

## **Introdução**

Foi desenvolvido no algoritmo a possibilidade de criação de conta bancária e suas devidas funções, utilizando todos seus devidos dados, sendo eles número de conta, agência, nome, cpf e até saldo possibilitando a simulação completa do banco, também sendo possível a execução de depósito e extração em conta junto com transferências bancárias de valores entre contas utilizando os métodos de alocação de memória e TADs.

## **Desenvolvimento**

Implementamos no algoritmo duas classes sendo elas, classe principal (Main) e classe secundária (TContaBancaria) com dados abstratos (TADs).

Passamos os dados de cada conta tendo setters e getters para atribuir valores e buscar valores em cada atributo, tais como: número, agência, nome, CPF e saldo.

Na classe Main, implementamos toda a funcionalidade do código. Iniciamos variáveis que utilizamos durante o código, por exemplo: tipos de saldo, tipos TContaBancaria, Arraylist.

Para funcionalidade do algoritmo iniciamos com um loop (while) que continuaria executando enquanto não fosse selecionado a função sair. Para cada opção disponível para o usuário utilizamos um número que representa cada uma delas e de acordo com o número selecionado executamos um switch/case que executa uma funcionalidade diferente, sendo elas:

- 1 - Iniciar uma nova conta;
- 2 - Depositar valores;
- 3 - Saque;
- 4 - Exibir seu saldo;
- 5 - Transferência;
- 6 - Sair;

No índice 1 incrementamos um objeto do tipo TContaBancaria e o salvamos em um arraylist. A partir dos outros índices (2 a 5) utilizamos loops para acessar, buscar e “setar” dados de acordo com a necessidade do índice

## **Conclusão**

A atividade fez jus à sua proposta de relembrar conceitos orientados a objeto e dinâmica de TADs.