

# Ficha Prática 12

## 12.1 Objectivos

1. Criação de programas interactivos.

## 12.2 Conceitos

### 12.2.1 Interactividade em programas Prolog

A forma de criação de um predicado que implemente a interacção com o utilizador, segue um padrão reutilizável, e recorre ao predicado `repeat` já anteriormente utilizado.

```
menu :-
    carrega('ficheiros.dat'),
    repeat,
    write('*** MENU ***'),nl,
    write('1 - Opção 1'),nl,
    write('2 - Opção 2'),nl,
    ...
    write('6 - Opção 6'),nl,
    write('7 - Sair'),nl,
    write('Opção:'),nl,
    read(X),
    processa(X),
    X=7.                % se falhar volta ao repeat

processa(1) :-
    .....
    !.

...
...
...

processa(7) :- !.
processa(_) :- write('Opção Inválida! Tente Novamente!').
```

Se quisermos efectuar validações ao tipo de entrada que queremos obter, pode ser utilizado o predicado `repeat` por forma a forçar leitura de valores que estejam dentro de uma determinada gama.

Para criar um predicado que valide a entrada de números inteiros, podemos escrever o código que se apresenta:

```
lerInteiro(I) :-
    repeat,
        read(I),
        integer(I),
    !.
```

## 12.3 Exercícios

1. Relembre o predicado `aluno/2` apresentado na secção 1.4.2. Escreva um programa que permita, de forma interactiva, realizar as seguintes operações:
  - inscrever um aluno a uma disciplina;
  - remover a inscrição de um aluno a uma disciplina;
  - saber se um aluno está inscrito a uma disciplina;
  - saber a que disciplinas está um aluno inscrito;
  - saber quais os alunos inscritos a uma disciplina;
  - saber qual a disciplina com mais alunos inscritos;
  - saber qual o aluno inscrito a mais disciplinas.
2. Considere que tem, numa base de conhecimento, predicados `bib/2` com informação relativa a livros e respectivos autores:

```
bib(asterix,[uderzo, gosciny]).
bib(relatorio_minoritario,[philip_k_dick]).
bib(maias,[eca_queiroz]).
bib(farpas,[eca_queiroz,ramalho_ortigao]).
bib(ubik,[philip_k_dick]).
```

- (a) escreva o predicado `autores/1` que constrói uma lista que permita mais facilmente responder à questão “Que livros escreveu determinado autor?”. Para a base de conhecimento anterior a lista resultante deverá ter o seguinte formato:

```
[auth(eca_queiroz,[maias,farpas]),
 auth(gosciny,[asterix]),
 auth(philip_k_dick,[ubik,relatorio_minoritario]),
 auth(ramalho_ortigao,[farpas]),
 auth(uderzo,[asterix])
]
```

- (b) desenvolva uma interface que permita ao utilizador inserir, consultar e remover livros, bem como consultar a lista de livros por autor.
3. Considere agora que tem a informação relativa aos livros numa lista. Para a base de conhecimento anterior essa lista assume a forma que a seguir se exemplifica:

```
[bib(asterix,[uderzo, gosciny]),
 bib(relatorio_minoritario,[philip_k_dick]),
 bib(maias,[eca_queiroz]),
 bib(farpas,[eca_queiroz,ramalho_ortigao]),
 bib(ubik,[philip_dick])
]
```

Utilizando os conhecimentos adquiridos durante o semestre apresente duas implementações diferentes para um predicado que transforma a lista de livros na lista de autores referida no ponto anterior. As implementações devem obedecer aos requisitos que se apresentam de seguida:

- `bibtoauth_meta/2` — deve recorrer aos meta-predicados (ex: `setof`, `bagof`, etc.);
- `bibtoauth_recur/2` — deve recorrer apenas à recursividade sobre listas.