

Algoritmos gulosos

Marcos Castro

Algoritmos gulosos

- Algoritmos gulosos são aqueles que, a cada decisão, sempre escolhem a alternativa que **parece** mais promissora naquele momento.
- **Nunca** reconsideram essa decisão, ou seja, uma escolha que foi feita nunca é revista, não há backtracking.
- Exemplo de algoritmo guloso: Dijkstra.
- Fazer a escolha que parece ser a melhor num dado momento é fazer uma decisão localmente ótima.

Algoritmos gulosos

- Geralmente os algoritmos gulosos são utilizados em problemas de otimização.
- Um problema de otimização consiste em encontrar a partir de um conjunto S um subconjunto E de S que deva possuir o menor ou maior custo que satisfazem certa propriedade.

Problema do Troco

- Iremos abordar o problema do troco para exemplificarmos.
- Problema do Troco:
 - Suponha que temos as seguintes moedas disponíveis com valores de 100, 25, 10, 5 e 1.
 - O problema consiste em criar um algoritmo para obter um determinado valor com o menor número de moedas possível.

Problema do Troco

- Algoritmo:

Troco(n)

```
1.  C ← { 100 , 50 , 10 , 5 , 1 }
2.  S ← ∅ //Conjunto de moedas escolhida
3.  soma ← 0 //Soma obtida
4.  i ← 1
5.  enquanto i ≤ 5 e soma ≠ n faça
6.      se s + C[i] ≤ n então
16.         S = S ∪ {i}
7.         soma = soma + C[i]
8.     Senão
9.         i=i+1
```

Bibliografia

- <http://marathoncode.blogspot.com.br/2012/05/algoritmos-gulosos.html>

Contato

mcastrosouza@live.com

www.geeksbr.com

www.twitter.com/mcastrosouza

<https://about.me/mcastrosouza>

<http://youtube.com/c/marcoscassouza>