1.

$n, \frac{n}{2}, \frac{n}{4} \cdots, (\frac{n}{2^k})^2 \le M$ 求 k

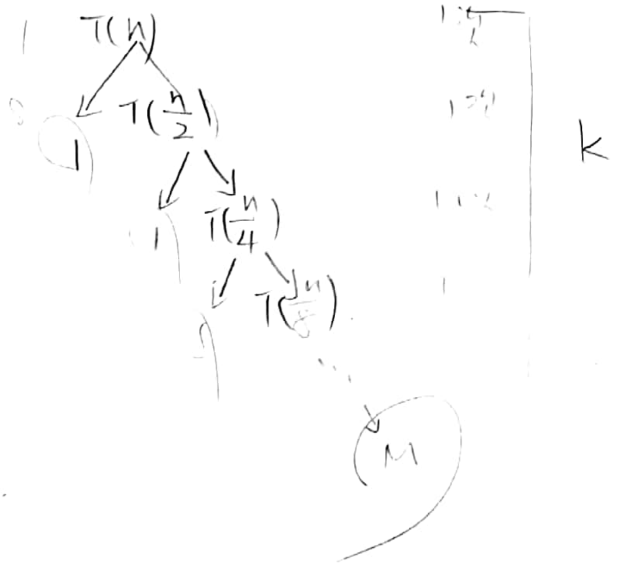
$n^2 (\frac{1}{2})^{2k} \le M$

$n (\frac{1}{2})^k \le M^{\frac{1}{2}}$

$\log n - k \le \frac{1}{2}\log_4 M$

$k \ge \log \frac{n}{\sqrt{M}}$

$T(n) = (8^{\log \frac{n}{\sqrt{M}}} \times M) + \sum_{i=0}^{\log \frac{n}{\sqrt{M}}} 8^i$



2. (a) 令 $a_n = 6a_{n-1} - 8a_{n-2}$

$a_n = x^n$

$x^n = 6x^{n-1} - 8x^{n-2}$

$x^2 - 6x + 8 = 0$

$(x-4)(x-2) = 0$

$x = 4, 2,$

設 $a_n = c_1 4^n + c_2 2^n$

$$\begin{cases} a_0 = c_1 + c_2 = 3 & -① \\ a_1 = 4c_1 + 2c_2 = 8 & -② \end{cases}$$

② − ①×2 :

$2c_1 = 2 \quad c_1 = 1, c_2 = 2$

$a_n = 4^n + 2^{n+1} = T(n)$ ✗✗

(b) $a_n = 4a_{n-1} - 4a_{n-2}$

令 $a_n = x^n$

$x^n = 4x^{n-1} - 4x^{n-2}$

$x^2 - 4x + 4 = 0$

$(x-2)^2 = 0$

$x = 2, 2.$ 相同根

設 $a_n = (c_1 + c_2 n) 2^n$

$$\begin{cases} a_0 = c_1 = 3 \\ a_1 = (c_1 + c_2 \times 1) 2 = 8 \end{cases}$$

$c_2 = 1$

$a_n = (3 + 1 \times n) 2^n = 3 2^n + n 2^n$ ✗✗

3.

```
def func(x, y):
    x, y = list()          令x之長度為m
                             y之長度為n
    dp [m][n] = 0

    for i in range(m):
        for j in range(n):
            if x[i] = y[j]
                c[i][j] = c[i-1][j-1] + 1     長度增長

            elif c[i-1][j] ≥ c[i][j-1]
                c[i][j] = c[i-1][j]

            else
                c[i][j] = c[i][j-1]

    return c.
           ↑
         最長的
```

Time: $O(mn)$

空間 complexity 因為存 $m \times n$ 格子, complexity 為 $O(mn)$