

**UNIDADE CURRICULAR:** Linguagens de Programação

**CÓDIGO:** 21077

**DOCENTE:** Ricardo Batista

**A preencher pelo estudante**

**NOME:** Hernâni Filipe Resendes Coelho

**N.º DE ESTUDANTE:** 1800045

**CURSO:** Engenharia Informática

**DATA DE ENTREGA:** 22/05/2022

**TRABALHO / RESOLUÇÃO:**

Este trabalho foi desenvolvido utilizando a linguagem lógica PROLOG, e tem como objectivo implementar um conjunto de predicados definidos no enunciado.

A organização da estrutura de ficheiros do programa está organizada da seguinte forma, a partir do diretório base:

* efolioB/
  + src/
    - corposCelestes.exe
    - corposCelestes.pl
    - dados.pl
    - regras.pl
  + readme.txt

**Organização do código:**

O código está organizado por módulos, de modo a permitir uma melhor legibilidade. As principais escolhas que tomei para implementação do programa foram:

* Definição do ficheiro dados.pl com os factos da base de conhecimento do programa;
* Definição do ficheiro regras.pl com as regras da base de conhecimento do programa;
* Definição do ficheiro corposCelestes.pl como ficheiro que inicializa o programa e apresenta uma interface menu ao utilizador;

O programa inicia na regra “main” a qual disponibiliza ao utilizador um conjunto de opções que permite indirectamente executar as queries. Em cada uma destas opções existe uma regra “retornamenu” que permite simular um loop do menu até o utilizador decidir sair do programa.

A opção 1, implementa o predicado orbita, o qual recebe dois argumentos que são validados e se conforme, é então verificado se o corpo X orbita em torno do corpo Y, sendo a resposta “yes/no”. A opção 2, implementa o predicado listar\_planeta, o qual recebe um argumento e verifica se é válido, se o for segue então para a criação de uma lista que será composta por todos os planetas que respeitem a condição e por fim, invoca a regra para imprimir os elementos da lista na tela. A opção 3, implementa o predicado listar\_satelites, e de forma semelhante a opção 2, produz uma lista com os satélites que respeitem a condição. Posteriormente imprime os elementos da lista. A opção 4, implementa o predicado listar\_planetas\_com\_satelites, produz uma lista com todos os planetas que têm satélites e imprime. A opção 5, implementa o predicado tem\_satelite, recebe um argumento, neste caso deverá ser um planeta, valida e verifica a condição. A opção 6, implementa o predicado maior\_planeta, produz uma lista de diâmetros que respeitem a condição de planeta e posteriormente verifica o maior diâmetro através da regra max. Esta percorre todos os elementos da lista de forma recursiva até atingir o caso base, situação na qual estará perante o maior diâmetro encontrado. A opção 7, implementa o predicado total\_corpos\_celeste, valida o argumento que recebe e procede à contagem dos corpos que respeitem a condição. A opção 8, implementa o predicado media\_diametro\_satelites, após validar o argumento que recebe, produz uma lista com todos os diâmetros dos satélites que orbitam o planeta (argumento). Posteriormente executa uma soma dos valores dessa lista e verifica o comprimento da mesma, para fazer o cálculo da média. A opção 9, implementa o predicado ano\_descoberta\_corpo, se o argumento que recebe for válido procura o facto correspondente para verificar o ano. A opção 10, não a implementei. A opção 11, implementa o predicado alterar\_ano\_descoberta\_corpo, valida os argumentos que recebe e se forem válidos procede removendo o facto que lhe corresponde e volta a inserir novo facto na tabela com a alteração efectuada.

**Exemplo de testes de execução:**

Uma imagem com texto

Descrição gerada automaticamente

Figura 1 Menu inicial com opções dos predicados

Uma imagem com texto

Descrição gerada automaticamente

Figura 2 Opção 1

Uma imagem com texto

Descrição gerada automaticamente

Figura 3 Opção 2

Uma imagem com texto

Descrição gerada automaticamente

Figura 4 Opção 3

Uma imagem com texto

Descrição gerada automaticamente

Figura 5 Opção 4

Uma imagem com texto

Descrição gerada automaticamente

Figura 6 Opção 5

Uma imagem com texto

Descrição gerada automaticamente

Figura 7 Opção 6

Uma imagem com texto

Descrição gerada automaticamente

Figura 8 Opção 7

Uma imagem com texto

Descrição gerada automaticamente

Figura 9 Opção 8

Uma imagem com texto

Descrição gerada automaticamente

Figura 10 Opção 9

Uma imagem com texto

Descrição gerada automaticamente

Figura 11 Opção 11

Uma imagem com texto

Descrição gerada automaticamente

Figura 12 Exemplo de controlo de validação de dados