

MO417 – Complexidade de Algoritmos

Ata do exercício 16.2-2

Solução apresentada no dia 30/04/2009

Redator: Renato de Jesus Manzoni

Enunciado: 16.2-2

Forneça uma solução de programação dinâmica para o problema da mochila 0-1 que seja executado no tempo $O(Wn)$, onde n é o número de itens e W é o peso máximo dos itens que o ladrão pode pôr em sua mochila.

Solução:

Para resolvermos o problema da mochila 0-1, é necessário mostrar que este problema tem subestrutura ótima. Considere uma carga que pesa no máximo W quilos. Se removermos o item i dessa carga, a carga restante deve ser a carga mais valiosa que pese no máximo $W - w_i$ que o ladrão pode levar dos $n-1$ itens originais, excluindo i .

Seja $c[i, w]$ o valor da solução para os itens $1, \dots, i$ e peso máximo w . A definição recursiva da solução deste problema para n itens de w peso cada um, é:

$$c[i, w] = \begin{cases} 0 & \text{se } i = 0 \text{ ou } w = 0 \\ c[i - 1, w] & \text{se } w_i > w \\ \max(v_i + c[i - 1, w - w_i], c[i - 1, w]) & \text{se } i > 0 \text{ e } w \geq w_i \end{cases}$$

Na qual c é uma matriz de tamanho $n \times W$, v_i expressa o i -ésimo elemento e w_i o peso do i -ésimo elemento. Note que, no último caso, quando $i > 0$ e w maior ou igual a w_i , a fórmula determina se o i -ésimo elemento deve ser incluído na solução ótima ou não.

Referências

[1] Cormen, T.H.; Leiserson, C.E.; Rivest, R.L.; Stein, C.; Algoritmos: Teoria e Prática. Tradução da 2ª edição americana, 2002