1. 使用 tail recursion

```
int fib(int n, int a, int b){
    if(n==0){
        return a;
    }
    return fib(n-1, b, a+b);
}
```

2. (a) $T(n)$

$T(\frac{n}{4})$ $T(\frac{n}{4})$ $T(\frac{2n}{3})$

$\vdots$ $\vdots$ $\vdots$

$\Rightarrow T(n) \leq (1 + \frac{7}{6} + (\frac{7}{6})^2 + \cdots) cn$

$= \frac{(1-(\frac{7}{6})^n)}{1-\frac{7}{6}} cn =$

$= \infty cn = O(n)$

(b) $n = 2^{2^k}$ $T(n) = 2T(\sqrt{n}) + \log n$

$T(n) = T(2^{2^k}) = 2T(2^{2^{k-1}}) + 2^k$

$\Rightarrow 2 \times (2 \times T(2^{2^{k-2}}) + 2^{k-1}) + 2^k \Rightarrow 2^2 \times T(2^{2^{k-2}}) + 2^k + 2^k$

$\vdots\ k$

$2^k \times T(2) + k \times 2^k \Rightarrow O(k \times 2^k) \Rightarrow O(\log n \log\log n)$

3. ∵ heap要是 complete binary tree，最多會有 $\frac{n}{2}$ 個 node 在底部

公式 $\lceil \frac{n}{2^{h+1}} \rceil$ 代入0 符合 最多有 $\frac{n}{2}$ nodes

則 $h=k$. 有 $\lceil \frac{n}{2^{k+1}} \rceil$ 個 node 在高為 k 時,

$h=k+1$ 有 $\lceil \frac{\frac{n}{2^{k+1}}}{2} \rceil \Rightarrow \lceil \frac{n}{2^{k+1+1}} \rceil$

∴ 得證最多有 $\lceil \frac{n}{2^{h+1}} \rceil$ nodes 在高為 h 時

4. 會 overflow，因為會使小數變很大

且 counting sort 的複雜度為 $O(n+k)$，k 太大也會有機會不是

linear time.

5. (a) $LCS(i,j,k) = \begin{cases} 0, & \text{if } i=0 \text{ or } j=0 \text{ or } k=0 \\ LCS(i-1,j-1,k-1)+1 & \text{if } x_i = y_j = z_k \\ \max\{LCS(i-1,j,k), LCS(i,j-1,k), LCS(i,j,k-1)\} & \text{otherwise.} \end{cases}$

$O(nk)$

(b) 如果取 LCS(LCS(x,y),z) 因 LCS 有多組解，不能保證 LCS(x,y) 是對 z 的最佳解

如: x = badly , z = badminton

y = bad school

LCS(x,y) = badl

LCS (LCS(x,y), k) = bad

(C) recurrence:

$$LRS(i,j) = \begin{cases} 0, & \text{if } i=0 \text{ or } j=0 \\ 1+LRS(i-1,j-1), & \text{if } char_i = char_j \text{ and } i \neq j \\ \max(LRS(i,j-1), LRS(i-1,j)), & \text{otherwise} \end{cases}$$

i j		A	T	A	C	T	C	G	A	G
	0	0	0	0	0	0	0	0	0	0
A	0	0	0	1	1	1	1	1	1	1
T	0	0	0	1	1	2	2	2	2	2
A	0	0	0	1	1	2	3	3	3	3
C	0	0	0	1	1	2	3	3	3	3
T	0	0	1	1	1	2	3	3	3	3
C	0	0	1	1	2	2	3	3	3	3
G	0	0	1	1	2	2	3	3	3	4
A	0	1	1	1	2	2	3	3	3	4
G	0	1	1	1	2	2	3	4	4	4

Ans. 4.

(d) recurrence

$$LPS(i,j) = \begin{cases} 0, & \text{if } i=0 \text{ or } j=0 \\ LPS(i-1,j-1)+1, & \text{if } x[i]=y[j] \\ \max\{LPS(i-1,j,k), LPS(i,j-1,k)\}, & \text{otherwise} \end{cases}$$

追蹤

i j	null	c	h	a	r	a	c	t	e	r
null	0	0	0	0	0	0	0	0	0	0
r	0	0	0	0	1	1	1	1	1	1
e	0	0	0	0	1	1	1	1	2	2
t	0	0	0	0	1	1	1	2	2	2
c	0	1	1	1	1	1	2	2	2	2
a	0	1	1	2	2	2	2	2	2	2
r	0	1	1	2	3	3	3	3	3	4
a	0	1	1	3	3	4	4	4	4	4
h	0	1	2	3	3	4	4	4	4	4
c	0	1	2	3	3	4	5	8	5	5

Ans. carac

6. recurrence

$$dp(V,i) = \begin{cases} 0, & j=0 \text{ or } i=0 \\ \max(dp(i-1,j),\ dp(i-1,j-V_i)+w_i), & \text{otherwise} \end{cases}$$

令 dp 為組合出價值為 j 的最小重量。

總 value

item	0	1	2	3	4	5	6	7	8	9	10	11	12
1	0	100	X	X	X	X	X	X	X	X	X	X	X
2	0	100	200	300	X	X	X	X	X	X	X	X	X
3	0	100	200	300	250	350	450	550					
4	0	100	200	300	250	350	400	500					

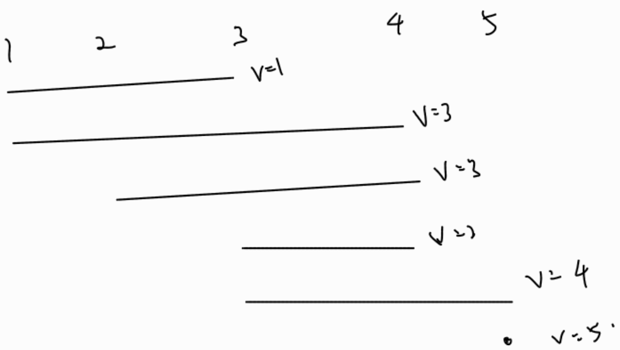
Ans. 7

7. sort by finish time

$H(i) = \max\{ l = \{1,2,\cdots,i-1\} \mid f_l \le s_i \}$

$$A(i) = \begin{cases} 0, & \text{if } i=0 \\ \max\{A(i-1),\ A(H(i))+V_i\}, & \text{otherwise} \end{cases}$$

activity	1	2	3	4	5	6
start	1	1	2	3	3	5
finish	3	4	4	4	5	5
value	1	3	3	2	4	5
H(i)	0	0	0	1	1	5
A(i)	1	3	3	3	5	10



activity: 1,5,6

value: 10