

**Fundação CECIERJ - Consórcio CEDERJ**  
**Curso de Tecnologia em Sistemas de Computação**  
**Disciplina: Programação com Interfaces Gráficas (EAD05030)**  
**Professores: Dianne Medeiros e Luís Henrique Costa**  
**AD1 1º semestre de 2025**

## 1 Objetivo

Implementar um simulador de sistema bancário com contas e transações utilizando orientação a objetos e Python.

### 1.1 Descrição do Sistema

O sistema bancário deve simular operações entre contas existentes no sistema, exibir informações financeiras das contas e permitir o gerenciamento da conta. O sistema também deve suportar contas de tipos distintos que funcionarão com base em regras específicas para o tipo de conta. Um cliente pode ter mais de uma conta.

As operações suportadas são:

- Depósito;
- Saque;
- Transferência.

As informações financeiras das contas que devem ser exibidas são:

- Operações realizadas (extrato);
- Limite de cheque especial, quando for o caso;
- Saldo atual.

Os tipos de conta possíveis são:

- Conta corrente;
- Conta poupança.

As contas devem ter associadas a elas as seguintes informações:

- Número da agência;
- Número da conta;
- Tipo de conta;
- Titular da conta;
- Endereço do titular;
- Saldo atual.

## 1.2 Requisitos

- Uma conta corrente deve ter um limite de cheque especial;
- Uma conta corrente deve ter uma taxa de manutenção mensal;
- Uma conta poupança não pode ter saldo negativo;
- Uma conta poupança deve ter um rendimento mensal;
- Todas as operações devem ser registradas com a informação de tipo de operação, valor da operação e data da operação.
  - No caso de transferência, informar a origem e o destino da transferência;
  - Em caso de depósito, informar a origem do depósito.
- Um cliente pode solicitar a exclusão da conta;
  - A exclusão deve ser confirmada por um usuário administrador do sistema;
  - As solicitações pendentes devem aparecer para o administrador do sistema e ele deve ter a opção de aceitar ou recusar.
- Um cliente pode modificar seu próprio endereço;
- Um cliente não pode solicitar a alteração de nenhuma outra informação da conta;
- O cliente pode emitir um extrato;
  - Deve ser possível indicar as datas de início e fim do extrato;
  - Ao limitar com essas datas, apenas as operações realizadas entre as datas devem ser exibidas.
- Tratar erros mostrando mensagens de erro ao cliente e impedindo a realização de solicitações que não podem ser realizadas. Por exemplo, um cliente deseja transferir valores para uma conta que não existe, deve ser mostrada uma mensagem de erro informando que não é possível realizar a operação porque a conta não existe. Outro exemplo, um cliente deseja sacar valores sem ter saldo o suficiente (na conta corrente isso ocorre apenas se ultrapassar o limite do cheque especial), deve ser mostrada uma mensagem de erro informando que não há saldo suficiente;
- A interação entre o cliente e o sistema deve ser feita por meio da CLI.

---

## 2 O que Entregar

Deve ser enviado o código fonte em mídia digital e o programa deverá rodar em ambiente Linux com Python 3 (recomendável que rode em Python 2 também). O código fonte é composto pelo conjunto de códigos utilizados para que o programa execute com sucesso. Por exemplo, se você criou uma biblioteca que é importada para um programa principal, deverá enviar tanto o arquivo contendo o código da biblioteca quanto o arquivo contendo o código principal.

Não fornecemos código “gabarito” nesse curso, visto que não existe gabarito de programa. Cada pessoa implementa a mesma funcionalidade e resolve o mesmo problema de formas distintas. Contudo, o seu código deve atender a todos os requisitos listados na Seção 1.2 e deve estar de acordo com a descrição do sistema na Seção 1.1.