Problem 1:

Thứ
Ngày . . No.

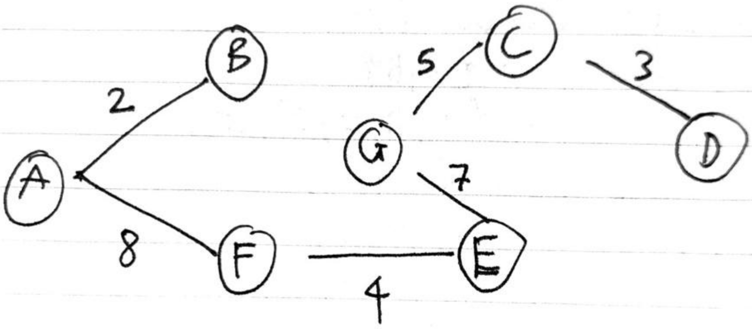
Krustal

edges:

AB = 2
CD = 3
FE = 4
CG = 5
GD = 6
GE = 7
AF = 8
FB = 9
BG = 10
BC = 11
ED = 12
FG = 13

edge	decision	MST
AB	take	{AB}
CD	take	{AB, CD}
FE	take	{AB, CD, FE}
CG	take	{AB, CD, FE, CG}
GD	no	{AB, CD, FE, CG}
GE	take	{AB, CD, FE, CG, GE}
AF	take	{AB, CD, FE, CG, GE, AF}

we have |V| = 7 => MST must have |V|-1 edge, we stop here because we have 6 edges in MST

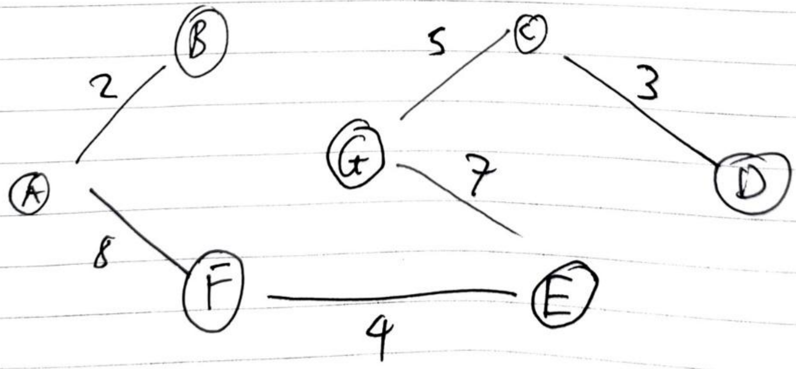


Prim

we take source S = {A}
other vertices V = {B, C, D, E, F, G}

- smallest edge from {A}
  is AB, then S = {A, B}
  V = {C, D, E, F, G}
- smallest edge from {A, B}
  is AF, then S = {A, B, F}
  V = {C, D, E, G}
- and so we have FE then S = {A, B, F, E}
  V = {C, D, G}
- then we have EG then S = {A, B, F, E, G}
  V = {C, D}
- then GC then S = {A, B, F, E, G, C}

finally : CD then S = {A, B, F, E, G, C, D}



Problem 2:

2,

when we start, let consider path value, ~~start~~ start at A

(1), (2), (3), (4), (5)

0 ∞ ∞ ∞ ∞

first loop

$1 \to 2 = 6 < \infty \Rightarrow (2) = 6$

$1 \to 4 = 7 < \infty \Rightarrow (4) = 7$

$2 \to 3 = 6+5 = 11 < \infty \Rightarrow (3) = 11$

$2 \to 4 = 6+8 = 14 > (4) \Rightarrow$ no change

$2 \to 5 = 6 - 4 = 2 < \infty \Rightarrow (5) = 2$

$3 \to 2 = 11 - 2 = 9 > (2) \Rightarrow$ no change

$4 \to 3 = 7 - 3 = 4 < (3) \Rightarrow (3) = 4$

$4 \to 5 = 7 + 9 = 16 > (2) \Rightarrow$ no change

$5 \to 1 = 2 + 2 = 4 > (1) \Rightarrow$ no change

$5 \to 3 = 2 + 7 = 9 > (3) \Rightarrow$ no change

Then after first loop:

(1), (2), (3), (4), (5)

0 6 4 7 2

2nd loop

$1 \to 2 = 0 + 6 = (2) \Rightarrow$ no

$1 \to 4 = 0 + 7 = (4) \Rightarrow$ no

$2 \to 3 = 6 + 5 > (3) \Rightarrow$ no

$2 \to 4 = 6 + 8 > (4) \Rightarrow$ no

$2 \to 5 = 6 - 4 = (5) \Rightarrow$ no

$3 \to 2 = 4 - 2 = 2 < (2) \Rightarrow (2) = 2$

$4 \to 3 = 7 - 3 = (3) \Rightarrow$ no

$4 \to 5 = 7 + 9 = 16 > (5) \Rightarrow$ no

$5 \to 1 = 2 + 2 > (1) \Rightarrow$ no

$5 \to 3 = 2 + 7 > (3) \Rightarrow$ no

(1), (2), (3), (4), (5)

0 2 4 7 2

3rd		4.th	
1	0	1	0
2	2	2	2
3	4	3	4
4	7	4	7
5	-2	5	-2

Problem 3:

3, vertices: A, B, C, D, E, F, G

start:

A	0
B	∞
C	∞
D	∞
E	∞
F	∞
G	∞

take A:

A	0	
B	∞	
C	1	A
D	2	A
E	∞	
F	∞	
G	∞	

take C:

A	0	
B	3	C
C	1	A
D	2	A
E	4	C
F	∞	
G	∞	

take D:

A	0	
B	3	C
C	1	A
D	2	A
E	4	C
F	∞	
G	3	D

take G:

A	0	
B	3	C
C	1	A
D	2	A
E	4	C
F	4	G
G	3	D

take B:

A	0	
B	3	C
C	1	A
D	2	A
E	4	C
F	4	G
G	3	D

choose E :

A	0	
B	3	C
C	1	A
D	2	A
E	4	C
F	4	G
G	3	D

finally F :

A	0
B	3
C	1
D	2
E	4
F	4
G	3

stop here because this is our destination

F: 4 => shortest path from A to F is

$\boxed{A \rightarrow D \rightarrow G \rightarrow F}$

Problem 4:

4) BFS:

Q = [A]

visited = {A}

take A, add B, C, D

Q = [B, C, D]

visited = {A, B, C, D}

take B, add E, F

Q = [C, D, E, F]

visited = {A, B, C, D, E, F}

take C, no add, Q = [D, E, F]

take D, no add, Q = [E, F]

take E, no add, Q = [F]

take F, no add, Q = ∅

A → B → C → D → E → F

DFS:

Stack = [A]

pop A, push D, C, B, stack = [D, C, B]

pop B, push F, E, stack = [D, C, F, E]

pop E, no push, Stack = [D, C, F]

pop F, ——, Stack = [D, C]

pop C, ——, Stack = [D]

pop D, ——, Stack = ∅

so A → B → E → F → C → D

Problem 5:

Pseudocode:

rod-cutting-wCost(price, n, c)

create array dp[0..n]

dp[0] = 0

```
for j = 1 to n
    best = price[j]
    for i = 1 to j - 1
        best = max(best, price[i] + dp[j - i] - c)
    dp[j] = best
return dp[n]
```

This problem is more complicated than the original problem which is just cutting rod, because when we cut now we have to think whether cost < revenue or not.

We set `best = price[j]` because maybe no cut is better than cut, because cut may make cost > revenue.