9\5\12

Machine Replacement.

Produce one type of item etc.

Now a machine aged t produces x at a cost $c_t(x)$.

A new machine costs A and is available immediately.

current age of machine

$$f_r(i, t) = \min \begin{cases} \min_x \left[A + c_0(x) + f_{r+1}(i+x-d_r, 1) \right] & \text{replace} \\ \min_x \left[c_t(x) + f_{r+1}(i+x-d_r, t+1) \right] & \text{not replace} \end{cases}$$

inventory

Could add the chance to refurbish the machine at a lower cost B, making the age T instead of 0.

Knapsack Problem

Scout packing his\her back pack.

n items.

Item i, weighs w_i and has "value" c_i.

Can take more than one of each item.

[0-1 Knapsack if you can only take 0 or 1 of an item].

Scout can only carry $\leq W$.

Integer programming formulation:

x_j = # of items j to take

maximize

$$c_1 x_1 + c_2 x_2 + \cdots + c_n x_n$$

s.t.:

$$w_1 x_1 + w_2 x_2 + \cdots + w_n x_n \leq W$$

$$0 \leq x_j \qquad j = 1, 2, \ldots n$$

x_j integer

Ignoring integrality

put

$x_{j^*} = \frac{W}{w_{j^*}}$ where

$\frac{c_{j^*}}{w_{j^*}}$ maximizes c_j / w_j

Dynamic Programming Solution

Think of choosing $x_n, x_{n-1}, \ldots x_1$

$f_r(w)$ = max. value from knapsack of capacity w, using items $1, 2, \ldots, r$ only

$$= \max_{\substack{0 \le x_r \le \frac{w}{w_r} \\ \text{integer}}} \left[c_r x_r + f_{r-1}(w - w_r x_r) \right]$$