

O problema do Subset-Sum e uma solução em tempo polinomial

Allan Cordeiro Rocha de Araújo.

¹Centro de ciência e tecnologia– Universidade Federal de Roraima (UFRR)
Boa Vista– RR – Brasil

allanps32008@gmail.com

Resumo. Será explicado o problema do subset-sum, assim como mostrando códigos da versão exata e da sua versão aproximada.

1. Introdução

O objetivo neste artigo é, de forma clara, explicar o problema da soma de subconjuntos (subset-sum), assim como mostra os pseudocódigos das versões exata e aproximada.

2. Subset-Sum (Soma de subconjuntos)

O subset-sum é um caso especial do problema da mochila booleana, os dois são NP-Completo.

A definição formal do problema é: Dado um conjunto $\{p_1, \dots, p_n\}$ pertencente aos números naturais e c , decidir se existe um subconjunto X de $\{1, \dots, n\}$ tal que $p(X) = c$. Falando com palavras claras, será visto se, existe alguma soma de valores (no subconjunto) que resulte em c , se existe retorna verdadeiro, se não retorna falso.

3. Pseudocódigo versão exata

Como o problema é NP-Completo, sua complexidade de tempo é $O(2^n)$, sendo n o tamanho do vetor de entrada, no caso o conjunto $\{p_1, \dots, p_n\}$.

Subset-Sum-Rec (p, n, c)

se $n = 0$

então se $c = 0$

então devolva 1

senão devolva 0

senão $s \leftarrow \text{Subset-Sum-Rec}(p, n-1, c)$

se $s = 0$ e $p_n \leq c$

então $s \leftarrow \text{Subset-Sum-Rec}(p, n-1, c-p_n)$

devolva s

4. Pseudocódigo versão aproximada

Como o tempo para versão exata é extremamente grande, existe uma versão aproximada, ela usa programação dinâmica para armazenar o resultado de algumas instancias do problema. A complexidade de tempo dele é $O(nc)$, onde 'n' é o numero de linhas e 'c' é o número de colunas da tabela que será armazenado os resultados das instancias.

Subset-Sum-Prog-Din (p, n, c)

para i crescendo de 0 até n faça

$t[i,0] \leftarrow 1$

para b crescendo de 1 até c faça

$t[0,b] \leftarrow 0$

para i crescendo de 1 até n faça

$s \leftarrow t[i-1,b]$

se $s = 0$ e $p_i \leq b$

então $s \leftarrow t[i-1,b-p_i]$

$t[i,b] \leftarrow s$

devolva $t[n,c]$

Referências

"O problema subset-sum"; IME. Disponível em
<https://www.ime.usp.br/~pf/analise_de_algoritmos/aulas/mochila-subsetsum.html>.
Acesso em 12 de julho de 2018.