

MC346 - Paradigmas de Programação

Exame - 11/07/2016

Em cada questão abaixo, você pode definir funções ou predicados auxiliares à vontade, conforme ache conveniente.

Questão 1 (Valor 0,1) Das três linguagens vistas no semestre (Prolog, Haskell e Python), qual é a mais antiga?

Questão 2 (Valor 3,3) Escreva um predicado `reps(+L, -R)` em Prolog que é satisfeito quando a lista `R` contém os elementos repetidos da lista `L`, ou seja, aqueles elementos que aparecem duas ou mais vezes em `L`. A ordem em que os elementos aparecem em `R` não importa, mas a lista `R` **não** poderá conter elementos repetidos. Exemplos:

```
?- reps([1, 4, 2, 3, 4, 2, 3, 4], R)
R = [4, 2, 3]
?- reps([1, 2, 3, 4, 5], R)
R = []
```

Questão 3 (Valor 3,3) Escreva uma função `rotate(k, l)` em Python que recebe um inteiro `k` e uma lista `l` e retorna uma lista do mesmo tamanho da lista dada mas com os elementos deslocados em `k` posições para a esquerda em modo rotativo, ou seja, supondo que à esquerda da primeira posição está a última posição. Exemplo:

```
>>> rotate(3, [1, 2, 3, 4, 5])
[4, 5, 1, 2, 3]
```

Sua função será também avaliada pela performance. Portanto, tente fazer uma função eficiente, isto é, com tempo de processamento proporcional ao tamanho da lista, mesmo que o parâmetro `k` seja bem maior do que este tamanho. Uma boa dica é usar o operador `%`.

