FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

EXERCÍCIOS DE PROGRAMAÇÃO EM LÓGICA

LUÍS PAULO REIS
DANIEL CASTRO SILVA

MESTRADO INTEGRADO EM ENGENHARIA INFORMÁTICA E COMPUTAÇÃO

PROGRAMAÇÃO EM LÓGICA - 3º ANO

SETEMBRO DE 2007

Exercício MP1. Utilização do Operador =..

É frequente desejarmos realizar uma dada transformação em todos os elementos de uma lista. Para o efeito vamos recorrer a um predicado de aridade 2. A esta transformação chama-se também mapeamento duma lista. Construa um predicado de mapeamento utilizando o operador =.. na sua definição.

Exemplo1:

Tendo

```
f(X,Y):-Y is X*X.
```

vem

```
?-map([2,4,8],f,L).
L=[4,16,64]
```

Exemplo2:

Tendo

```
duplica(X,Y) :- Y is 2*X.
```

vem

```
?-map([1,2,3],duplica,L).
L=[2,4,6]
```

Solução:

```
map([],_,[]).
map([C|R],Transfor,[TC|CR]):-
      aplica(Transfor, [C,TC]),
      map(R,Transfor,CR).

aplica(P,LArgs) :- G =.. [P|LArgs], G.
```

Exercício MP2. Lista de Elementos que tornam Predicado Verdadeiro

Implemente o predicado separa(+L,+Pred,-Lista) que dada uma lista L e um nome de um predicado de aridade 1, devolve a lista com exactamente os mesmos elementos mas em que primeiro aparecem todos aqueles que tornam verdadeiro o predicado.

```
separa(L,P,Res) :- sepDL(L,P,Res-Nots,Nots-[]).

sepDL([],_,P-P,N-N).
```

```
sepDL([V|L],P,[V|Y]-DY,N) :- aplica(P,[V]), !, sepDL(L,P,Y-DY,N).
sepDL([V|L],P,Y,[V|N]-DN) :- sepDL(L,P,Y,N-DN).
```

Exercício MP3. Idades Mais Próximas

Implemente utilizando o setof/3, o predicado mais_proximos(+Idade,-ListaProximos) que, assumindo a existência de factos idade(Nome,Idade) para representar que um dado indivíduo chmado Nome tem idade Idade, devolve em ListaProximos o nome dos indivíduos cuja idade é mais próxima de Idade.

Solução:

```
mais_proximos(I,[N1|Prox]) :-
      setof(Dif-Nome,prox(I,Dif,Nome),[D1-N1|L]),
      primeiros(D1,L,Prox).

prox(I,Dif,Nome) :- idade(Nome,Id), dif(I,Id,Dif).

dif(A,B,D) :- A > B, !, D is A - B.
dif(A,B,D) :- D is B - A.

primeiros(_,[],[]).
primeiros(D1,[D-_|_],[]) :- D > D1, !.
primeiros(D1,[_-N|L],[N|NL]) :- primeiros(D1,L,NL).

%Dados para teste:
idade(maria,30).
idade(pedro,25).
idade(jose,25).
idade(rita,18).
```

Exercício MP4. Definição de functor(Term,F,N) e arg(N,Term,Arg) em termos do operador =..

a) Defina o predicado functor2(Term,F,Arity) que é verdadeiro se Term é um termo cujo functor principal tem o nome F e a aridade Arity.

Solução:

```
functor_(Term,F,N):- Term=..[F|Args], length(Args,N).
```

b) Defina o predicado arg(N,Term,Arg) que é verdadeiro se Arg é o N-ésimo argumento do termo Term.

Solução:

```
arg_(N,Term,Arg):- Term=..[F|Args], position(N,Args,Arg).

position(1,[X|_],X).
position(N,[_|Xs],Y):-N>1, N1 is N-1, position(N1,Xs,Y).
```