 11 2 6 9 20

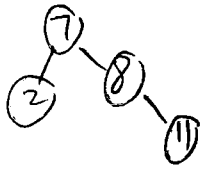
step 1:

step 2:

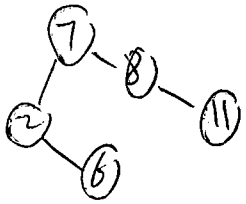
step 3:



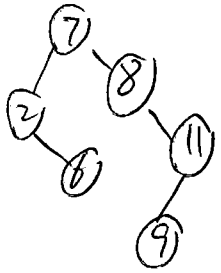
step 4:



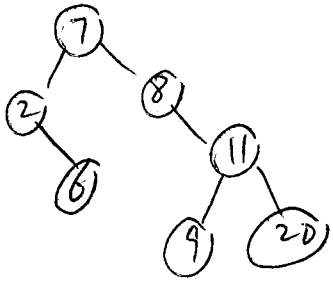
step 5:



step 6

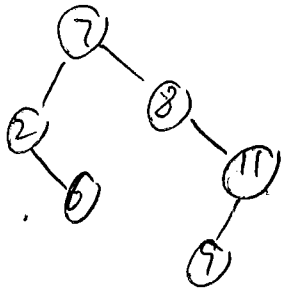


step 7

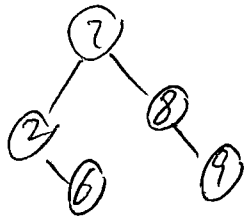


3. In the BST from problem 2, delete the following nodes

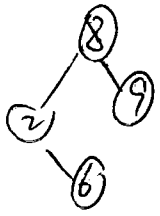
① delete ⑩ : just delete it.



② delete ⑪ : move ⑨ up



③ delete ⑦ : find successor of ⑦



4. find the optimal grouping of the following matrices

P_i	0	1	2	3	4	5
	20	40	15	50	20	35

$$m[i,j] = \min_{i \le k < j} \left(m(i,k) + m(k+1,j) + P_{i-1} * P_k * P_j \right)$$

record the k in $S[i,j]$

M

	1	2	3	4	5
1	0	1200	4200	33000	47000
2		0	3000	27000	46500
3			0	15000	25500
4				0	35000
5					0

S

	1	2	3	4	5
1	0	1	2	2	4
2		0	2	2	2
3			0	3	4
4				0	4
5					0

5. Align the following two sequences given match = +2, mismatch = -1, gap = ~~-5~~ -2

X : AACCT

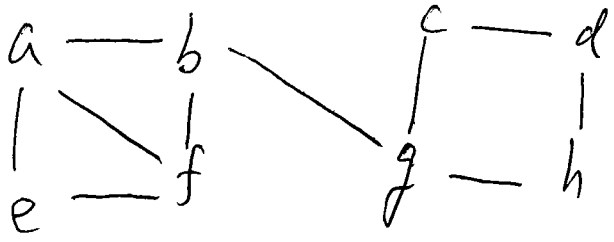
Y : GACGTA

	-	G	A	C	G	T	A
-	0	-2	-4	-6	-8	-10	-12
A	-2	-1	0	-2	-4	-6	-8
A	-4	-3	1	-1	-3	-5	-4
C	-6	-5	-1	3	1	-1	-3
C	-8	-7	-3	1	2	0	-2
T	-10	-9	-5	-1	0	4	2

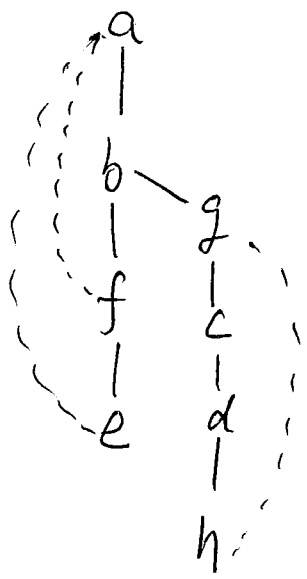
G A C G T A

A A C C T -

6. find the DFS / BFS tree of the following graph starting from a

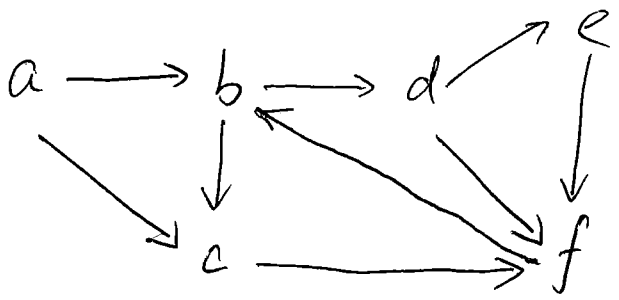


DFS



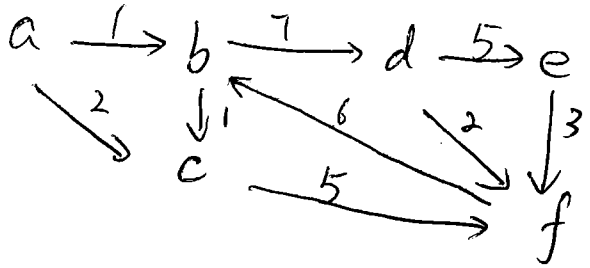
BFS

7. Use appropriate algorithm to find the shortest paths from a to every other node.



Criterion: if negative-weight : Bellman ford

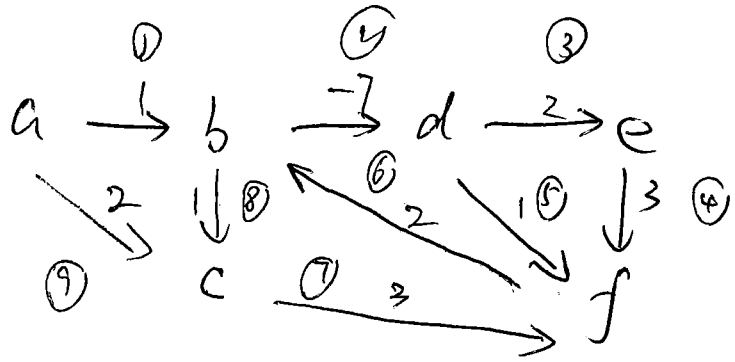
if all positive weight : Dijkstra



Q:

	b	c	d	e	f
a(0,-)	(1,b)	(2,b)	(+∞,-)	(∞,-)	(∞,-)
a(0,-) b(1,b)		(2,c)	(8,b)	(∞,-)	(∞,-)
a(0,-) b(1,b) c(2,c)			d (8,b)	e (∞,-)	f (7,c)
a(0,-) b(1,b) c(2,c) d(8,b)				e (13,d)	f (7,c)

then include f, then e.



Bellman ford:

round 1:

d :	a	b	c	d	e	f
	0	∞	∞	∞	∞	∞
#1 :	0	1	∞	∞	∞	∞
# 2:	0	1	∞	-6	∞	∞
#3:	0	1	∞	-6	2	∞
#4:	0	1	∞	-6	2	5
#5:	0	1	∞	-6	2	-5
#6:	0	-3	∞	-6	2	-5
#8: no change						
#9:	0	-3	2	-6	2	-5

after 6 rounds, the ~~vector~~ doesn't converge.