Tuesday, February 15, 2022 12:29 PM

- Chain Matrix Multiplication. (CMM)

- Making change

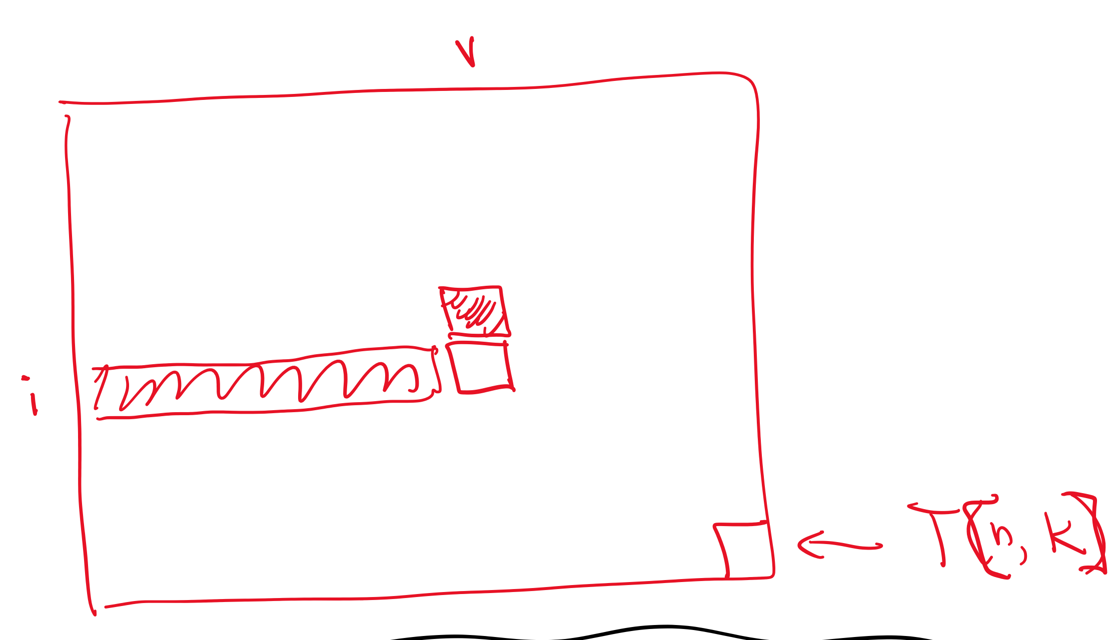
$C = [c_1, c_2, \dots, c_n]$ $K \in \mathbb{N}$.

$T[i,v] = 1/0$ if we can make change for v using denominations $C = [1 \dots i]$ $0 \le v \le K$, $0 \le i \le n$

$$T[i,v] = \max\{T[i-1,v],\ T[i, v-c_i]\}$$

BASE CASE(S): $T[i,0] = 1$ $\forall\, 0 \le i \le n$

$T[0,v] = 0$ $\forall\, 1 \le v \le K$.



Note: $T[v] = 1/0$ if we can make change for v. $0 \le v \le K$

$T[v] = \max_{1 \le i \le n}\{T[v - c_i]\}$

Runtime: $O(nK)$

Knapsack: (0/1 version)

objects	1	2	3	...	n
weights	w_1	w_2	w_3	...	w_n
values	v_1	v_2	v_3	...	v_n

B = capacity.

WANT: $S \subseteq \{1, 2, \dots, n\}$

$\sum_{i \in S} w_i \le B$

$\sum_{i \in S} v_i$ is max.

$T[i,b]$ = max value we can get from objects $\{1, 2, \dots, i\}$ and capacity b.

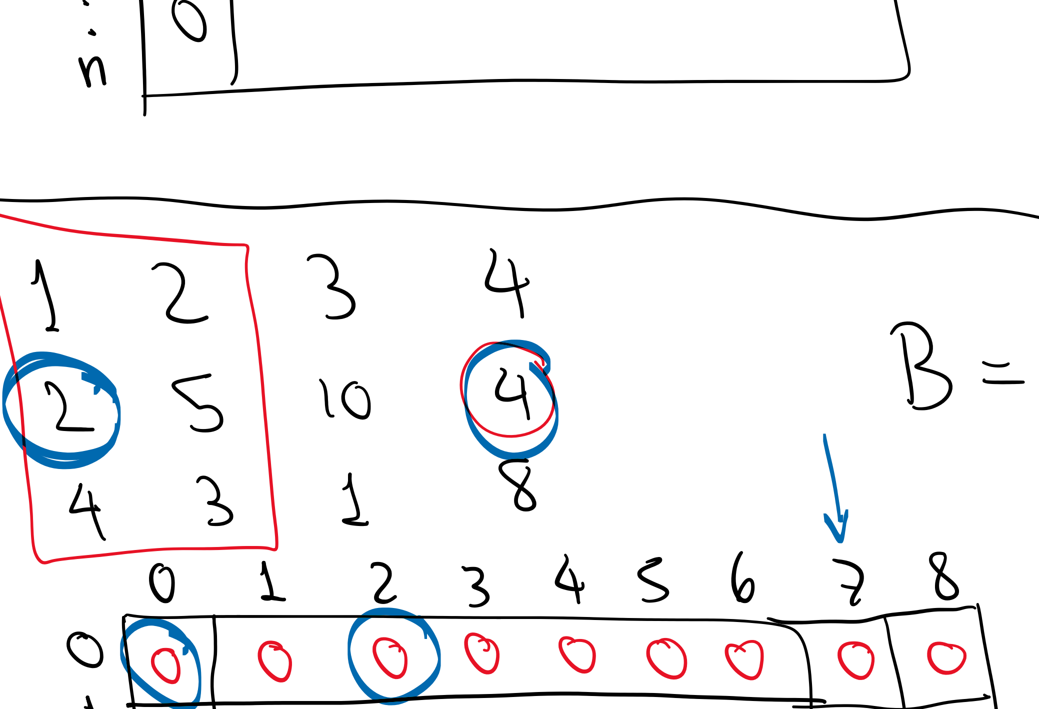
$0 \le i \le n$, $0 \le b \le B$.

$$T[i,b] = \max\{T[i-1,b];\ v_i + T[i-1, b-w_i]\}$$

for $w_i \le b$.

BASE CASES: $T[i,0] = 0$ $\forall\, 0 \le i \le n$

$T[0,b] = 0$ $\forall\, 1 \le b \le B$.



	1	2	3	4
W	2	5	10	4
V	4	3	1	8

$B = 8$

	0	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0	0
1	0	0	4	4	4	4	4	4	4
2	0	0	4	4	4	4	4	7	7
3	0	0	4	4	4	4	4	7	7
4	0	0	4	4	8	8	12	12	12
					0			4+0	

$T[1,2] = \max\{T[\underset{i-1=0}{0}, 2];\ \underset{v_1}{4} + T[\underset{i-1=0}{0}, \underset{\substack{b-w_1 \\ 2-2=0}}{0}]\}$

$T[2,5] = \max\{\underset{4}{T[1,5]};\ \underset{3}{3 + T[1, \overset{0}{\cancel{b-w_2}}]}\}$

Runtime $O(nB)$

$= 2^{\log_2 B}$

size of input is $\log_2 B$!!!

Knapsack (infinitely many copies)

$T[i,b]$ = max profit ... (same table!)

$$T[i,b] = \max\{T[i-1,b];\ v_i + T[i, b-w_i]\}$$

Knapsack-search (0/1)

1	2	...	n
w_1	w_2	...	w_n
v_1	v_2	...	v_n

B capacity

g goal.

WANT: $S \subseteq \{1, 2, \dots, n\}$ s.t.:

$\sum_{i \in S} w_i \le B$

$\sum_{i \in S} v_i \ge g$

Knapsack $\longrightarrow$ Knapsack-search.

run your black-box, compare to g.

Knapsack-search $\longrightarrow$ Knapsack.

goal: $V = \sum_{i=1}^{n} v_i$

goal: $V \to V-1 \to V-2 \to \dots \to G$

in $O(V)$ rounds we get the optimal set.

Binary search: $O(\log(V))$ rounds.

Back to Making change:

Knapsack-search:

$w_i = v_i$ $\forall\, 1 \le i \le n$ $B = g$

Let S be a solution:

$$B \ge \sum_{i \in S} w_i = \sum_{i \in S} v_i \ge g$$