

# MC458 Projeto e Análise de Algoritmos I

Primeiro semestre de 2016

Docente: Orlando Lee

Monitora PED: Amanda Resende

## O problema do troco

Sempre que viaja para outros países, Zoroastro prefere levar moedas e notas do país destino a usar cartões de crédito. Não se sabe exatamente a razão para isso: alguns de seus amigos acham que ele é ligeiramente (ou muito) paranóico e não confia em bancos, outros acham que é apenas uma mania atrelada a seu espírito aventureiro. De qualquer forma, Zoroastro gosta de comprar lembrancinhas em suas viagens, para ele ou para amigos e parentes. É comum ele se ver na seguinte situação ao tentar comprar algum item.

Vendedor: São 14 dólares!

Zoroastro: Aqui está uma nota de 20.

Vendedor: Humm... Lamento, estou sem troco.

Zoroastro: Tudo bem. Aqui tem uma nota de 10 e uma de 5. Pode ficar com o troco!

Em outras palavras, Zoroastro quer comprar um item de um certo valor. No entanto, os tipos de nota que ele tem podem não permitir a ele que pague o valor exato do item. Ele gostaria então de minimizar a quantia que ele pagaria (claro, cobrindo o preço do item). Além disso, ainda pagando a quantia mínima, Zoroastro quer também minimizar o número de moedas e notas que ele usaria para pagar pelo item.

Você deve ajudar Zoroastro projetando um algoritmo de programação dinâmica para resolver seu problema!

## Entrada

A entrada consiste de 3 linhas. A primeira linha contém um inteiro  $V$  ( $1 \leq V \leq 10.000$ ) especificando o valor do item em centavos. A segunda linha contém um inteiro  $n$  ( $1 \leq n \leq 100$ ) especificando o número de moedas/notas que ele tem. A terceira linha contém  $n$  inteiros positivos, cada um representando o valor da moeda ou nota. O valor máximo de uma moeda ou nota não excede o valor máximo de um item, ou seja, o maior valor de uma moeda ou nota é 10.000. Pense que um valor de 200 centavos é uma nota de 2, tá?

## Saída

Imprima a quantia mínima em centavos que deve ser paga e o número mínimo de moedas e notas usadas. Caso Zoroastro não tenha dinheiro suficiente para pagar o item deve-se retornar 0 0.

### Exemplo 1

```
7          /* entrada */
3          /* entrada */
2 5 7      /* entrada */
7 1        /* saída */
```

### Exemplo 2

```
1400
3
500 1000 2000
1500 2
```

### Exemplo 3

```
80
10
1 2 3 4 5 6 7 8 9 10
0 0
```

### Exemplo 4

```
50
10
51 70 3 17 5 10 13 8 44 31
51 1
```

### Dica

Considere o seguinte problema ligeiramente diferente. Suponha que o item tem valor  $U$  e que Zoroastro quer saber se é possível pagar o **valor exato**  $U$  com as moedas que tem e usando o **menor número possível de moedas**. Seja  $r(n, U)$  o menor número das  $n$  moedas que devem ser usadas para pagar exatamente o valor  $U$ . Definimos  $r(n, U) = +\infty$  se não é possível pagar o valor exato. Por exemplo, se  $U = 6$ ,  $n = 4$  e as moedas são 1, 2, 3, 4, há duas formas de pagar:  $6 = 1 + 2 + 3$  ou  $6 = 2 + 4$ . A segunda maneira usa menos moedas; logo,  $r(4, 6) = 2$  nesta instância. Se  $U = 15$ ,  $n = 4$  e as moedas são 1, 3, 5, 8, então  $r(4, 15) = +\infty$ .

Para resolver o problema original de Zoroastro, basta então deve descobrir o menor  $U$  tal que  $U \geq V(\text{valor do item})$  e  $r(n, U) < +\infty$ . Tente descobrir uma recorrência para  $r(n, V)$ : se o valor da  $n$ -ésima moeda for maior que  $U$  então ela não pode ser usada; caso contrário, a solução ótima ou usa a moeda ou não (Uau!). Você então terá dois subproblemas do mesmo tipo.

## Especificações

O programa deve ser implementado em C ou C++. A versão dos processadores usados pelo SuSy é C99 e C++ respectivamente. **Seu programa deve ser compilável nestas versões.**

## Avaliação

Haverá 15 testes abertos e 5 teste fechados. A nota do projeto é proporcional ao número de testes bem-sucedidos. Mais precisamente, se  $N$  é o número de testes bem-sucedidos do seu programa, então sua nota será  $(\frac{N}{20}) \times 10 = \frac{N}{2}$ .

## Prazo de submissão

O programa pode ser submetido até as seguintes datas:

- até 23:59h de 22 de maio (domingo) - sem penalidade,
- até 23:59h de 23 de maio (segunda) - serão descontados  $-4$  pontos da nota, ou seja, a nota será  $(\frac{N}{2}) - 4$ .

## Observações

1. O número máximo de submissões é 10.
2. Para a avaliação será considerada apenas a última versão do programa submetido.
3. Para submissão no Susy utilize o número do seu RA em **Usuário** e a senha utilizada na DAC em **Senha**.
4. Você não pode aproveitar código da Web ou do seu colega. Isto será considerado **plágio (média final zero)**.
5. Você pode usar recursão com memorização se quiser. Entretanto, fazer o projeto usando programação dinâmica o(a) ajudará a entender melhor o assunto. Além disso, na prova vai cair programação dinâmica. . .