

Universidade Federal de Roraima
Departamento de Ciência da Computação
Linguagens de Programação
DISCIPLINA: Linguagens de Programação – DCC206
2ª Lista
Prazo de entrega: 11/07/2017
ALUNO(A): Vancarla Soares Aquino

ATENÇÃO: Descrever as soluções com o máximo de detalhes possível, inclusive a forma como os testes foram feitos. Para as questões que requisitem a escrita/implementação de programas, deve ser enviado código fonte do programa. Na resposta para a questão deve ser apresentado: o modo de compilar/executar o programa; a linha de comando para executar o programa; e um exemplo de entrada/saída do programa.

[Questão 01]. Descreva o que é programação funcional, bem como, apresente suas vantagens e desvantagens.

Programação funcional é um paradigma de programação que trata a computação como uma avaliação de funções matemáticas e que evita estados ou dados mutáveis. Ela enfatiza a aplicação de funções, em contraste da programação imperativa, que enfatiza mudanças no estado do programa.

VANTAGENS:

- Um alto nível de abstração, especialmente quando as funções são utilizadas, suprimindo muitos detalhes da programação e minimizando a probabilidade da ocorrência de muitas classes de erros.
- A não dependência das operações de atribuição permite aos programas avaliações nas mais diferentes ordens. Esta característica de avaliação independente da ordem torna as linguagens funcionais as mais indicadas para a programação de computadores maciçamente paralelos.
- A ausência de operações de atribuição torna os programas funcionais muito mais simples para provas e análises matemáticas do que os programas procedurais.

DESVANTAGENS:

- Menor eficiência.
- Problemas que envolvam muitas variáveis (ex. contas de banco) ou muitas atividades sequenciais são muitas vezes mais fáceis de se trabalhar com programas procedurais ou programas orientados a objeto.

[Questão 03]. Defina Orientação a Objetos (OO) e apresente benefícios.

É um modelo de análise, projeto e programação de sistemas de software baseado na composição e interação entre diversas unidades de software chamadas de objetos.

BENEFÍCIOS:

- A codificação fica mais próxima do cenário real do problema a ser resolvido
- A manutenção futura fica mais simples e rápida
- Maior reutilização de código
- Padronização do sistema
- Segurança