

Paradigmas de Resolução de Problemas

Programação Dinâmica: Problema do Troco

Prof. Edson Alves - UnB/FGA

2020

1. Definição

Definição

Problema do Troco

Seja $C = \{c_1, c_2, \dots, c_N\}$ uma sequência ordenada de N inteiros positivos distintos e M um inteiro positivo. O problema do troco consiste em determinar um vetor de inteiros não-negativos $x = \{x_1, x_2, \dots, x_N\}$ tal que

$$M = \sum_{i=1}^N x_i c_i$$

e que a soma

$$\sum_{i=1}^N x_i$$

seja mínima.

Características do problema do troco

- Os elementos do conjunto C são denominados *moedas*
- M é o *troco*
- O problema pode ser definido informalmente como: *Qual é o menor número de moedas necessárias para dar o troco M ?*
- Se $c_1 = 1$, há solução para qualquer M
- Se C é o conjunto de moedas utilizadas no sistema financeiro da maioria dos países, o problema do troco pode ser resolvido por meio de um algoritmo guloso

Algoritmo guloso para o problema do troco

- O algoritmo guloso para o problema do troco escolhe, dentre as moedas, a maior delas (c_k) que é menor ou igual a M
- Em seguida, ele atribui a x_k o valor M/c_k e subtrai de M o valor $x_k c_k$
- O algoritmo então prossegue até que M se torne igual a zero
- Para todos os valores x_i não atribuídos durante o algoritmo, vale que $x_i = 0$

Implementação do algoritmo guloso para o problema do troco

```
1 #include <bits/stdc++.h>
2
3 using namespace std;
4
5 vector<int> coin_change(int M, const vector<int>& cs)
6 {
7     int N = (int) cs.size();
8     vector<int> xs(N);
9
10    for (int i = N - 1; i >= 0; --i)
11    {
12        xs[i] = M / cs[i];
13        M -= (xs[i] * cs[i]);
14    }
15
16    return xs;
17 }
18
```

Implementação do algoritmo guloso para o problema do troco

```
19 int main()
20 {
21     vector<int> cs { 1, 2, 5, 10, 25, 50 };
22     int M;
23
24     cin >> M;
25
26     auto xs = coin_change(M, cs);
27
28     for (size_t i = 0; i < cs.size(); ++i)
29         cout << cs[i] << ": " << xs[i] << '\n';
30
31     cout << accumulate(xs.begin(), xs.end(), 0) << " moedas\n";
32
33     return 0;
34 }
```


1. **CORMEN**, Thomas H.; **LEISERSON**, Charles E.; **RIVEST**, Ronald; **STEIN**, Clifford. *Introduction to Algorithms*, 3rd Edition, MIT Press, 2009.
2. **LAARKSONEN**, Antti. *Competitive Programmer's Handbook*, 2017.
3. **HALIM**, Steve; **HALIM**, Felix. *Competitive Programming 3*, Lulu, 2013.