COMBINATORIAL OPTIMIZATION

KNAPSACK PROBLEM

n objects

weights $w_1 \; \dots \; w_n$
values $v_1 \; \dots \; v_n$
weight limit W

} positive reals

NOTATION

$[n] = \{1, \dots, n\}$
$[3] = \{1, 2, 3\}$
$[1] = \{1\}$
$[0] = \emptyset$

objective function — "subject to" bar — constraint

$$\boxed{\max_{I \subseteq [n]} \left\{ \sum_{i \in I} v_i \;\middle|\; \sum_{i \in I} w_i \le W \right\}}$$

can solve this in 2^n trials
(try every subset)
"BRUTE FORCE"

$O(2^n)$ additions, comparisons of reals

exponential in n

Sol'n if weights $\in \mathbb{N}$

Thm $O(nW)$ steps suffice

♣ **brain**

$$m[i,j] = \max_{I \subseteq [i]} \left\{ \overbrace{\sum_{k\in I} v_k}^{\text{objective function}} \ \overset{\text{"subject to" bar}}{\Big|} \ \overbrace{\sum_{k\in I} w_k \le j}^{\text{constraint}} \right\}$$

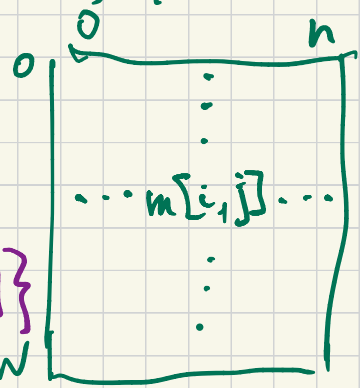
pick only items from $[i]$

weight limit j

$i = 0, \dots, n$

$j = 0, \dots, W$

$(n+1)\times(W+1)$ array of problems



$m[0,j] = 0$

$m[i,0] = 0$

$i,j \ge 1$

♥ **heart**

$$m[i,j] = \max \{ \underbrace{v_i + m[i-1, j-w_i]}_{i \in I}, \underbrace{m[i-1,j]}_{i \notin I} \}$$

♣ $$m[i,j] = \max_{I \subseteq [i]} \left\{ \sum_{k \in I} v_k \;\middle|\; \sum_{k \in I} w_k \le j \right\}$$

♡ $$m[i,j] = \max\{v_i + m[i-1, j-w_i],\, m[i-1,j]\}$$

for $i := 0$ **to** n
    $m[i,0] := 0$
for $j = 0$ **to** W
    $m[0,j] := 0$

} initialization

for $i = 1$ **to** n
    for $j = 1$ **to** W
        if $j < w_i$ **then** $m[i,j] := m[i-1,j]$
        else ♡

return $m[n,W]$

Analysis : array correct b/k { correct init. / correct ♡ }

Cost : $(n+1) \times (W+1)$ steps

each step : $O(1)$ operations < reals / integers: ± 1

DYNAMIC PROGRAMMING