

Lista02_TADS

Luca Cariolin - (119120572) - Ciência da Computação - UNIBH

Raphael Vieira Alves - (119126889) - Ciência da Computação - UNIBH

Introdução

Desenvolvimento de um algoritmo simulando operações básicas de um sistema bancário, tendo por objetivo rever conceitos básicos de programação e alocação de memória bem como explorar os conceitos de Tipos Abstratos de Dados (TADs).

A descrição passo a passo de como o algoritmo deverá ser executado pode ser encontrado no readme.md dentro da pasta raiz do projeto.

Desenvolvimento

Inicialmente, foi criada a classe TContaBancaria, com o objetivo de conter todos os atributos referentes a conta de um cliente. É através dela, que criou-se uma lista do tipo TContaBancária para armazenar as contas cadastradas no decorrer do uso do programa.

Os atributos da classe TContaBancaria são:

agencia - INT

numero - INT

nome - STRING

CPF - STRING

saldo - DOUBLE

Foi criado os getters e setters desses atributos visando o encapsulamento, e além disso, foi criado um método construtor passando todos esses atributos, com o objetivo de facilitar a criação de uma nova conta.

Também foi feita a sobrescrita do método **toString** da classe para exibir todos os valores dos atributos quando acionado esse método.

Posteriormente a isso, foi iniciado a criação da classe Main, responsável por fazer toda a interação com o usuário, além de todas as operações nas contas cadastradas.

Na classe Main, criamos uma lista do tipo TContaBancaria para adicionar os objetos criados do tipo TContaBancaria. É com essa lista de contas cadastradas, que será realizada a busca por uma conta específica para realizar uma operação selecionada pelo usuário.

Ao iniciar o programa, o sistema apresentará opções desde cadastro de contas, até transferência de valores. Essas opções apresentadas ao usuário estão dentro de um Switch com opções que vão de 1 a 6.

Para cada opção de operação, foi criado um método estático para receber os valores de entrada do usuário. Esses métodos estáticos foram criados com o objetivo de manter um código limpo e reutilizável.

Também foi criado um método estático chamado **buscaConta**, esse método tem por objetivo de retornar uma conta cadastrada através de agência e conta informado pelo usuário. Caso a conta não seja identificada será retornado null e será exibida uma mensagem de conta não encontrada.

Para operações de retirada em conta ou transferência entre contas, a conta que sofrer a retirada de saldo, terá uma validação se o valor a ser retirado é maior que o saldo existente. Caso o saldo seja menor que o valor de retirada, não será realizada a operação.

Conclusão

O trabalho prático realmente cumpriu o propósito de relembrar os diversos conceitos aprendidos durante o curso, entre eles: conceitos de programação orientada a objetos, conceitos de estruturas de dados, conceitos de lista, e estruturas condicionais e de repetição.

Além disso, foi interessante a aplicação desses conceitos, simulando um sistema real, que poderia ser usado no dia a dia de uma corporação.