UNIVERSIDADE FEDERAL DE LAVRAS

DEPARTAMENTO DE CIÊNCIAS DA COMPUTAÇÃO

GCC 198 – Paradigmas de Linguagens de Programação

Professor: Erick Galani Maziero

*Lista de Exercícios : Introdução, Histórico e Critérios de Avaliacão das LPs*

1. Em que ano Plankalkül foi projetada? Em que ano foi publicado o projeto?

Foi projetada em 1945, e só veio a ser publicada em 1972

1. Que recursos de hardware apareceram pela primeira vez no computador IBM 704 e afetaram fortemente a evolução das linguagens de programação? Explique por quê.

Indexação e instruções em ponto flutuante direto no hardware

1. Que recursos de SIMULA 67 são agora partes importantes de algumas linguagens orientadas a objetos?

O SIMULA 67 criou o conceito de classe que hoje é amplamente utilizar em POO

1. Para qual área de aplicação JavaScript é mais usada?

Computação para visualização das informações (HTML)

1. Qual é o relacionamento entre JavaScript e PHP, em termos de utilização?

O JavaScript, geralmente embarcado em documentos HTML e interpretado pelo navegador. Já o PHP, é processado ao lado do servidor.

1. Qual a sua opinião sobre o argumento de que as linguagens muito complexas tam-bém são muito perigosas, e que devemos manter todas as linguagens pequenas e simples?

Creio que deve haver um meio termo, onde existam linguagens menores e mais simples, que sejam usadas para iniciação na área de programação. Ao mesmo tempo, linguagens muito complexas com amplas áreas de utilização precisam ser mantidas para tais execuções

1. Construa uma tabela identificando todas as principais evoluções das linguagens, em que conste quando elas ocorreram, em quais linguagens apareceram primeiro e as identidades dos desenvolvedores.
2. Como o conhecimento de linguagens de programação pode beneficiar toda a comunidade da computação?

BLBLBL

1. O que significa para um programa ser confiável?

Significa estar de acordo com suas especificações em todas as condições a qual for submetido

1. O que é o tratamento de exceções?

É a habilidade de um programa interceptar erros em tempo de compilação, trata-los e continuar a execução do programa

1. Por que a legibilidade é importante para a facilidade de escrita?

Os dois critérios estão amplamente vinculados, pois uma vez que o programa tem alta legibilidade, essas caracteristicas também o tornam mais facil de escrever.

1. Qual é o nome da categoria de linguagens de programação cuja estrutura é ditada pela arquitetura de computadores von Neumann?
2. Qual produz uma execução de programas mais rápida: um compilador ou um interpretador puro?

Compilador produz uma execução mais rápida, uma vez que após compilado está totalmente pronto para execução

1. Cite alguns dos recursos de linguagens de programação específicas que você conhe-ce, cujo objetivo seja um mistério para você.
2. Java usa uma chave de fechamento para marcar o término de todas as sentenças compostas. Quais são os argumentos a favor e contra essa decisão de projeto?
3. Muitas linguagens fazem distinção entre letras minúsculas e maiúsculas em nomes de- finidos pelo usuário. Quais são as vantagens e desvantagens dessa decisão de projeto?
4. Descreva alguns trade-offs de projeto entre a eficiência e a segurança em alguma linguagem que você conheça.
5. Descreva as vantagens e desvantagens de alguns ambientes de programação que você já tenha usado.