



## Prova 01

Nome: \_\_\_\_\_ Matrícula: \_\_\_\_\_

Data: \_\_\_\_\_ 07/10/2017 \_\_\_\_\_ Valor: \_\_\_\_\_ 30 pontos \_\_\_\_\_ Nota: \_\_\_\_\_

Em linguagem de programação Java e usando o **IDE Eclipse**, escreva o algoritmo para gerência das contas de uma agência bancária. A agência bancária disponibiliza duas modalidades de conta: corrente e de poupança. O algoritmo deve:

### Conta (5 pontos)

- 1) Possuir um pacote **conta**. O pacote conta deve possuir: **(1 ponto)**
  - a. A interface **Conta**, com as assinaturas de método: **(1 ponto)**
    - i. `public void setNumeroConta (String numeroConta).`
    - ii. `public String getNumeroConta ().`
    - iii. `public void setDeposito (double valor).`
    - iv. `public void setSaque (double valor).`
    - v. `public void setSaldo ().`
    - vi. `public double getSaldo ().`
  - b. A classe abstrata **ContaAbstrata**, que implementa a interface Conta e possui os atributos privados: **(1 ponto)**
    - i. Número da conta.
    - ii. Saldo.
  - c. A classe concreta **ContaCorrente**, que estende a classe ContaAbstrata e: **(1 ponto)**
    - i. Possui o atributo privado "taxaServico", para cobrança da taxa de serviço do banco.
    - ii. Sobrescreve o método que retorna o saldo, descontando a taxa de serviço do banco.
  - d. A classe concreta **ContaPoupanca**, que estende a classe ContaAbstrata e: **(1 ponto)**
    - i. Possui o atributo privado "taxaRendimento", para conceder o rendimento sobre o saldo depositado.
    - ii. Sobrescreve o método que retorna o saldo, concedendo o rendimento sobre o saldo depositado.

### Lista (10 pontos)

- 2) Possuir um pacote **agencia**. O pacote agencia deve possuir: **(1 ponto)**
  - a. A classe **Lista** com os atributos e métodos necessários para: **(1 pontos)**
    - i. Inserção na lista. **(2 pontos)**
    - ii. Pesquisa recursiva na lista. **(2 pontos)**
    - iii. Impressão do conteúdo da lista. **(2 pontos).**
    - iv. Verificação de lista vazia. **(2 pontos)**

A classe Lista deve possuir a estrutura de célula:

```
private static class Celula {  
    Object item;  
    Celula prox;  
}
```

## Principal (15 pontos)

- 3) Possuir um pacote **inicio**. O pacote inicio deve possuir: **(1 ponto)**
- A classe **Principal** com a definição e chamada de métodos que:
    - Exemplifiquem a inserção de contas correntes e contas de poupança na lista de contas da agência bancária. **(2 pontos)**
      - Os dados de entrada das contas correntes e das contas de poupança devem ser gerados aleatoriamente por métodos apropriados. **(2 pontos)**
    - Exemplifiquem a pesquisa de contas correntes e contas de poupança na lista de contas da agência bancária. **(2 pontos)**
      - Os dados de pesquisa das contas correntes e das contas de poupança devem ser gerados aleatoriamente por métodos apropriados. **(2 pontos)**
    - Imprima o conteúdo da lista de contas da agência bancária. **(2 pontos)**
    - Calcule e apresente o total em dinheiro depositado nas contas da agência bancária. **(2 pontos)**
    - Grave em arquivo o conteúdo da lista de contas da agência bancária. **(2 pontos)**

Formulário para consulta:

```
try {
    ...
} catch (Exception e) {
    System.out.println ("ERRO ao gravar o aluno [" + aluno.getId() + "] " +
        aluno.getNome() + "no disco rígido!");
    e.printStackTrace ();
} finally {
    try {
        if (buffer_saida != null)
            buffer_saida.close ();
        if (gravador != null)
            gravador.close ();
        if (saida != null)
            saida.close ();
    } catch (Exception e) {
        System.out.println ("ERRO ao fechar os manipuladores de escrita do
            arquivo aluno.txt");
        e.printStackTrace ();
    }
}
}
buffer_saida.write (aluno.getId () + separadorDeLinha);
FileOutputStream saida = null;
BufferedWriter buffer_saida = null;
File arq = null;
buffer_saida.flush();
OutputStreamWriter gravador = null;
saida = new FileOutputStream (arq, true);
arq = new File ("aluno.txt");
buffer_saida = new BufferedWriter (gravador);
gravador = new OutputStreamWriter (saida);
String separadorDeLinha = System.getProperty ("line.separator");
```