

# Concurso Nacional de Programação Lógica e Funcional CNPLf'2006

Faculdade de Engenharia da Universidade do Porto

5-7 de Maio de 2006

## SISTEMAS L

### Introdução

Um sistema L é um autómato desenhado por Aristid Lindenmayer em 1968 para descrever o desenvolvimento de células. As células são representadas por *símbolos* e a divisão celular ocorre substituindo símbolos por *sequências de símbolos*.

Eis um pequeno exemplo com dois tipos de células representadas pelas letras A e B:

1. a célula A transforma-se na sequência AB, ou em notação,  $A \rightarrow AB$
2. a célula B transforma-se na sequência BA, i.e.,  $B \rightarrow BA$

Cada regra de divisão descreve como é a transformação de um símbolo em particular. No início, o organismo é representado por uma sequência de símbolos, designado *axioma*. Seja, neste exemplo, por A.

Depois de uma subdivisão o organismo cresce, sendo representado pela cadeia AB (uso da regra 1). Depois de duas subdivisões, cada um dos símbolos em AB é dividido de acordo com as regras, ficando o organismo com quatro células, ABBA. Depois de três subdivisões, fica o organismo representado pela sequência ABBABAAB.

A *ordem* de um sistema-L é o número de vezes que a divisão celular ocorre.

### Tarefa

A sua tarefa consiste em escrever um programa que seja capaz, dado uma lista de regras de divisão  $L$ , um axioma  $X$  e um número natural  $N$  que represente a ordem, calcule o organismo final após  $N$  divisões usando as regras descritas em  $L$ .

Assuma que não há duas regras diferentes para o mesmo símbolo. Se, porventura, um símbolo  $C$  não for associado a uma regra é assumida a regra  $C \rightarrow C$ . Considere, para efeitos de programação, que um símbolo é representado por um carácter.

Se optar pela Programação Lógica, implemente o programa através de um predicado **lSystem/4** em que os argumentos são, respectivamente, a lista de regras *L*, o axioma *X*, a ordem *N* e o organismo final.

Se escolher a Programação Funcional, desenvolva uma função **lSystem** que recebe os três primeiros argumentos e devolve o organismo final.

## Os Dados

Os dados do problema, passados como argumentos ao programa a desenvolver, são como dito acima, a lista de regras de divisão (cada regra é definida por um símbolo e a correspondente sequência de símbolos), uma sequência inicial *X* e um número natural que represente a ordem.

## Os Resultados

O resultado de invocar o predicado ou a função é o organismo na ordem pedida, baseado no axioma e no sistema *L* descrito pelas regras.

### 1º Exemplo (Prog. Lógica)

Segue-se um exemplo ilustrativo do programa pretendido:

```
?- lSystem (["A","AB"],["B","BA"],"A",3,R).
```

```
R = "ABBABAAB"
```

### 2º Exemplo (Prog. Funcional)

Segue-se um outro exemplo ilustrativo do programa pretendido:

```
> lSystem [('A',"AB'),('B',"BA"]) "A" 3
```

```
"ABBABAAB"
```