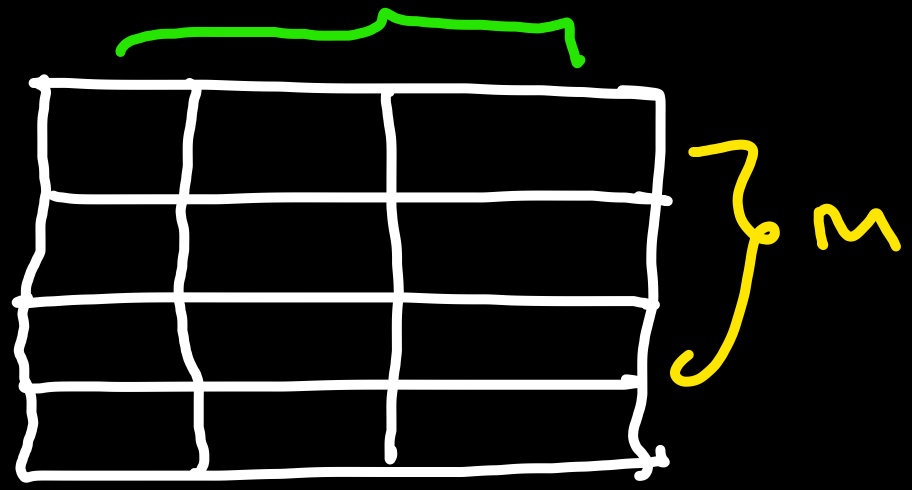
M x N

Baris Kolom

N



M = 4
N = 3
} 4 x 3

Urutan baris - 1, 2, 3, ..., M
Kolom - 1, 2, 3, ..., N

dp(i,j) = Berapa item terbanyak yang bisa dikumpulkan oleh dengklek jika bergerak dari (1,1) ke (i,j) [baris ke-i, kolom-j (i,j) 1 based

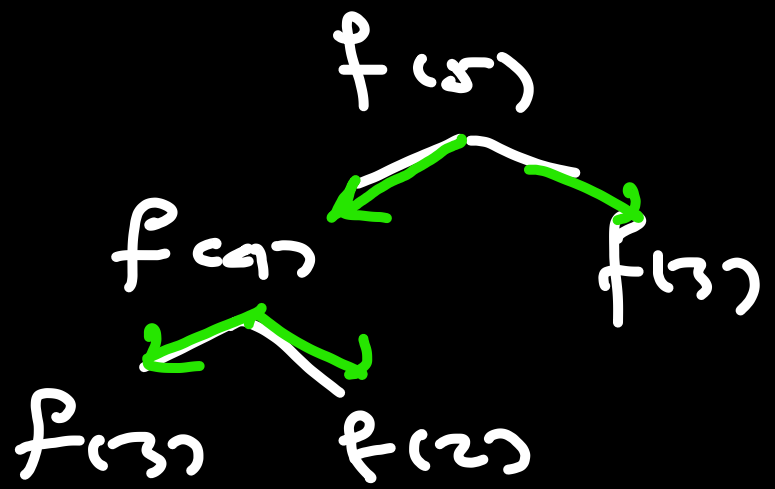
Dynamic Programming
- Top down
- Bottom up

Top down

$$f(n) = f(n-1) + f(n-2)$$

$$f(5) = f(4) + f(3)$$

$$f(4) = f(3) + f(2)$$



Bottom Up : Pre Compute

$f(n) = 3\,f(n-1)$

$f(1) = 3 \cdot f(0) = 3$ → memo[1] = 3

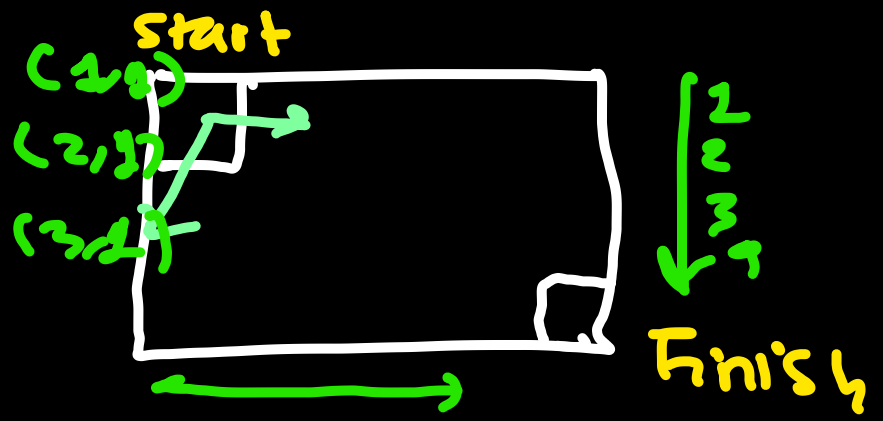
$f(2) = 3 \cdot 3 = 9$ → memo[2] = 9

— — — —

$f(n) = 3\,f(n-1)$

cout << memo[N]

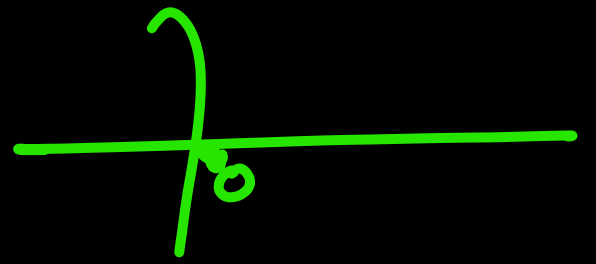
Pre compute

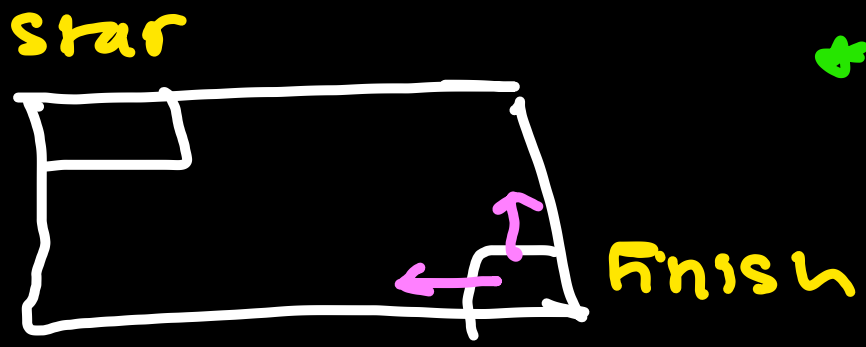


Dari start kita bisa ke kanan atau ke bawah

* Bottom - up

(1,2) → (4,4)





* top down

Dari Finish lalu bisa belok ke kiri atau

- Setiap perak (i,j) gedean item gerak ke mana

dp(i,j)

kiri (i, j-1)

atas (i-1, j)

dp(i,j) = Max (dp(i,j-1), dp(i-1,j)) + item[i][j]

dp(1,2) = max(dp(1,1) , dp(0,2))

dp(1,1) = max(dp(0,1) , dp(1,0))

dp(0,2) = Max (dp(-1,2))

return false

String S =

SONOSONO

0 1 2 3 4 5 6 7

S O N O S O N O

ada berapa dan di mana aja ?

* locate O

O_1 → count banyak string S yang berada index > 1

* locate S

S_4 → count banyak N yang berada index > 4

O_1 — S_q — $N_t = 1$

O_2 — S_q — N_t

for (o)

for (s)

for (n)

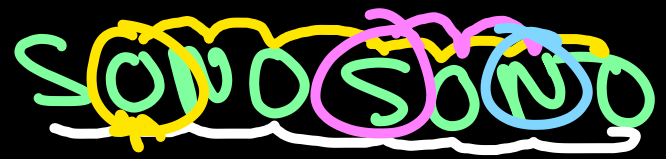
$O(N^3)$

Subtask 1 → $1 \le N \le 200$ → $(200^3) \to \underline{10^7}$ (V)

50 point

Subsoal 2 → 10^7 → $(10^7)^3 \rightarrow 10^{21} \rightarrow$ >1.2 detik

Brute Force



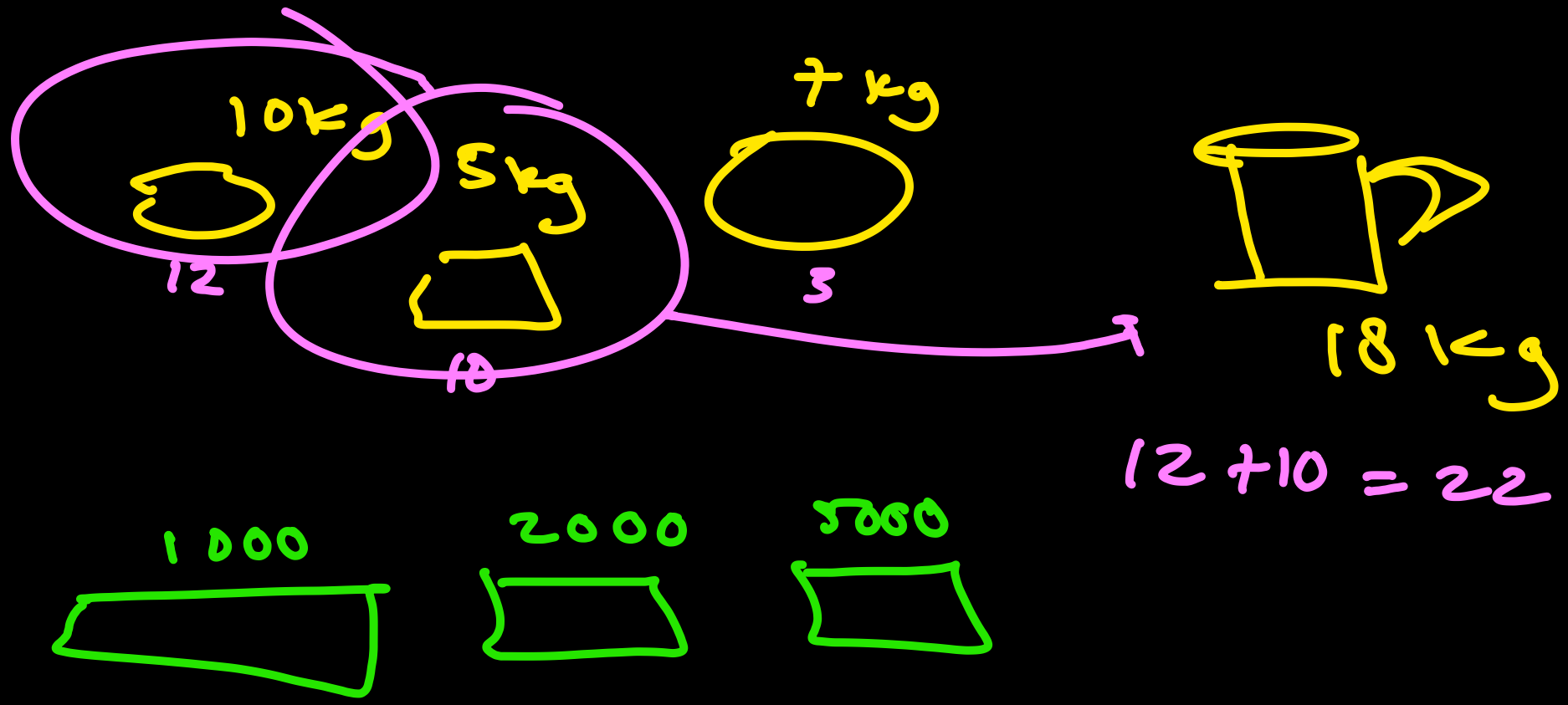
≡

For loop pertama "O"
For loop kedua "S"
For loop ketiga "N"

DP Sering Keluar

Cari aja di GeeksFor Geeks

1. DP Tilling (Pengubinan)
2. DP LIS (Longest Increasing Subsekuens), LCS (Longest Common Subsequences), Counting Subsequences
3. DP Grid / Maze / Matrix
4. DP Knapsack



Berapa minimal lembar uang untuk belanja senilai 10.000?

5. DFS - BFS, Graf Transversal