

Prova de Programação de Computadores

Prof.: Carlos Camarão

27 de Novembro de 2008

1. Escreva programa para ler seqüência de números inteiros e imprimir sua soma.

O programa deve prever a possibilidade de ocorrência de dados incorretos (que não representam números inteiros). Esses dados incorretos devem simplesmente ser ignorados (não devem alterar a soma).

A existência de dado incorreto deve ser tratada por um *tratador de exceção*, causada por chamada ao método *nextInt* definido na classe *Scanner*. Uma chamada a *nextInt* propaga a exceção *InputMismatchException* caso o valor lido não constitua um valor inteiro válido.

Para testar o fim dos dados de entrada, você pode:

- usar o método *hasNext* da classe *Scanner*, que retorna *false* se e somente se não há mais dados de entrada, ou
- tratar a exceção *NoSuchElementException*, propagada por chamada a *nextInt* quando não há mais dados de entrada.

2. Escreva um programa que leia uma seqüência de caracteres (valor de tipo *String*) *t* e imprima uma seqüência de linhas tal que cada linha contém dois valores, separados por um espaço, sendo:

- (a) o primeiro valor um caractere **presente** em *t*
- (b) o segundo valor o número de ocorrências desse caractere em *t*

Use arranjo *contChar* para armazenar, para cada caractere (valor de tipo **char**), o número de ocorrências desse caractere em *t*.

O tamanho do arranjo *contChar* é dado por *Character.MAX_VALUE*.

3. Uma seqüência de números a_1, \dots, a_n é uma seq-*A* se satisfaz:

- $a_{i+1} < a_i$ para $i = 1, \dots, n-1$
- para $i = 3, \dots, n$, não existe $1 \leq j < i$ tal que a_i é igual à soma de termos a_1 até a_j .

Por exemplo, devemos ter:

$$\begin{aligned} a_3 &\neq (a_1 + a_2) \\ a_4 &\neq (a_1 + a_2 + a_3), \quad a_4 \neq (a_1 + a_2) \\ a_5 &\neq (a_1 + a_2 + a_3 + 4), \quad a_5 \neq (a_1 + a_2 + 3), \quad a_5 \neq (a_1 + a_2) \end{aligned}$$

Escreva um programa que:

- Leia várias linhas, cada linha contendo i) o número n de inteiros componentes de uma seqüência de inteiros, e n inteiros componentes da seqüência, cada inteiro na linha separado do seguinte por um ou mais espaços.

Use a classe *Scanner* e os métodos *hasNext* e *nextInt* definidos nessa classe para, respectivamente, testar o fim dos dados de entrada e obter o próximo inteiro da entrada.

- Imprima, para cada seqüência, o número da seqüência — cada seqüência é identificada por um número, que começa por 1 —, um espaço e em seguida "seq-A" se a seqüência for uma seq-A, ou "Nao seq-A" se a seqüência não for uma seq-A.

Use uma variável *somas* de tipo *ArrayList* para armazenar, para cada número da seqüência, os valores de todas as somas de termos anteriores da seqüência.

Sendo *somas* inicialmente igual a $a_1 + a_2$, para cada a_i , para $i = 3, \dots, n$, adicione uma nova entrada a *somas*, contendo a soma de a_i com o último valor adicionado a *somas* ($a_1 + a_2 + \dots + a_{i-1}$).