Fontes principais

1. Cormem T. H.; Leiserson C. E.; Rivest R.: Stein C.. *Introduction to Algorithms*, 3^a edição, MIT Press, 2009

2. Análise de algoritmo - IME/USP (prof. Paulo Feofiloff) **http://www.ime.usp.br/~pf/analise_de_algoritmos**

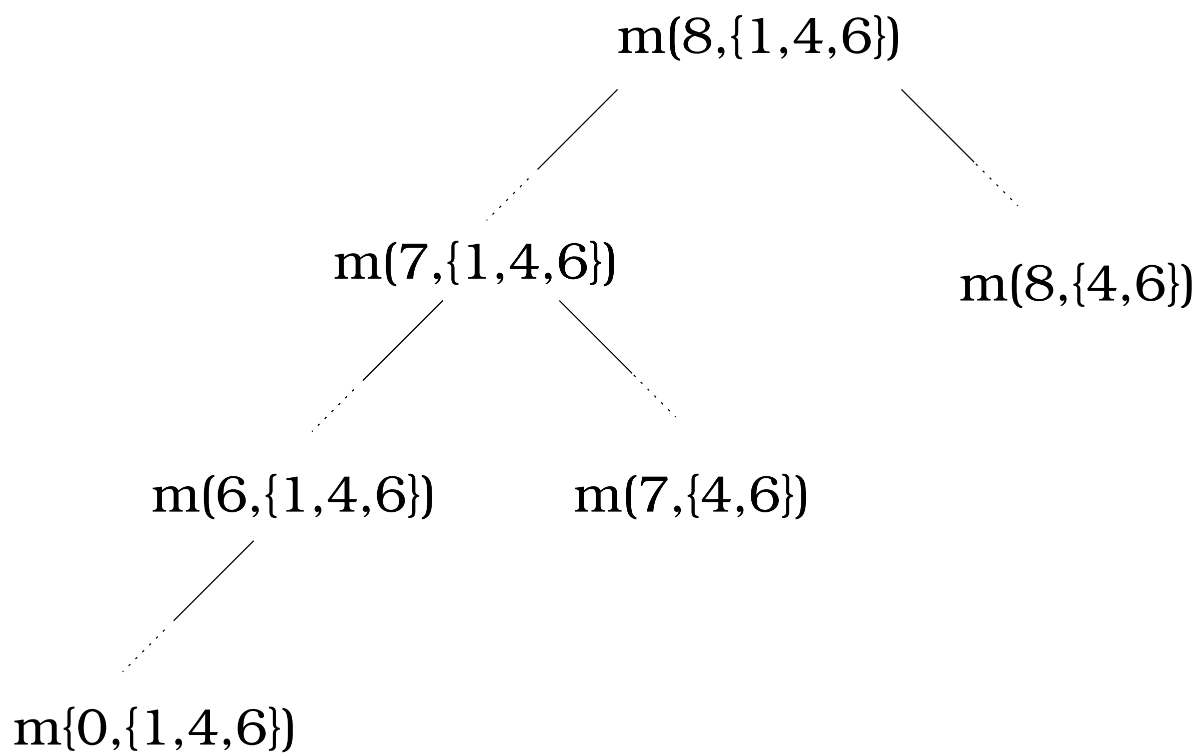
Programação dinâmica (parte 2)

Problema do troco

Dado um valor P, e um conjunto S de n moedas, cada um com um valor c_i, precisamos determinar o número mínimo de moedas para obter a quantidade P.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$



Problema do troco

Perceba que temos três casos base:

$\triangleright$ Se o valor do troco é 0, então temos 0 moedas como solução.
$\triangleright$ Se o valor do troco é negativo (< 0), então não é possível dar o troco.
$\triangleright$ Se o conjunto S é vazio ($\{\}$ ou $\emptyset$), então não é possível dar o troco.

Problema do troco

Definimos a função $m(i, Q)$ como o número mínimo de moedas necessárias para obter uma quantidade Q, usando os i primeiros tipos de moedas ($1 \cdots i$).

A solução para o troco pode utilizar 0 ou mais moedas do tipo i, e assim:

$$m(i, Q) = \begin{cases} m(i-1, Q) & \text{, se não usa moeda tipo} i \\ m(i-1, Q - k \cdot c_i) & \text{, se usa } k \text{ moedas do tipo } i \end{cases}$$

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0								
$c_2 = 4$	0								
$c_3 = 6$	0								

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1							
$c_2 = 4$	0								
$c_3 = 6$	0								

Para o valor 1 utilizar $\{1\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2						
$c_2 = 4$	0								
$c_3 = 6$	0								

Para o valor 2 utilizar $\{1, 1\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3					
$c_2 = 4$	0								
$c_3 = 6$	0								

Para o valor 3 utilizar $\{1, 1, 1\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4				
$c_2 = 4$	0								
$c_3 = 6$	0								

Para o valor 4 utilizar $\{1, 1, 1, 1\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5			
$c_2 = 4$	0								
$c_3 = 6$	0								

Para o valor 5 utilizar $\{1, 1, 1, 1, 1\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0								
$c_3 = 6$	0								

Para o valor 8 utilizar $\{1, 1, 1, 1, 1, 1, 1, 1\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1							
$c_3 = 6$	0								

Para o valor 1 utilizar $\{1\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2						
$c_3 = 6$	0								

Para o valor 2 utilizar $\{1, 1\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3					
$c_3 = 6$	0								

Para o valor 3 utilizar $\{1, 1, 1\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1				
$c_3 = 6$	0								

Para o valor 4 utilizar $\{4\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2			
$c_3 = 6$	0								

Para o valor 5 utilizar $\{1, 4\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3		
$c_3 = 6$	0								

Para o valor 6 utilizar $\{1, 1, 4\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3	4	
$c_3 = 6$	0								

Para o valor 7 utilizar $\{1, 1, 1, 4\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3	4	2
$c_3 = 6$	0								

Para o valor 8 utilizar $\{4, 4\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3	4	2
$c_3 = 6$	0	1							

Para o valor 1 utilizar $\{1\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3	4	2
$c_3 = 6$	0	1	2						

Para o valor 2 utilizar $\{1, 1\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3	4	2
$c_3 = 6$	0	1	2	3					

Para o valor 3 utilizar $\{1, 1, 1\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3	4	2
$c_3 = 6$	0	1	2	3	1				

Para o valor 4 utilizar $\{4\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3	4	2
$c_3 = 6$	0	1	2	3	1	2			

Para o valor 5 utilizar $\{1, 4\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3	4	2
$c_3 = 6$	0	1	2	3	1	2	1		

Para o valor 6 utilizar $\{6\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3	4	2
$c_3 = 6$	0	1	2	3	1	2	1	2	

Para o valor 7 utilizar $\{1, 6\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3	4	2
$c_3 = 6$	0	1	2	3	1	2	1	2	

Para o valor 7 utilizar $\{1, 6\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3	4	2
$c_3 = 6$	0	1	2	3	1	2	1	2	3

Para o valor 8 utilizar $\{1, 1, 6\}$.

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3	4	2
$c_3 = 6$	0	1	2	3	1	2	1	2	3

Para o valor 8 utilizar $\{1, 1, 6\}$ ou $\{4, 4\}$?

Problema do troco

Exemplo: $P = 8$, $S = \{1, 4, 6\}$, $n = 3$

	Q								
Moedas	0	1	2	3	4	5	6	7	8
$c_1 = 1$	0	1	2	3	4	5	6	7	8
$c_2 = 4$	0	1	2	3	1	2	3	4	2
$c_3 = 6$	0	1	2	3	1	2	1	2	3

Para o valor 8 utilizar $\{1, 1, 6\}$ ou $\{4, 4\}$?

Queremos o número mínimo de moedas.

Problema do troco

Para determinar o número mínimo de moedas devemos ter:

$$m(i, Q) = \min_{k=0\cdots Q} \{m(i-1, Q), m(i-1, Q-k\cdot c_i)+k\}$$

Como é o algoritmo de programação dinâmica para resolver este problema?

Problema do troco

```
calcula_total_moedas(P, n, c[])
para i = 0 até n faça m[i][0] = 0
para i = 1 até P faça m[0][i] = ∞
para i = 0 até n faça
    para j = 1 até P faça
        se (c[i] > j) então m[i][j] = m[i − 1][j]
        senão
            se (m[i − 1][j] < m[i][j − c[i]] + 1) então
                m[i][j] = m[i − 1][j]
            senão
                m[i][j] = m[i][j − c[i]] + 1
```

Problema do troco

calcula_total_moedas($P, n, c[]$)

para $i = 0$ **até** n **faça** $m[i][0] = 0$
para $i = 1$ **até** P **faça** $m[0][i] = \infty$
para $i = 1$ **até** n **faça**
  para $j = 1$ **até** P **faça**
    se $(c[i] > j)$ **então** $m[i][j] = m[i-1][j]$
    senão
      se $(m[i-1][j] < m[i][j-c[i]] + 1)$ **então**
        $m[i][j] = m[i-1][j]$
      senão
        $m[i][j] = m[i][j-c[i]] + 1$

Complexidade: $O(n \cdot P)$

Problema da mochila

Problema da mochila

Mochila

Dados dois vetores $m[1 \cdots n]$ e $w[1 \cdots n]$, denotamos por $m \cdot w$ o produto escalar $m[1]w[1] + m[2]w[2] + \cdots + m[n]w[n]$.

Suponha um número inteiro não-negativo W e vetores positivos $w[1 \cdots n]$ e $v[1 \cdots n]$.

Problema da mochila

Uma mochila é qualquer vetor $m[1 \cdots n]$ tal que $m \cdot w \leq W$ e $0 \leq m[i] \leq 1$ para todo i.

Denotamos por W a capacidade da mochila. O **valor** de uma mochila é o produto escalar $m \cdot v$.

Dizemos que uma mochila é **ótima** se ela tem valor máximo.

Problema da mochila booleana

Uma mochila $m[1 \cdots n]$ tal que $m[i] = 0$ ou $m[i] = 1$, para todo i, é chamada de mochila booleana (ou binária ou $0 - 1$).

Exemplo: $n = 4$, $W = 50$

	1	2	3	4
w	40	30	20	10
v	840	600	400	100
m	0	0	0	0

Valor: 0

Peso: 0

Problema da mochila booleana

Uma mochila $m[1 \cdots n]$ tal que $m[i] = 0$ ou $m[i] = 1$, para todo i, é chamada de mochila booleana (ou binária ou $0 - 1$).

Exemplo: $n = 4$, $W = 50$

	1	2	3	4
w	40	30	20	10
v	840	600	400	100
m	1	0	0	0

Valor: 840

Peso: 40

Problema da mochila booleana

Uma mochila $m[1 \cdots n]$ tal que $m[i] = 0$ ou $m[i] = 1$, para todo i, é chamada de mochila booleana (ou binária ou $0 - 1$).

Exemplo: $n = 4$, $W = 50$

	1	2	3	4
w	40	30	20	10
v	840	600	400	100
m	1	0	0	1

Valor: 940

Peso: 50 $\triangleright$ atingiu a capacidade

Problema da mochila booleana

Uma mochila $m[1 \cdots n]$ tal que $m[i] = 0$ ou $m[i] = 1$, para todo i, é chamada de mochila booleana (ou binária ou $0-1$).

Exemplo: $n = 4$, $W = 50$

	1	2	3	4
w	40	30	20	10
v	840	600	400	100
m	0	0	0	0

Valor: 0

Peso: 0

Problema da mochila booleana

Uma mochila $m[1 \cdots n]$ tal que $m[i] = 0$ ou $m[i] = 1$, para todo i, é chamada de mochila booleana (ou binária ou $0 - 1$).

Exemplo: $n = 4$, $W = 50$

	1	2	3	4
w	40	30	20	10
v	840	600	400	100
m	0	1	0	0

Valor: 600

Peso: 30

Problema da mochila booleana

Uma mochila $m[1 \cdots n]$ tal que $m[i] = 0$ ou $m[i] = 1$, para todo i, é chamada de mochila booleana (ou binária ou $0 - 1$).

Exemplo: $n = 4$, $W = 50$

	1	2	3	4
w	40	30	20	10
v	840	600	400	100
m	0	1	1	0

Valor: 1000 $\triangleright$ ótimo

Peso: 50 $\triangleright$ atingiu a capacidade

Subestrutura ótima

Suponha que $m[1..n]$ é uma mochila ótima para o problema (w, v, n, W).

Se $m[n] = 1$
  então $m[1..n-1]$ é uma mochila booleana
    ótima para o subproblema $(w, v, n-1, W - w[n])$
  senão $m[1..n]$ é uma mochila booleana
    ótima para o subproblema $(w, v, n-1, W)$

Simplificação

Problema: Encontrar o **valor** de uma mochila boolean ótima.

$t[i, Y] =$ valor de uma mochila booleana ótima
para (w, v, i, W)

Possíveis valores de $Y = 0, 1, 2, \cdots, W$

Recorrência

$t[i,Y] =$ valor da expressão $m \cdot v$
sujeito à restrição $m \cdot w \leq Y$ (pesquisa operacional)

$t[0,Y] = 0$ para todo Y
$t[i,0] = 0$ para todo i

$t[i,Y] = t[i-1,Y]$ se $w[i] > Y$
$t[i,Y] = \max\{t[i-1,Y], t[i-1,Y-w[i]] + v[i]\}$ se $w[i] \leq Y$

Solução recursiva

Devolve o **valor** de uma mochila booleana ótima para (w, v, n, W).

```
mochila-recursiva(w, v, n, W)
  se n = 0 ou W = 0
      então retorne 0
  se w[n] > W
      então retorne mochila-recursiva(w, v, n − 1, W)
  a = mochila-recursiva(w, v, n − 1, W)
  b = mochila-recursiva(w, v, n − 1, W − w[n]) + v[n]
  retorne max{a, b}
```

Solução recursiva

Consumo de tempo no pior caso é $\Omega(2^n)$, pois o mesmo subproblema é resolvido muitas vezes.

Com programação dinâmica cada subproblema, valor de uma mochila booleana ótima para (w, v, i, Y), é resolvido **apenas uma vez**.

Como é o algoritmo?

Problema da mochila booleana (PD)

Devolve o **valor** de uma mochila booleana ótima para (w, v, n, W).

mochila-pd(w, v, n, W)

para $Y = 0$ **até** W **faça**
    $t[0, Y] = 0$
    para $i = 1$ **até** n **faça**
        $a = t[i-1, Y]$
        se $w[i] > Y$ **então** $b = 0$
                senão $b = t[i-1, Y - w[i]] + v[i]$
        $t[i, Y] = \max\{a, b\}$
retorne $t[n, W]$

Problema da mochila booleana (PD)

Devolve o **valor** de uma mochila booleana ótima para (w, v, n, W).

mochila-pd(w, v, n, W)

para $Y = 0$ **até** W **faça**
  $t[0, Y] = 0$
  para $i = 1$ **até** n **faça**
    $a = t[i-1, Y]$
    se $w[i] > Y$ **então** $b = 0$
      senão $b = t[i-1, Y - w[i]] + v[i]$
    $t[i, Y] = \max\{a, b\}$
retorne $t[n, W]$

Complexidade: $O(n \cdot W)$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

	0	1	2	3	4	5	Y
0	0	0	0	0	0	0	
1	0						
2	0						
3	0						
4	0						
i							

capacidade: 0, objetos: $\{\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

	0	1	2	3	4	5	Y
0	0	0	0	0	0	0	
1	0	0					
2	0						
3	0						
4	0						
i							

capacidade: 1, objetos: $\{\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

	0	1	2	3	4	5	Y
0	0	0	0	0	0	0	
1	0	0	0				
2	0						
3	0						
4	0						
i							

capacidade: 2, objetos: $\{\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

	0	1	2	3	4	5	Y
0	0	0	0	0	0	0	
1	0	0	0	0			
2	0						
3	0						
4	0						
i							

capacidade: 3, objetos: $\{\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

	0	1	2	3	4	5	Y
0	0	0	0	0	0	0	
1	0	0	0	0	500		
2	0						
3	0						
4	0						
i							

capacidade: 4, objetos: $\{1\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0					
3	0					
4	0					

capacidade: 5, objetos: $\{1\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0					
3	0					
4	0					

capacidade: 0, objetos: $\{\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0	0				
3	0					
4	0					

capacidade: 1, objetos: $\{\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0	0	400			
3	0					
4	0					

capacidade: 2, objetos: $\{2\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

	0	1	2	3	4	5	Y
0	0	0	0	0	0	0	
1	0	0	0	0	500	500	
2	0	0	400	400			
3	0						
4	0						
i							

capacidade: 3, objetos: $\{2\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

	0	1	2	3	4	5	Y
0	0	0	0	0	0	0	
1	0	0	0	0	500	500	
2	0	0	400	400	500		
3	0						
4	0						
i							

capacidade: 4, objetos: $\{1\}$ $\triangleright$ objeto 1 vale mais que objeto 2

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0	0	400	400	500	500
3	0					
4	0					

capacidade: 5, objetos: $\{1\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

	0	1	2	3	4	5	Y
0	0	0	0	0	0	0	
1	0	0	0	0	500	500	
2	0	0	400	400	500	500	
3	0						
4	0						
i							

capacidade: 0, objetos: $\{\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0	0	400	400	500	500
3	0	300				
4	0					

capacidade: 1, objetos: $\{3\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

	0	1	2	3	4	5	Y
0	0	0	0	0	0	0	
1	0	0	0	0	500	500	
2	0	0	400	400	500	500	
3	0	300	400				
4	0						
i							

capacidade: 2, objetos: $\{2\}$ $\triangleright$ objeto 2 vale mais que objeto 3

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0	0	400	400	500	500
3	0	300	400	700		
4	0					

capacidade: 3, objetos: $\{2, 3\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0	0	400	400	500	500
3	0	300	400	700	700	
4	0					

capacidade: 4, objetos: $\{2, 3\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0	0	400	400	500	500
3	0	300	400	700	700	800
4	0					

capacidade: 5, objetos: $\{1, 3\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

	0	1	2	3	4	5 Y
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0	0	400	400	500	500
3	0	300	400	700	700	800
4	0					
i						

capacidade: 0, objetos: $\{\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0	0	400	400	500	500
3	0	300	400	700	700	800
4	0	300				

capacidade: 1, objetos: $\{3\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0	0	400	400	500	500
3	0	300	400	700	700	800
4	0	300	400			

capacidade: 2, objetos: $\{2\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0	0	400	400	500	500
3	0	300	400	700	700	800
4	0	300	400	700		

capacidade: 3, objetos: $\{2, 3\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0	0	400	400	500	500
3	0	300	400	700	700	800
4	0	300	400	700	750	

capacidade: 4, objetos: $\{3, 4\}$

Problema da mochila booleana (PD)

Exemplo: $n = 4$, $W = 5$

	1	2	3	4
w	4	2	1	3
v	500	400	300	450

i \ Y	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	0	0	0	500	500
2	0	0	400	400	500	500
3	0	300	400	700	700	800
4	0	300	400	700	750	850

capacidade: 5, objetos: $\{2, 3\}$

Obtenção da mochila booleana (PD)

mochila(w, v, n, W)

$Y = W$
para $i = n$ **até** 1 **faça**
  se $(t[i, Y] = t[i-1, Y])$ **então** $m[i] = 0$
  senão $m[i] = 1$
  $Y = Y - w[i]$
retorne m

Complexidade: $O(n)$

Obrigado