

Relatório Trabalho Prático - IA

Grupo 32:

António Luís de Macedo Fernandes (a93312)

José Diogo Martins Vieira (a93251)

João Silva Torres (a93231)

Ricardo Lopes Santos Silva (a93195)

Dec 2, 2021



Contents

1	Introdução	3
2	Base de Conhecimento	4
3	Queries	5
3.1	maisEcologico	5
3.2	byEstafeta	5
3.3	toCliente	5
3.4	faturadoDia	5
3.5	rankClienteEstafetaMedia	5
3.6	zonaMaiorVolume	5
3.7	periodoTempoTransportes	6
3.8	periodoTempoEstafetas	6
3.9	periodoTempoEntregues	6
3.10	pesoPorEstafeta	7
3.11	diasSemana	7
4	Interface	8
5	Conclusão	10

1 Introdução

No âmbito do desenvolvimento do projeto da unidade curricular Inteligência Artificial foi-nos proposto, nesta primeira fase, desenvolver um programa utilizando a linguagem de programação em lógica *PROLOG*. Ora, tivemos de por em prática o que nos foi ensinado durante as aulas, de forma a representar o nosso conhecimento e também construir mecanismos de raciocínio para a resolução de problemas.

Dito isto, representamos um centro de encomendas com entrega em meios ecológicos, a empresa *Green Distribution*. Esta é composta por estafetas que têm associado um conjunto de entregas a efetuar a clientes.

Após este breve resumo, toda a informação relativa a este projeto estará explicada com rigor ao longo deste relatório.

Para elaborar todo este trabalho tivemos de pensa-lo e estrutura-lo. Primeiramente, começamos por definir a nossa base de conhecimento conforme o que nos foi pedido no enunciado. De seguida, passamos ao desenvolvimento das diferentes queries que nos foram fornecidas. Vamos explicar cada uma delas com detalhe mais à frente.

Por fim, na última secção deste relatório iremos falar da interface que criámos e da conclusão.

2 Base de Conhecimento

Como dito anteriormente, começamos então por definir a nossa base de conhecimento, nomeadamente as seguintes regras:

- Morada, que é definida pela rua, número, andar, freguesia e cidade.
- Estafeta, que é definido pelo nome.
- Encomenda, que é definida pelo nome e id ou só pelo id.
- PrazoLimite, que é definido pelas horas.
- Localidade, que é definido pela rua e freguesia.
- MeioTransporte, que é definido pelo veículo, peso máximo e velocidade média.
- Ecologico, que é definido pelo veículo e o seu número ecológico.
- Entrega, que é definida pelo seu idEncomenda, cliente, estafeta responsável pela entrega, meio de transporte utilizado, preço da Entrega, prazo limite, rank que se trata da avaliação da entrega por parte do cliente, data e time.
- PreçoEntrega, que é definido pelo preço, tempo de entrega e meio de transporte utilizado.

Tentámos definir uma base de conhecimento simples e de fácil compreensão com a finalidade de facilitar a resolução das queries.

3 Queries

Nesta secção vamos abordar levemente as estratégias utilizadas nas diferentes queries.

3.1 maisEcologico

Esta query determina o *estafeta* que utilizou mais vezes um meio de transporte mais ecológico para efetuar as *entregas*.

Para tal, obtemos o meio de transporte mais ecológico, e procurámos nas entregas todos os estafetas que o utilizaram, criando uma lista para cada estafeta, o resultado é uma lista de listas. Posteriormente, realizamos uma auxiliar que indica qual das listas tem maior comprimento (dentro da lista de listas) que corresponderá ao estafeta que utilizou mais vezes o meio de transporte mais ecológico.

3.2 byEstafeta

Esta query determina os clientes servidos por um determinado *estafeta*.

Então, para a realização desta query procurámos nas *entregas* pelo nome do *estafeta* pretendido e devolvemos uma lista com os clientes servidos por esse *estafeta*.

3.3 toCliente

Esta query identifica que *estafetas* entregaram certas *encomendas* a um determinado cliente.

Então, procurámos nas entregas o cliente desejado e colocámos numa lista os estafetas que lhe efetuaram entregas.

3.4 faturadoDia

Esta query determina o valor total faturado num determinado dia.

Inicialmente utilizámos a auxiliar *checkdiaprice* que calcula a lista com os preços desse dia e depois faz a soma desses valores, calculando assim o valor total fraturado.

3.5 rankClienteEstafetaMedia

Esta query calcula a classificação média de satisfação de um cliente para um determinado estafeta.

Para tal, calculámos a lista das classificações das entregas feitas pelo estafeta ao cliente, somamos esta lista e dividimos pela *length* da mesma. Obtendo assim a média de rank do estafeta.

3.6 zonaMaiorVolume

Esta query calcula as 3 ruas com maior volume de entregas.

Para isso, começamos por ir buscar todos os números de encomenda das entregas, e a partir destes, ir à encomenda associada e obter a rua, fazendo associar através de um par a rua com o número de encomenda (rua-nrEncomenda). De seguida, fazemos uso do predicado *sort* do Prolog, que organiza a lista de forma a que as ruas fiquem organizadas alfabeticamente, fazendo com que as ruas com o mesmo nome fiquem seguidas.

Isto é necessário para o uso do predicado *group_pairs_by_key*, que vai associar os pares com a mesma key que se encontrem consecutivos.

Assim, a Lista [rua1-1,rua1-2,rua2-4,rua2-5] transformar-se-ia em [rua1-[1,2],rua2-[4,5]].

Com isto, transformamos o valor do par que era uma lista para a sua length, para depois ordenar por ordem decrescente os pares em relação ao valor.

Concluindo com um predicado auxiliar take, que vai retirar, no máximo, os primeiros 3 elementos da lista, que será assim o resultado das 3 zonas com maior volume de entregas.

3.7 periodoTempoTransportes

Esta query identifica o número total de entregas pelos diferentes meios de transporte, num determinado intervalo de tempo.

Para isso, começamos por formar uma lista de Pares, que para todas as entregas associa o Estafeta com a Data (Estafeta-Data).

Depois recorremos ao uso do predicado *group_pairs_by_key*, explicado em pormenor na query anterior, para obter uma lista com os pares (Estafeta-[Data]), que para cada estafeta associa todas as datas que realizou entregas.

Após isto, vamos filtrar as datas pelo intervalo tempo dado pelo utilizador, para tal, fazemos uso do *built-in* predicado do Prolog (date), que permite comparar datas.

Concluindo assim a query com uma lista de pares Estafeta-Nr, sendo Nr o nr de entregas realizadas naquele intervalo de tempo.

3.8 periodoTempoEstafetas

Esta query identifica o número total de entregas pelos estafetas, num determinado intervalo de tempo.

Esta query faz uso do mesmo raciocínio e implementação da query anterior, com apenas uma diferença na parte inicial, para a formação do par. O par desta vez é formado por meioTransporte-Data.

3.9 periodoTempoEntregues

Esta query calcula o número de encomendas entregues e não entregues pela Green Distribution, num determinado período de tempo.

Decidimos considerar como encomenda não entregue, todas as encomendas que tenham sido feitas neste intervalo de tempo e que a entrega não tenha sido efetuada ou a entrega foi efetuada fora deste intervalo de tempo.

Assim, começamos por ir a todas as encomendas e associar num par (Data-NrEncomenda), verificamos se a data da encomenda está entre o intervalo de tempo, caso não esteja é ignorada. Se tiver, verifica-se se já foi entregue e caso tenha sido, se a entrega foi feita dentro do intervalo de tempo, caso passe estas duas condições, vamos incrementar o acumulador das encomendas entregues, caso falhe alguma delas, é incrementado o acumulador das não entregues.

Por fim, é dado o resultado do número de encomendas entregues e não entregues no intervalo de tempo.

3.10 pesoPorEstafeta

Esta query calcula o peso total transportado por estafeta num determinado dia.

Para tal, começamos por formar uma lista de pares (Estafeta-NrEncomenda) com todas as entregues feitas no dia dado.

Usamos o método já explicado na query *ZonaMaiorVolume*, para agrupar os pares pela chave. Após isto, temos uma lista de pares (Estafeta-[NrEncomenda]), usamos uma função auxiliar, que para cada par, vai à lista associar cada nr de Encomenda ao seu peso, somando tudo no final com recurso ao predicado *sumlist*.

Assim é dado o resultado, que fica na forma de uma lista de pares (Estafeta-Peso), sendo Peso, o peso total transportado no dia fornecido.

3.11 diasSemana

Esta query extra determina o número de *entregas* efetuadas em cada dia da semana.

Para tal, começamos por formar uma lista com todas as datas das entregas. Para cada data é criado um par nome-number, o nome corresponde ao nome do dia da semana, o number ao dia da semana (1->segunda,etc).

Depois usamos o método utilizado anteriormente para agrupar os pares com a memsma key. Após isso, tiramos a length de cada lista e, temos assim, o resultado numa lista de pares (diaSemana-numero). O diaSemana é o nome da dia da semana (ex.segunda), e o numero é o número de entregas efetuadas.

4 Interface

Como fomos encorajados a desenvolver mais o nosso trabalho além do proposto no enunciado, desenvolvemos um menu simples com o intuito de facilitar a testagem das queries e a interação do utilizador com a aplicação.

Já que desenvolvemos muitas queries, decidimos dividir o menu em 3 partes de forma a que o nosso menu não ficasse muito extenso. Então, inicialmente, são apresentadas as queries mais genéricas, a opção de sair da aplicação e é dada a possibilidade de escolher os menus das *Queries para Dia* e das *Queries Num Intervalo de Tempo*. Nesses menus mais pequenos, estão inseridas as queries relativas aos dias e a intervalos de tempo.

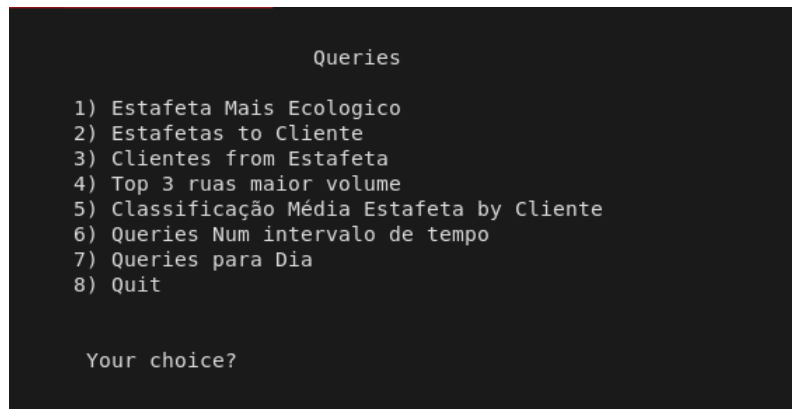


Figure 1: Menu Principal.

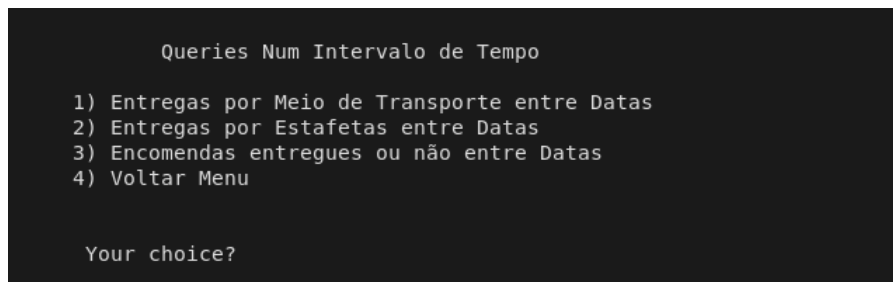


Figure 2: Menu Queries de Intervalo de Tempo.

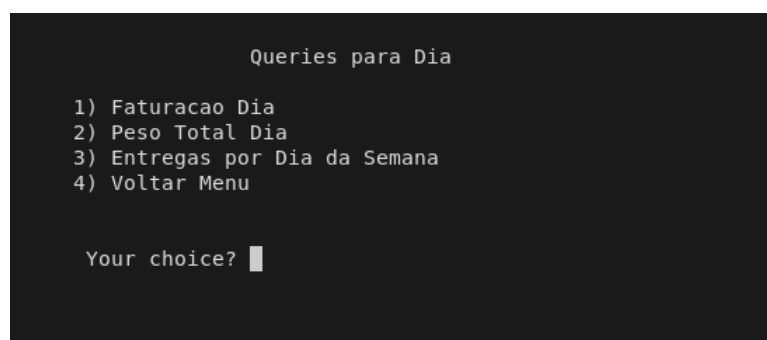


Figure 3: Menu Queries Dia.


```

                                Queries

1) Estafeta Mais Ecologico
2) Estafetas to Cliente
3) Clientes from Estafeta
4) Top 3 ruas maior volume
5) Classificação Média Estafeta by Cliente
6) Queries Num intervalo de tempo
7) Queries para Dia
8) Quit

Your choice? Clientes from Estafeta
Id do Estafeta
|: ricardo.
Clientes -> [antonio,diogo]

1. Voltar Menu
2. Quit
|: 2.
```

Figure 4: Exemplo Query (Mais Ecologico).

5 Conclusão

Para concluir, consideramos que, nesta primeira fase do projeto, conseguimos desenvolver todas as queries com sucesso e ainda acrescentar uma query extra que achamos interessante. O objetivo da mesma é determinar o número de entregas efetuadas num determinado dia da semana.

Posto isto, cremos estar mais familiarizados com a linguagem *PROLOG* e julgamos ter tido uma boa evolução no que diz respeito à programação lógica e a entender melhor linguagens declarativas.

Sentimos que a parte da interface foi um extra bem conseguido, mas sabemos que está algo bastante simples e que poderia ser melhorado em vários aspetos, tal como a validação de input, a melhor apresentação de resultados, entre outros. O fator tempo limitou-nos para a continuação do desenvolvimento desta parte.

Assim, apesar das dificuldades que enfrentamos, acreditamos que, de uma forma geral, o grupo trabalhou bem e teve um aproveitamento positivo, visto que cumpriu com tudo o que foi pedido e ainda incluiu novas funcionalidades: uma nova query bem como a interface.