

Universidade Federal de Roraima
Departamento de Ciência da Computação Linguagens de Programação
DISCIPLINA: Linguagens de Programação – DCC206
2ª Lista

Aluno(a): Débora Bianca Taveira de Moura

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

Programação funcional é um paradigma onde os programas são executados por meio de funções, sem haver mudança de estado. Dessa forma, não existem sentenças de atribuição e nem variáveis no sentido imperativo, e como iterações são regidas pela mutabilidade das variáveis elas não são realizadas nesse paradigma.

Vantagens: Tendo em vista que o código é mais fácil de ser refeito, o modelo pode receber mudanças e manutenção com facilidade. Isso não ocorre na programação imperativa, em que as transformações manuais podem gerar repetições ou incongruências. Outro benefício é a facilidade nos testes e na busca por bugs. Como as funções são mais fáceis de serem avaliadas de forma isolada, os testes podem ser direcionados para expressões específicas, possibilitando programas mais concisos e imunes a erros.

Desvantagens: Apesar desses benefícios, a programação funcional também tem desvantagens. O esforço inicial para o desenvolvimento do programa é uma delas, tendo em vista que pode ser maior que nas linguagens tradicionais. Além disso, na programação funcional é mais difícil estimar o tempo e os recursos necessários para o trabalho de desenvolvimento, tornando o trabalho de gestão do projeto um pouco mais complexo.

[Questão 02] Pesquise e apresente o nome de 03 empresas que utilizam linguagens funcionais, bem como, o nome da linguagem e em qual domínio a linguagem de programação é utilizada.

Linux, Haskell;
Oracle, R;
RaspberryPi, Wolfram;

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

Orientação de objetos é um paradigma de análise, projeto e programação de sistemas de software baseado na composição e interação entre diversas unidades de software chamadas de objetos. Ou seja, é um modelo utilizado no desenvolvimento de software onde trabalhamos com unidades chamadas objetos.

A orientação a objetos visa um pensamento o mais próximo possível da vida real. Ao programar orientado a objetos não teremos apenas funções que são chamadas uma seguida da outra, mas sim objetos que possuem estado e comportamento.

Vantagens: maior facilidade para reutilização de código e por consequência do projeto, possibilidade do desenvolvedor trabalhar em um nível mais elevado de abstração, utilização de um único padrão conceitual durante todo o processo de criação de software, maior adequação à arquitetura cliente/servidor, maior facilidade de comunicação com os usuários e com outros profissionais de informática, ciclo de vida mais longo para os sistemas, desenvolvimento acelerado de sistemas, possibilidade de se construir sistema muito mais complexos, pela incorporação de funções prontas, menor custo para desenvolvimento e manutenção de sistemas

[Questão 04] No programa em java abaixo, apresente e determine: o nível de acesso; o escopo; e o tipo (primitivo ou objeto) de cada variável/atributo.

year: default, Classe Car, int (Primitivo).
make: default, Classe Car, String(Primitivo).
speed: default, Classe Car, double (Primitivo).
num: default, Classe Car, int (Primitivo).
tmp: private, Método getYear, int (Primitivo).
roda: default, Método getYear, int(Primitivo).
r: default, Método getYear, roda(Objeto).

[Questão 05] No programa escrito na linguagem de programação C abaixo, determine se no programa ocorre algum erro (bug) relacionado à referência de memória ou gerenciamento de memória. Caso a resposta seja sim, justifique a sua resposta.

Sim, ocorre um erro relacionado a memória. O problema está na linha 21, pois o código está incrementando o valor de uma constante, e quando a função free for utilizada, ela não desalocará o local correto.