

Terceiro Exercício-Programa

Márcio Moretto Ribeiro

Fabio Nakano

Norton Trevisan Roman

26 de Maio de 2013

1 Adedanhas

Um grupo de crianças (ou não tão crianças) se encontra para jogar bola. Acontece que nenhuma delas quer ser goleira. Para decidir quem jogará no gol elas se posicionam em um círculo e uma delas grita “adedaaaaaanhas” (ou “deeeeedos” dependendo de onde são as crianças). Elas começam um processo para decidir em que ordem jogarão no gol.

O processo para decidir a ordem das goleiras é o seguinte: No momento em que a primeira criança grita adedanhas cada criança estende sua mão simultaneamente mostrando uma determinada quantidade de dedos. A criança que gritou soma o número de dedos total. Ela então aponta para si mesma e diz 0, aponta para a criança à sua direita e diz 1, aponta para a próxima criança à direita e diz 2 e assim sucessivamente até que o número atinja o valor total dos dedos. A criança apontada no fim do processo sai da roda. Ela será a primeira a jogar no gol. A criança que estava à sua direita grita novamente “adedaaaaaanhas” e repete o processo até que sobre apenas uma criança.

2 Tarefa

Você receberá três classes devendo implementar o método `sorteia` da classe `ListaCircular`. As classes recebidas são:

- **Pessoa**, contém apenas dois atributos: **String nome** que contém o nome da pessoa e **Pessoa proxima** que contém uma referência para a próxima pessoa na lista.
- **ListaCircular**, responsável por gerenciar uma lista ligada circular. A lista é formada por uma sequência de **Pessoas**. Cada pessoa possui um **nome** e uma referência para o próximo na lista. Como a lista é circular, a última pessoa sempre deve conter uma referência para a primeira da lista.

Essa classe possui um atributo **último**, que deve conter uma referência para a última pessoa na lista.

O método **inicializa** garante que em uma lista vazia o **ultimo** contenha uma referência para **null**.

O método (**insere**) recebe uma string, e deve criar uma nova **Pessoa**, a ser inserida no **fim** da lista circular. Depois da inserção a lista deve continuar sendo circular, ou seja, o último elemento deve apontar para o primeiro.

O método **sorteia** deverá ser implementado por você. Ele deve receber um inteiro n , identificar quem foi sorteado, remover essa pessoa do círculo e retornar seu nome. Além disso, o método deve atualizar o atributo **ultimo**, que deve ser agora uma referência para a pessoa anterior à sorteada (dessa forma a pessoa à frente da sorteada passa a ser a primeira).

- **Adedanhas**, contém o programa em si. O programa inicializa uma lista circular e insere uma série de pessoas na lista. Depois disso é inicializado um array de números sorteados. Por fim, para cada número sorteado é impresso na tela a ordem em que as pessoas devem jogar no gol (da primeira para a última).

Atenção! Modifique apenas os corpos dos métodos **sorteia e **insere****

3 Entradas

A entrada do método **sorteia** é um inteiro que contém o número sorteado. Lembre-se que o atributo **ultimo** é uma referência para a última pessoa da lista.

4 Saída

O método `sorteia` deve retornar o nome da pessoa sorteada, deve alterar a lista de forma a removê-la e deve atualizar a referência para a última pessoa da lista (que agora será a pessoa anterior àquela que foi removida).

5 Material a Ser Entregue

Deverá ser entregue o arquivo `ListaCircular.java` com o método `sorteia` implementado. A entrega deverá ser feita pelo TDIA-AE ou pelo COL (dependendo da opção do professor) até a data e hora marcadas. Deverá ser postado um zip com o arquivo `.java` correspondente ao exercício, tendo seu número USP como como, ou seja:

`numero.usp.zip`

Somente este arquivo deve ser postado sendo a postagem de responsabilidade exclusiva do aluno.

6 Avaliação

Para avaliação serão observados os seguintes quesitos:

1. Documentação: se há comentários explicando o que se faz nos passos mais importantes e para que serve o programa (Tanto o método quanto o programa em que está inserido)
2. Apresentação visual: se o código está legível, edentado etc.
3. Corretude: se o programa funciona.

Além disso, algumas observações pertinentes ao trabalho, que influem na nota:

- Este EP é individual.
- Plágio não será tolerado em nenhum hipótese.
- EPs que não compilarem receberão ZERO.

Atenção! Você deve resolver o problema alterando somente o método `sorteia`. Todos o resto do programa deve continuar como está.