

# Ficha Prática 3

José Creissac Campos, António Nestor Ribeiro  
{jose.campos, anr}@di.uminho.pt  
PPIII (LESI)

2002/03

## Objectivos

1. Praticar a escrita de predicados envolvendo expressões aritméticas.
2. Praticar a escrita de predicados sobre listas.

## Exercícios

### Expressões Aritméticas

1. Para cada uma das seguintes funções matemáticas, defina um predicado Prolog equivalente:

(a) factorial:

$$fact(n) = \begin{cases} 1 & \Leftarrow n = 0 \\ fact(n-1) * n & \Leftarrow n > 0 \end{cases}$$

(b) multiplicação de números naturais por somas sucessivas:

$$mult(x, y) = \begin{cases} 0 & \Leftarrow y = 0 \\ x + mult(x, y-1) & \Leftarrow y > 0 \end{cases}$$

(c) divisão inteira de números naturais por subtracções sucessivas:

$$div(x, y) = \begin{cases} 0 & \Leftarrow x < y \\ 1 + div(x-y, y) & \Leftarrow x \geq y \end{cases}$$

(d) resto da divisão inteira de números naturais por subtrações sucessivas:

$$\text{mod}(x, y) = \begin{cases} x & \Leftarrow x < y \\ \text{mod}(x - y, y) & \Leftarrow x \geq y \end{cases}$$

(e) função de Fibonacci:

$$F(N) = \begin{cases} F(N - 1) + F(N - 2) & \Leftarrow N > 2 \\ 1 & \Leftarrow N = 2 \\ 1 & \Leftarrow N = 1 \end{cases}$$

## Listas

Escreva os seguintes predicados sobre listas (note que para muitos deles já existe equivalente em Prolog — no entanto, vale a pena o exercício):

1. `pertence/2` — “?- `pertence(E,L)`.” sucede se E pertence à lista L;
2. `tamanho/2` — “?- `tamanho(L,N)`.” sucede se N é o tamanho de L;
3. `conc/3` — “?- `conc(L1,L2,L)`.” sucede se L é a concatenação de L1 e L2;
4. `ultimo/2` — “?- `ultimo(E,L)`.” sucede se E é o último elemento de L;
5. `adjacente/3` — “?- `adjacente(E1,E2,L)`.” sucede se E1 e E2 são elementos adjacentes em L;
6. `inverte/2` — “?- `inverte(L1,L2)`.” sucede se a lista L2 é a lista L1 invertida;
7. `nesimo/3` — “?- `nesimo(L, N, X)`.” sucede se X é o elemento que se encontra na posição N da lista L;
8. `conta/3` — “?- `conta(E,L,N)`.” sucede se E existe N vezes em L;
9. `apaga_primeiro/3` — “?- `apaga_primeiro(L1,X,L2)`.” sucede se L2 é resultado de retirar a L1 a primeira ocorrência de X;
10. `apaga/3` — “?- `apaga(E,L1,L2)`.” sucede se L2 é a lista L1 com os elementos iguais a E removidos.