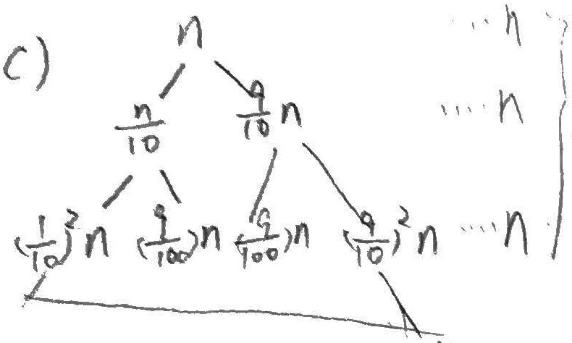
1.

1a) $\Theta(\log_3 n)$ ✓

1b) $\Theta(n)$ ✓

1c)



lg n $\Theta(n \lg n)$ ✓

1d) 令 $n = 2^{2^k}$ 則 $T(2^{2^k}) = 2T(2^{2^{k-1}}) + \lg 2^{2^k} = 2T(2^{2^{k-1}}) + 2^k$

令 $a_k = T(2^{2^k})$，則 $a_k = 2a_{k-1} + 2^k$ → $k = \lg\lg n$ 代入.

$a_k = 2a_{k-1} + 2^k$
$= 2(2a_{k-2} + 2^{k-1}) + 2^k$
$= 2^2 a_{k-2} + 2^k + 2^k$
$\Rightarrow 2^k a_0 + k \cdot 2^k$

$\Rightarrow 2^{\lg\lg n} a_0 + \lg\lg n \cdot 2^{\lg\lg n}$
$= \lg n + \lg\lg n \cdot \lg n$
$= O(\lg n \lg\lg n)$

2.

(a) Yes   (b) No. $\begin{array}{c} 385 \\ 372 \quad 412 \end{array}$ (c) Yes

3

i ↓ \ j →	a_1	a_2	a_3	a_4	a_5
b_1	↖1	←1	←1	←1	↖1
b_2	↑1	↖2	←2	←2	←2
b_3	↑2	↑2	↑2	↑2	↑2
b_4				↖3	
b_5					↖4
b_6					

① 若 a_i 和 b_j 相同 ⇒ $len[i,j] = len[i-1, j-1] + 1$ ✓

3A ⇒ 左上角 $len + 1$

② 觀察遞迴若↖條件成立 ⇒ 往左上角遞迴

3B ⇒ LCS_OUTPUT(A, prev, i-1, j-1) ✓

③ 印該格之a或b之字元

$\left\{ \begin{array}{l} \text{print } a_i \\ \text{print } b_j \end{array} \right.$ 皆可 ✓

4

已知

① C cities, R roads

② All road 只有單向 ($u \rightarrow v$, then $u \not\leftarrow v$)

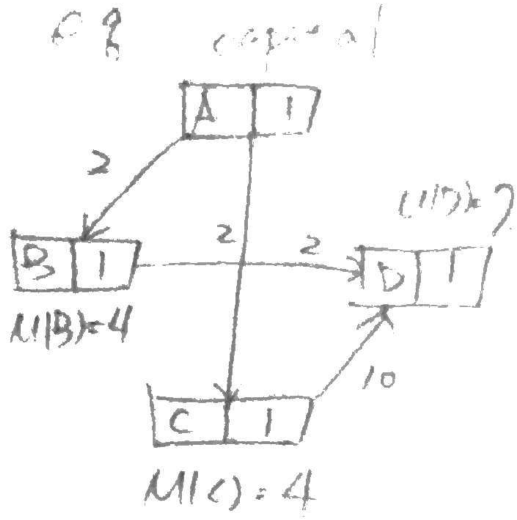
③ road 沒有cycle (不會繞回起点)

④ 存在一個点都可以到各個city

⑤ $C(u)$: 待在該city天數 ($\forall c \in C(u)$ 皆 >0)

⑥ $r(u,v)$: u到v之路程天數 ($\forall r \in r(u,v)$ 皆 >0)

⑦ 用 $M(v)$ 表minimum number of day.



(a) 將問題視為一個有向圖且點和边具有weight，找出single vertex to all vertex之問題

① 先找出各城市間的topological sort (用DFS) ⇒ $O(|V|+|E|)$

② 对topological上每個点 依序做relax ⇒ $O(|V|)$

relax step:

1) if $(M(v) > C(u)+r(u,v))$ then

2) $M(v) = C(u)+r(u,v)$ 考慮点weight + 路weight

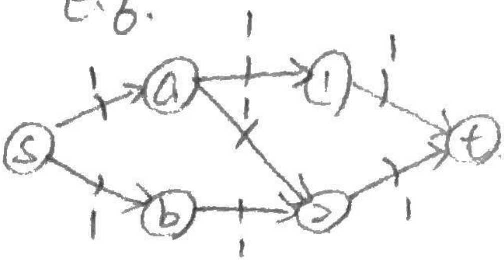
(b) 使用以上演算法耗時 $O(|V|+|E|) + O(|V|) = O(|V|+|E|)$

因為題目給C和R，全部trace一次至少也要 $O(|V|+|E|)$

⇒ 所以此演算法已達最佳.

5. ① C classes, R classroom
② M個 (c,r) = class c 可到 r classroom
③ classes 只能到有assign 的classroom

e.g.



(a) 利用Ford-Fulkerson algorithm求解 ⇒ 可解
①令兩個假想點ⓢ, ⓣ
- ⓢ連向所有classes且capatity為 1
- ⓣ被所有classroom所連且capatity為 1

②再定義 (c,r) 的capatity皆為 1
③依序找出augmenting path 直到不能找為止
④最後residual network即為最佳解之圖，max flow為最大可配置量

(b) ① 因所有容量皆為 1，不會有超額配置問題
(即 max flow ≤ min(r的數量, c之數量))
② 但考慮下限問題，若圖中具有負cycle則Ford-Fulkerson無解
(不確定).