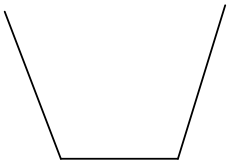
# Knapsack Problem - using greedy approach

| object | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| profit | 10 | 5 | 15 | 7 | 6 | 16 | 3 |
| weight | 2 | 3 | 5 | 7 | 1 | 4 | 1 |



max capacity
$m = 15$

want to fill bag,
sell for profit

Constraint:

$$\sum x_i w_i \leq 15$$

$x_i$: object, $w_i$: weight

total weight of objects in bag cannot exceed 15

objective:

$$\sum x_i w_i \longrightarrow \max$$

maximum profit from selling items in bag

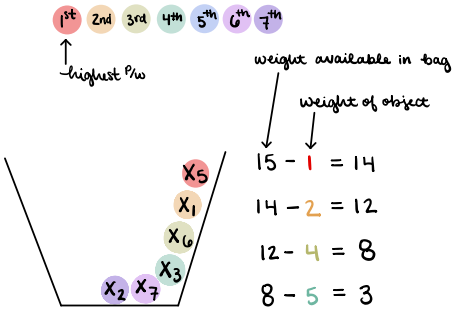
## ONE optimal solution:

Select max profit 16, 15, 10... nope

Consider profit per weight instead!

| object | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| profit | 10 | 5 | 15 | 7 | 6 | 16 | 3 |
| weight | 2 | 3 | 5 | 7 | 1 | 4 | 1 |
| P/W | 5 | 1.3 | 3 | 1 | 6 | 4 | 3 |

1. Select based on highest P/W




2.

$$(\overset{1}{X_1}\ \overset{1.3}{X_2}\ \overset{3}{X_3}\ \overset{0}{X_4}\ \overset{1}{X_5}\ \overset{1}{X_6}\ \overset{1}{X_7})$$

$X_4$: not carried because reached max weight

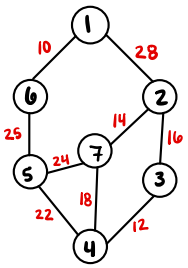
3. Check if optimal solution

$$\sum x_i w_i \leq 15$$

$$\text{capacity} = [(1 \times 2) + (1.3 \times 3) + (1 \times 5) + (0 \times 7) + (1 \times 1) + (1 \times 4) + (1 \times 1)]$$

(1 = how much of object is included; 2 = weight)

$$= 15 \leq 15 \checkmark$$



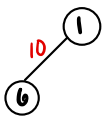

## Prim's Algorithm

$V' = 7$

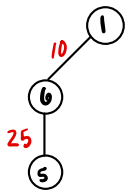
$E' = |V| - 1 = 6$

↳ # of edges in a spanning tree

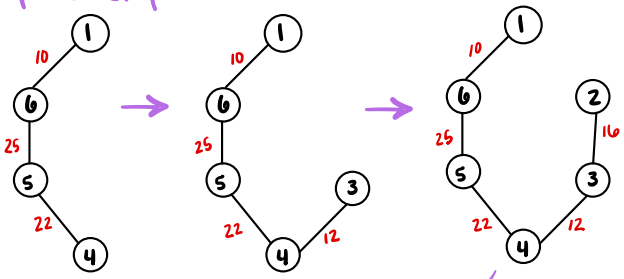
1. Choose min-cost edge

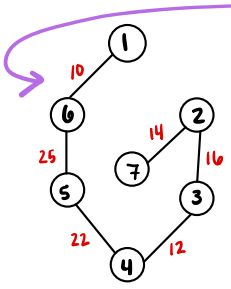



2. Choose min-cost edge attached to ALREADY SELECTED vertices (1 and 6)




3. repeat step 2 ...




$E'$ has to be 6 so stop here!

10+25+22+12+16+14 = MIN COST SPANNING TREE

# Job Sequencing w/ Deadline (greedy)

Show sequence & Max profit

| Jobs | $J_1$ | $J_2$ | $J_3$ | $J_4$ | $J_5$ | $J_6$ | $J_7$ |
|---|---|---|---|---|---|---|---|
| Profits | 35 | 30 | 25 | 20 | 15 | 12 | 15 |
| deadlines | 3 | 4 | 4 | 2 | 3 | 1 | 2 |

Constraint: deadline → 4 units of time longest wait in chart
Objective: max profit
Assumption: each job can be done in 1 unit of time

1. Select based on highest profit
2. Put in earliest avail time slot if max deadline taken

35 → 3 hours ($J_1$)
30 → 4 hours ($J_2$)
25 → 4 hours ($J_3$)
↳ earliest time slot = 2 hrs
20 → 2 hours ($J_4$)
↳ earliest time slot = 1 hr

0 ($J_4$) 9:00 | 1 ($J_3$) 10:00 | 2 ($J_1$) 11:00 | 3 ($J_2$) 12:00 | 4 1:00

SEQUENCE $J_4 \rightarrow J_3 \rightarrow J_1 \rightarrow J_2$

MAX PROFIT $\sum P_i$ $35 + 30 + 25 + 20 = 110$