

LABORATÓRIOS DE INFORMÁTICA 3

RELATÓRIO GRUPO 88

22 NOVEMBRO DE 2023

ILUSTRE PARTICIPAÇÃO DE:

Gabriel Ribeiro a104171

João Rodrigues a104274

Tomás Barbosa a104532



Índice

Objetivo.....	3
Método.....	3
Organização.....	4
Modularização e encapsulamento.....	5
Queries.....	7

Objetivo

Com este projeto, nomeadamente com a fase 1, é pretendido processar os dados apresentados nos ficheiros csv e executar uma serie de queries, sendo estas chamadas nos ficheiros de comandos, bem como o input correspondente. Assim, para cada comando deve ser criado o ficheiro no qual será impresso o respetivo output.

Método

Para dar início á primeira parte do projeto, foi crucial o planeamento e estruturação dos vários módulos e ficheiros que dariam forma ao projeto, esquematizando o seu fracionamento para melhor precessão e clareza na sua execução, bem como a definição de prioridades no cumprimento de tarefas, queries, etc.

Com a chegada do enunciado tornou-se clara a necessidade e intenção do uso de hash tables no seu desenvolvimento, estruturas essas que ainda não havíamos usado, requerendo por nossa parte um estudo maior sobre as mesmas.

Organização

Decidimos utilizar as seguintes as seguintes estruturas de dados:

Relativamente aos utilizadores, criamos uma hash table para aceder aos dados de um utilizador de forma rápida pelo id para, por exemplo, utilizar na querie 1.

Quanto aos voos, também criamos uma hash table para aceder aos dados pelo id, e ainda uma lista de voos ordenada pelo aeroporto de origem para facilitar a querie 5.

Para as reservas criamos uma hash table e também uma lista de reservas ordenada pelo hotel para aplicar na querie 3, 4 e 8.

Por fim, no modulo das estatísticas, temos uma hash table para aceder ao número de passageiros de um voo pelo seu id, uma hash table para aceder às estatísticas de um utilizador, número de voo e reservas, e total gasto e uma lista de estatísticas dos aeroportos como número de passageiros, isto para usar na querie 6.

No que aos utilizadores diz respeito, foi organizada a informação na hash table referente aos mesmo, onde a chave usada seria o seu id.

Analogamente, foi criada a hash table com informação referente às reservas, com o id de como chave. Posteriormente, aquando a realização da querie 4, foi também criada uma lista ligada onde em cada nodo estariam os ids de cada hotel, ordenados pela data da mesma, tendo como critério de desempate o próprio id.

Por fim, também foi criada a hash table para os voos, estruturada de maneira análoga às demais, necessitando também da criação da lista ligada, desta vez com os ids dos voos, ordenados pela data, desempatado pelo id.

Modularização e Encapsulamento

Para explorar os conceitos de modularização e encapsulamento, tivemos de dividir o projeto em vários módulos com base nas suas funções.

Assim, decidimos separar o projeto nos seguintes módulos, constituídos pelos seus respetivos ficheiros .c e .h:

Interpreter: módulo responsável ler os comandos e separar as várias informações (id da *query*, argumentos, número de argumentos e a existência, ou não, de *flag* de formatação) e passar essas informações para o módulo das *queries*.

Parser: módulo onde é realizado o *parsing* dos ficheiros .csv que contêm os dados. Os vários parâmetros são inseridos na respetiva entidade, que por sua vez é guardada nas estruturas de dados. Se, por alguma razão, a entidade não for válida, esta é inserida no respetivo ficheiro de erros.

User: módulo da entidade dos utilizadores que contém a sua estrutura e todas as funções para aceder/modificar os seus parâmetros.

Reservation: módulo da entidade das reservas que contém a sua estrutura e todas as funções para aceder/modificar os seus parâmetros.

Flight: módulo da entidade dos voos que contém a sua estrutura e todas as funções para aceder/modificar os seus parâmetros.

Passenger: módulo da entidade dos passageiros que contém a sua estrutura e todas as funções para aceder/modificar os seus parâmetros.

ManageUsers: módulo de gestão dos utilizadores que contém as estruturas de dados, as funções de gestão dessas estruturas e as funções de gestão dos utilizadores. Contém as

funções que verificam a validade dos utilizadores e que processam um utilizador em caso válido e inválido.

ManageFlights: módulo de gestão dos voos que contém as estruturas de dados, as funções de gestão dessas estruturas e as funções de gestão dos voos. Contém ainda as funções que auxiliam a obter o resultado de certas *queries*.

ManageReservations: módulo de gestão das reservas que contém as estruturas de dados, as funções de gestão dessas estruturas e as funções de gestão das reservas. Também contém funções relacionadas com o resultado de algumas *queries*.

ManagePassengers: módulo de gestão dos passageiros que contém as funções de validação e processamento de um passageiro.

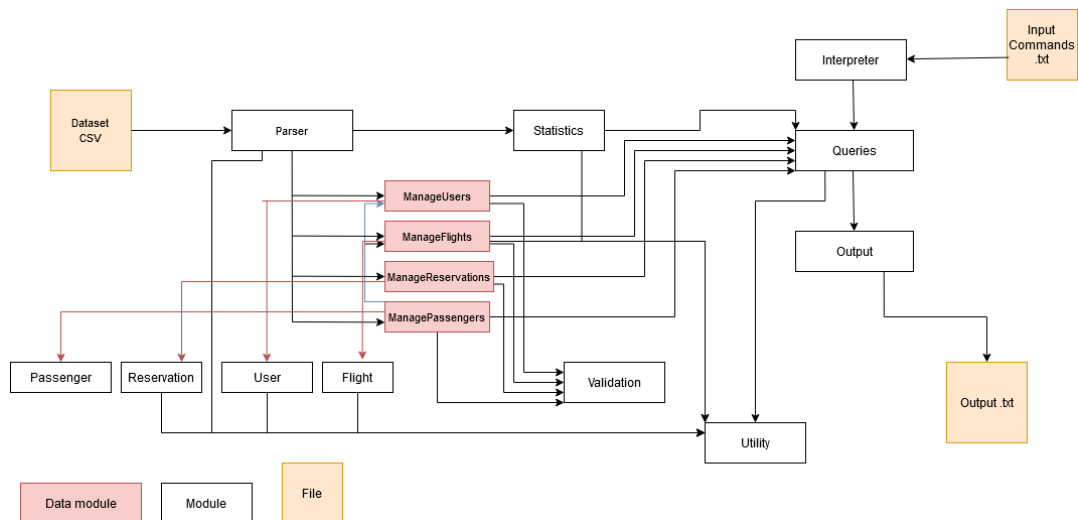
Queries: módulo responsável pela implementação das *queries*, obter o seu resultado e enviar para output.

Validation: módulo responsável pela validação dos parâmetros das várias entidades.

Output: módulo responsável pela escrita nos ficheiros de output, tanto de erros como de output de comandos.

Statistics: módulo que contém certos dados que são úteis para resolver algumas *queries*. Tem também as estruturas de dados e respetivas funções de gestão que armazenam os dados.

Utility: módulo que contém algumas funções mais genéricas que ajudam na manipulação de dados.



QUERIES

Querie 1 - inicialmente é feita uma filtragem, onde é detetado qual a entidade que o input se refere (utilizador, voo ou reserva). De seguida, conforme cada tipo, procura na respetiva hash table o id. Se não for encontrada correspondência é produzido um ficheiro vazio, caso contrário, é produzido um ficheiro com as informações necessárias para o output, sendo impressas no ficheiro de output.

Querie 3 - é corrida a lista das reservas ordenada por hotéis, sendo feito o somatório de ratings e reservas para o dado hotel, sendo impresso o resultado da média.

Querie 4 - é corrida a lista de reservas ordenada por hotéis, pesquisando as reservas do dado hotel e obtemos os parâmetros necessários para o output.

Querie 5 - se alguma das datas for invalida ou a data inicial mais recente que a final é produzida um ficheiro vazio. De seguida, é percorrida a lista de voos ordenada por aeroporto, pesquisando pelo aeroporto pretendido e obtemos as informações dos voos entre as datas.

Querie 6 - é percorrida a lista dos dados dos aeroportos ordenada por número de passageiros, obtendo o top N aeroportos com mais passageiros no dado ano.

Querie 8 - é percorrida a lista das reservas, ordenada por id dos hotéis, fazendo o somatório do preço total das reservas do hotel requerido, imprimindo o output.