

**Universidade Federal do Rio Grande do Sul -
UFRGS**

**Instituto de Informática - INF
Ciência da Computação**

Problema da Mochila

Daniel Beckert, Diogo Reali

Introdução

- É um dos 21 problemas NP-completos de Richard Karp, exposto em 1972.

- *“Dado um conjunto de itens, cada item contém um valor e um peso, uma mochila que suporta um determinado peso, preencher a mochila de tal forma que o valor total seja o maior possível e o peso não ultrapasse o limite da mochila.”*
- *“Dado um conjunto de itens, cada item contém um valor e um peso, uma mochila que suporta um determinado peso e um valor mínimo, existe um conjunto de itens que possam preencher a mochila, de forma que o peso não ultrapasse o limite da mochila e o valor arrecadado seja pelo menos igual ao valor mínimo?”*

Definição Matemática

- conjuntos de números positivos $\{p_1, p_2, \dots, p_n\}$, $\{v_1, v_2, \dots, v_n\}$, W , C e um conjunto X de $\{1, 2, \dots, n\}$,
 - p_i : peso do elemento i
 - V_i : valor do elemento i
 - W : capacidade da mochila
 - C : valor mínimo desejado
 - $p(X)$: soma dos pesos
 - $v(X)$: soma dos valores
- Busca-se: encontrar um subconjunto X de $\{1, 2, \dots, n\}$ que maximize $v(X)$ sob as restrições $p(x) \leq W$ e $v(x) \geq C$.
- Uma mochila viável é qualquer subconjunto X de $\{1, 2, \dots, n\}$ tal que $p(X) \leq W$ e $v(X) \geq C$.

Problema da Mochila é NP

- Prova:
 - elaborar um algoritmo de verificação em tempo polinomial.
 - aplicando ao problema proposto, deseja-se mostrar se uma solução qualquer é viável, ou seja, se a soma dos pesos dos itens respeita os limites da mochila.

Algoritmo

knapsackNP($W, C, X[1, 2, \dots, n], P[p_1, p_2, \dots, p_n],$
 $V[v_1, v_2, \dots, v_n]$)

1. somaPesos = 0

2. somaValor = 0

3. para i de 1 até n

4. somaPesos = somaPesos + $P[i]$

5. somaValor = somaValor + $V[i]$

6. se somaPesos \leq capacidadeMochila &&
 somaValor $\geq C$

7. entao Sim

8. senão Não

$O(n)$

Problema da Mochila é NP-completo

- Redução de uma instância do Subset Sum para uma instância do problema da mochila é bastante simples.

Subset Sum

- O problema da soma de subconjuntos.
- Dado um conjunto $X = \{x_1, x_2, \dots, x_n\}$ de inteiros positivos e um valor T inteiro positivo. É possível encontrar um subconjunto X' tal que a soma de todos os seus elementos seja igual a T ?

Redução

- Colocamos como limite de peso (W) e valor (C) o valor de T , e tomamos como peso e valor de cada item K_i o valor do elemento X_i do conjunto de entrada do subset sum.

Algoritmo

sumsetSum-to-knapsack($X[1,2,\dots,n],T$)

1. $W = T$
2. $C = T$
3. para i de 1 até n
4. $P[i] = X[i]$
5. $V[i] = X[i]$
6. retorna W,C,P,V,X

$O(n)$

Conclusão

- O problema da mochila é NP-Completo