



Ficha de trabalho nº 2	
Disciplina	Modelação de dados em Engenharia
Ano Lectivo	2020/2021
Objectivo	Modelar e implementar sistema de rotas
Aulas	5 aulas x 3 horas + 6 horas extra
Data de Entrega	2021/05/31

Objectivos concretos:

1. Modelação de factos
 - a. Representar o problema através de uma estrutura de factos adequada.
2. Definir regras de inferência
 - a. Efetuar *queries* sobre factos definidos na base de conhecimento
 - b. Inferir através dos factos e regras definidos.
3. Utilizar ferramentas de modelação para o problema apresentado
 - a. Utilizar o software Swi-prolog
 - b. Permitir a interactividade entre utilizador/base-de-conhecimento.
4. Responder à ficha de trabalho no Moodle.

Objetivos

Este trabalho destina-se a proporcionar aos alunos um contacto com os conceitos de modelação de conhecimento.

Descrição geral do problema

A produção de bens é realizada em ambientes complexos onde várias empresas e fábricas são clientes e fornecedoras umas das outras. Ou seja, quando uma fábrica é responsável por fazer a montagem de automóveis, esta não é habitualmente responsável por produzir o volante, bateria, rodas, etc. Todos esses subprodutos são fabricados por outras fábricas especialistas no seu fabrico, sendo uma delas conforme referido responsável pela montagem do produto final. Assim, quando falamos no fabrico destes bens, temos uma cadeia de valor que é composta por intervenientes que vão “alimentando” os intervenientes seguintes. A otimização desta rede de fábricas é naturalmente difícil devido ao número de possíveis combinações.



Figura 1. Exemplo de uma cadeia de valor

Tal como é apresentado na imagem da figura 1, a cadeia é composta por fábricas que enviam produtos/peças de umas para outras, de forma a que essas peças possam ser utilizadas pelas seguintes para poderem fabricar os seus produtos. O transporte entre as fábricas que constituem a cadeia é feita por transportadores (sejam eles camiões, comboios, barcos, etc). Cada um desses transportadores tem um custo associado, assim como uma pegada ecológica (produção gases estufa e consumo, seja ele sobre a forma de energia elétrica ou combustível).



É então pretendido que desenvolva um sistema que auxilie na procura de fornecedores que possam enviar peças para as fábricas que as requerem, calculando para isso também possíveis rotas para o fazer, utilizando os transportadores inscritos.

Para tal, o sistema possui um motor de inferência cujo papel consiste em determinar os itinerários, de acordo com os critérios colocados pelo utilizador. A solução proposta será desenvolvida através da modelação de uma base de conhecimento, utilizando a linguagem de programação em lógica – *Prolog*.

A base de conhecimento e o respetivo programa a desenvolver deverá disponibilizar uma interface na qual o utilizador poderá visualizar quais as fábricas existentes, os produtos disponibilizados por cada, o *stock* de cada fábrica, assim como os itinerários disponíveis considerando o meio de transporte a utilizar, com respectiva duração, distancia total percorrida, consumos, produção de gases estufa, etc.

Para além do *stock* de cada fábrica, a base de conhecimento também deverá possuir a informação de quais as peças utilizadas para fabricar um determinado produto. Ou seja, pegando novamente no exemplo do automóvel, para cada produto ou subconjunto deste, o sistema deve saber quais as peças que este requer para ser produzido.

Assim, sempre que o utilizador deseje encomendar produtos de outra fábrica, este deve poder inserir no sistema o que deseja e o sistema deve apresentar qual o parceiro que entrega peças mais rapidamente, e o parceiro mais “amigo do ambiente”. Tendo em conta que não podemos otimizar só para um fim ou outro, o utilizador, deverá poder procurar pela rota “mais amiga do ambiente” mas que faz a entrega até uma duração máxima de X tempo.

O processo de implementação deverá ser dividido em duas fases:

Fase1 – Gestão da base de conhecimento

Para a gestão da base de conhecimento, pretende-se a criação de um programa em *Prolog* que permita editar (inserir, remover e alterar) a base de conhecimento, definindo fábricas, transportadores, *stocks*, etc. Assim, o programa a desenvolver, deverá apresentar inicialmente um menu do tipo (estas opções são apenas como ilustração):

MENU 1: Gestão da base de conhecimento

- 1.1 Adicionar fábricas à cadeia/rede.
- 1.2 Adicionar transportadores.
- 1.3 Adicionar *stock* a uma fábrica.
- 1.4 Adicionar descrição de produto (Peças necessárias).
- 1.5 ...
- 1.6 Alterar fábricas da cadeia/rede.
- 1.7 Alterar *stock* numa fábrica.
- 1.8 Alterar transportadores.
- 1.9 Alterar descrição de produto (Peças necessárias).
- 1.10 ...
- 1.11 Remover fábricas da cadeia.
- 1.12 Remover *stock* duma fábrica.
- 1.13 Remover transportadores.
- 1.14 Remover descrição de produto (Peças necessárias).
- 1.15 ...



Fase 2 – Inferência

Numa segunda fase do desenvolvimento, deve ser dada ao utilizador a hipótese de fazer várias consultas/questões à base de conhecimento. Para tal deverá existir um menu do tipo (estas opções são apenas como ilustração):

MENU 2: Consultas à base de conhecimento

- 2.1 – Listar fábricas da cadeia.
- 2.2 – Listar produtos disponíveis na cadeia/rede.
- 2.3 – Listar fábricas por produtos disponibilizados.
- 2.4 – Listar transportes entre fábricas.
- 2.5 – Obter transporte entre fábricas com menor distância.
- 2.5 – Obter transporte entre fábricas com menor consumo.
- 2.5 – Obter transporte entre fábricas com menor produção de gases estufa.
- 2.9 – Obter itinerário entre fábricas com passagem por outras fábricas.
- 2.10 – Obter transporte entre fábricas com menor consumo até uma duração máxima.

Modelação:

O estudo e a implementação do problema proposto neste trabalho abordarão os seguintes conceitos: **átomos**, **predicados**, **factos**, **regras**, **inferência**, **bases de conhecimento**, **programação lógica** e a linguagem **Prolog**. Um aspecto importante consistem em estabelecer a diferença entre sistemas convencionais, baseados em linguagens imperativas, e as abordagens lógicas, mais vocacionadas para a criação de bases-de-conhecimento.

Tabela 1. Requisitos funcionais no desenvolvimento da base de conhecimento

Requisitos Funcionais	Descrição
RF 1	Adicionar fábricas à cadeia/rede e produtos à base de conhecimento.
RF 2	Adicionar transportadores.
RF 3	Alterar fábricas da cadeia/rede, transportadores, etc.
RF 4	Remover ligações entre fábricas da cadeia/rede, transportadores, etc.
RF 5	Listar as peças requeridas para um produto.
RF 6	Listar fábricas com o mesmo tipo de produto.
RF 7	Listar transportes disponíveis entre duas fábricas.
RF 8	Listar transportes entre duas fábricas, considerando a distância, duração, consumo e produção de gases estufa.
RF 9	Listar transportes possíveis entre duas fábricas, passando por uma ou duas outras fábricas (ex: transporte: fábricaX/fábricaY/fábricaZ).
RF 10	Calcular o menor transporte entre duas fábricas (em termos de distância), fornecendo os transportadores envolvidos tal como o total de consumo e gases estufa...



RF 11	Obter transportes de acordo com as preferências do utilizador, ou seja, que possibilitem a passagem por fábricas que possuam determinados productos (ex: transporte: fábricaX/fábricaY passando por fábricas com produtoA).
RF 12	Obter o(s) transporte(s) de acordo com o produto necessário, ou seja, o utilizador escolhe qual o produto que deseja fabricar e o sistema, procura pelo transporte das peças que o constituem.
RF 13	Proponha um requisito relevante baseado na determinação da centralidade de uma fábrica (procurar em bibliografia da Teoria dos Grafos), escolhendo o tipo de medida de centralidade que considerar mais adequado.
RF 14	Proponha um requisito relevante ainda por identificar
RF 15	Interface básica de interação com o utilizador, podendo ser em Prolog ou “ <i>embedded</i> ” numa outra linguagem Java/C++/C#/Visual basic (max. 10% da nota do trabalho)

Implementação:

Os alunos poderão adotar o ambiente de desenvolvimento mais adequado para o desenvolvimento do problema apresentado, mas garantindo a compatibilidade em termos de linguagem Prolog.

Elementos de Reflexão

Esta parte consiste em considerações mais reflexivas que poderiam servir como “inspiração” para a escrita dum relatório (no entanto, não é necessária entrega de relatório para este trabalho). No final do trabalho, as seguintes questões poderiam ser respondidas:

- O que se entende por modelação e noção de modelo.
- Conceito de abstração (ilustrando com pormenores específicos do problema apresentado)
- A utilidade da modelação no contexto da Engenharia.
- O carácter ambíguo, redundante e incompleto da linguagem natural como formalismo de modelação.
- As características que um formalismo de modelação (ou linguagem) deve possuir para que possa representar modelos de forma adequada.
- Representação do conhecimento e o que se entende por Base de Conhecimento.
- Noção de Inferência/raciocínio.
- Linguagens lógicas.

Este trabalho segue o seguinte plano:

1ª aula (26.04 / 30.04.2021):

Introdução à linguagem Prolog.

Instalação do ambiente de desenvolvimento Swi-prolog.

Exemplos com representação de factos e regras, queries, recursividade.

Apresentação do trabalho.



2ª aula (03 / 07.05.2021)

Modelação da base de conhecimento para o trabalho apresentado.

3ª aula (10 / 14 .05.2021):

Desenvolvimento e implementação dos requisitos funcionais definidos no trabalho prático.

4ª aula (17 / 21 .05.2021):

Desenvolvimento e implementação dos requisitos funcionais definidos no trabalho prático.

5ª aula (24 / 28.05.2021):

Desenvolvimento e implementação dos requisitos funcionais definidos no trabalho prático.

Finalização do trabalho.

Entrega (Moodle)

Os alunos devem submeter, através do Moodle, um ficheiro zip/rar que deverá conter o código fonte (incluindo factos/regras que modelam o problema), assim como um vídeo de demonstração do trabalho com cerca de 5 minutos.

Se realizarem o RF15 devem também submeter o respectivo projecto.

Este trabalho requer o preenchimento obrigatório dum questionário no Moodle, de um vídeo, que se destina a substituir a apresentação, e de um ponto de entrega para o projeto completo. Os docentes vão poder seleccionar aleatoriamente (ou sempre que necessário) alguns trabalhos para apresentação dos alunos. Entregas feitas através de email não vão ser consideradas.

Docentes Teórica-Prática:

Luis Camarinha-Matos, cam@uninova.pt

Docentes Prática:

Ana Inês Oliveira, aio@uninova.pt

André Rocha, andre.rocha@uninova.pt

Filipa Ferrada, faf@uninova.pt

João Rosas, jrosas@uninova.pt