

**Universidade Federal de Campina Grande – UFCG**  
**Centro de Engenharia Elétrica e Informática – CEEI**  
**Departamento de Sistemas e Computação – DSC**

Disciplina: Laboratório de Programação 2

Período: 2011.1

Professores: Livia Campos (turma1)  
Reinaldo Gomes (turma2)  
Nazareno Andrade (turma 3)

## **Desafio 03**

---

A partir desse desafio em grupo, deixaremos esta atividade mais independente do lab.

### **Instruções:**

- Data de entrega: 08/04/2011 (Turmas 1, 2 e 3)
- Crie o projeto desafio03 no eclipse e programe todos os exercícios dentro do pacote lp2.desafio03;
- **Salve o projeto** em um arquivo zip chamado desafio03-<seuNome>.zip (ou .tgz ou .rar) e o envie para [juiz.lp2@gmail.com](mailto:juiz.lp2@gmail.com) Use como *subject* do email desafio03-<seuNome> (turma 1, 2 ou 3). A nomeação pode ser em função do nome de um dos integrantes do grupo. **Nomeações incorretas dos arquivos (e subject do email) a serem enviados serão penalizadas na nota do desafio;**
- Certifique-se de que seus programas não têm erros de compilação;
- Antes da definição de cada classe escreva o seu nome em comentário (*/\*Aluno: <nomes do integrantes do grupo do desafio>\*/*);
- Não copie o programa do seu vizinho. Se tiver dúvida converse com o professor ou com um monitor.

Implemente um programa que simplifica o cálculo de quem deve dinheiro a quem na turma da graduação.

Seu programa deve ter o seguinte funcionamento:

- Inicialmente pergunta quantos alunos serão envolvidos no cálculo
- A seguir, imprime a lista de alunos com seus identificadores, identificando os alunos como a1, a2, a3, ...
- A seguir, pede repetidamente ao usuário por empréstimos ou pagamentos feitos entre alunos. O usuário deve digitar E para empréstimo, P para pagamento e T se não quiser informar novos pagamentos/empréstimos.

- A cada empréstimo ou pagamento informado, o programa deve mostrar ao usuário o saldo de cada aluno com cada outro, para todo par de alunos que tenha saldo diferente de zero.
- Depois que o usuário acabou de informar empréstimos e pagamentos, o sistema faz a simplificação dos saldos entre os usuários e imprime os saldos diferentes de zero.

O objetivo da simplificação de saldos é diminuir a quantidade de pagamentos que precisam ser feitos entre alunos e assim simplificar a vida deles. A nossa funciona da seguinte maneira:

Se o aluno a1 deve 10 ao aluno a2, o aluno a2 deve 10 ao aluno a3 e o aluno a3 deve 10 ao aluno a1, o sistema descobre que ninguém precisa pagar nada a ninguém. (Se a1 pagasse o que deve a a2, a2 pagasse a3 e a3 pagasse a1, o dinheiro voltaria para a mão de a1).

Se a1 deve 10 a a2, a2 deve 5 a a3 e a3 deve 3 a a1, o sistema deve calcular que os novos débitos são:

- a1-> a2: 7
- a2-> a3: 2
- a3 -> a1: 0

Note que:

- Nosso programa considera apenas simplificações com três usuários.
- Há várias simplificações possíveis para um conjunto de usuários. No nosso programa, começamos sempre a simplificação pelo usuário a1 (se for possível), depois a2, etc. Dessa forma, se é possível simplificar o débito considerando  $a1 \rightarrow a2 \rightarrow a4$  e  $a1 \rightarrow a2 \rightarrow a3$ , a segunda será feita antes da primeira.

Dicas, sugestões:

- Você pode talvez ter duas classes: Aluno e Banco
- Como você sabe a quantidade de alunos no começo do programa, você pode fazer com que cada objeto Aluno conheça o saldo do aluno que ele representa para cada outro aluno.
- O Banco tem um array de objetos Aluno e usa ele para simplificar os saldos.
- O seu programa pode ser implementado no método Main de Banco
- A classe Banco pode receber no construtor a quantidade de alunos que ela conterá.
- A classe Banco poderia ter métodos `novoEmprestimo(String credor, String devedor, float valor)` e `novoPagamento(String pagador, String beneficiário, float valor)`.