

Universidade Federal de Uberlândia
Faculdade de Computação

Disciplina: Programação Orientada a Objetos II – BSI – AARE – 2020-2

Professor: Fabiano Azevedo Dorça

Data limite para entrega: 09/10/2020

Valor: 20,0 pontos

[Apresente diagrama de classes e código fonte da solução para as seguintes questões.]

[Atividade avaliativa INDIVIDUAL] [Cópias serão anuladas]

Questão 01 [10 pontos]) Em computação e linguística, análise sintática (também conhecida pelo termo em inglês parsing) é o processo de analisar uma sequência de entrada (lida de um arquivo de computador ou do teclado, por exemplo) para determinar sua estrutura gramatical segundo uma determinada gramática formal.

A análise sintática transforma um texto na entrada em uma estrutura de dados, em geral uma árvore, o que é conveniente para processamento posterior e captura a hierarquia implícita desta entrada.

Utilizando o padrão composite, represente a estrutura de uma linguagem de programação em que um programa pode ser formado por um ou mais comandos. Para isto, deve ter os seguintes nós:

- comando composto: bloco de N comandos;
- comando simples: nó folha, deve armazenar uma string com o nome do comando;
- comando condicional: deve ser composto por dois nós representando cláusulas if e else, que por sua vez podem armazenar um comando simples ou um comando composto.

O método execute() deve ser usado para executar o programa.

O método execute() de um comando simples deve mostrar o nome do comando, fornecido em seu construtor.

O método execute() de um comando composto deve varrer a lista de comandos do bloco e executar cada um deles.

O método execute() de um comando condicional deve gerar um número aleatório entre 0 e 1. Caso o resultado seja menor que 0,5, o if deve ser executado. Caso contrário, o else deve ser executado.

Questão 02 [10 pontos]) Durante a implementação de um *framework* para controle bancário, verificou-se que os processos de abertura de contas Poupança e contas Corrente em um banco são muito parecidos. Aplique o padrão *template method* para o processo de abertura de contas.

Poupança	Corrente
cadastrarPessoaFísica(); validarPessoaFísicaSPC(); depositarValorInicial(); cadastrarSenha(); finalizar();	cadastrarPessoaFísica(); validarPessoaFísicaSPC(); depositarValorInicial(); configurarLimite(); cadastrarSenha(); SE (cliente quer cartão de crédito) solicitarCartãoCrédito() finalizar();