

Paradigmas de Linguagens de Programação

Exame Escrito

Centro de Informática - UFPE
17 de agosto de 2004

Questão 1 [1,5] Defina uma função *prefixo* que recebe como argumento um natural n e uma lista xs e retorna uma lista com os primeiros n elementos de xs . Se n for maior que o número de elementos de xs , a própria lista xs é retornada. Defina também uma função $\#$ que retorna o número de elementos de uma lista e prove a seguinte propriedade:

$$\# (\text{prefixo } n \text{ } xs) = n, \quad \text{assumindo } \# xs \geq n$$

Questão 2 [1,5] Qual a motivação para integrar paradigmas de programação e que cuidados se deve ter no projeto de linguagens multi-paradigmas?

Questão 3 [1,5] Explique a diferença entre Tipos Abstratos de Dados e Classes. Qual dos conceitos surgiu primeiro e influenciou o outro? Relacione estes conceitos com os paradigmas de programação, indicando em quais paradigmas cada um dos conceitos pode ser naturalmente implementado.

Questão 4 [1,5] Explique o que é o princípio da completude dos tipos (*type completeness principle*) e, em particular, o que o princípio sugere com relação a variáveis transientes e persistentes.

Questão 5 [4,0] Estenda a linguagem LI1 com o seguinte comando de repetição

```
do
  e1 -> c1
  e2 -> c2
  ...
  en -> cn
od
```

onde cada ei é uma expressão booleana (condição) e ci um comando. A execução do comando *do ... od* escolhe aleatoriamente uma das condições verdadeiras, executa o comando correspondente e reinicia a execução; quando todas as condições forem falsas, o comando termina. A implementação do comando *do ... od* deve considerar os métodos de avaliação e checka tipo, bem como todas as classes auxiliares necessárias à completa implementação do comando.