

Programa IT Academy – Processo Seletivo – Edição #18

Nome Completo: Eduardo Dalmás Faé

E-mail: eduardofaeef@gmail.com

Etapa 1 – Questões de lógica

Esta seleção possui 15 questões de lógica de caráter eliminatório. As questões são apresentadas no formulário de Exercício Técnico e devem ser respondidas no próprio formulário online, que deverá ser acessado através do link a seguir: <https://forms.gle/yZtVcv1b5fCgScLBA>

Etapa 2

RESUMO DA SOLUÇÃO

A minha solução foi feita inteiramente em JS, e fez uso da biblioteca papaparse para fazer o parse do CSV e criar uma matriz de distâncias. Além disso, para rodar o código é necessário utilizar o nodeJS.

*Obs: O CSV deve estar na mesma pasta dos códigos-fontes para ser lido.

Essa matriz de distância tem sua primeira linha preenchida pelos nomes das cidades, dessa forma, para acessá-la, basta buscar na primeira linha os índices das cidades, e fazer uso desses índices para encontrar na matriz qual a distância entre as cidades. Outrossim, vale ressaltar que, por a matriz ter sua primeira linha preenchida pelo nome das cidades, ao acessar a linha da matriz deve-se somar 1 no índice procurado, para compensar a linha gasta.

Após o fim do parse, é chamada a função consultas que gere todas as consultas, ela imprime na tela um menu para o usuário e espera um input para decidir que função deve chamar.

Existem 3 funções principais:

- parseTrechos: implementa a funcionalidade 1
- parseTransportes: implementa a funcionalidade 2
- showHistorico: implementa a funcionalidade 3 (apenas a parte de impressão)

Além disso existem funções internas:

- menu: imprime o menu no console.
- getCities: recebe do usuário e valida a lista de cidades.
- printCities: imprime a lista de cidades.
- getDistances: calcula um array com a distância entre cada trecho de uma viagem.
- getTotalDistance: calcula a distância total da viagem.
- getItens: recebe do usuário uma lista de itens e checa se o padrão está correto.

- getNumItens: calcula o número de itens sendo levados na viagem.
- getItensString: recebe a lista de itens e concatena a string de saída.
- getNumTrucksByPart: calcula a quantidade de caminhões de cada tipo para cada trecho da viagem.
- getNumTrucks: calcula uma matriz com a quantidade de cada tipo de caminhão para cada trecho de uma dada viagem.
- getNumByTruckType: recebe um peso e calcula a melhor distribuição de caminhões para aquele peso.
- getNewWeight: calcula o novo peso para o novo trecho de uma viagem.
- getTruckString: recebe a lista de caminhões usados e concatena a string de saída.
- getValueByPart: calcula o valor de cada trecho da viagem.
- getValueByTruck: calcula o valor da viagem para cada tipo de caminhão.
- getTotalValue: calcula o valor total da corrida.
- getValueByItem: calcula o valor médio gasto por cada categoria de item (valorTotal / número de itens).

Estruturas criadas:

- Truck: Gerencia um tipo de caminhão - armazena tipo, custo, peso e quantidade.
- Item: Gerencia um item - armazena nome, peso e quantidade.
- Cadastro: Gerencia um cadastro - armazena custo total, custo por trecho, custo por KM, custo médio por tipo de produto, custo por tipo de caminhão, número de caminhões utilizados e número de produtos carregados.
- TruckList: Criada unicamente para facilitar entendimento em parseTrechos, o uso das funções include e getCost.

Observações finais:

Por me faltar tempo e para facilitar a implementação, não levei em consideração como cada item poderia ser subdividido, pois, para calcular o uso de caminhões utilizei unicamente o peso total de todos os itens. Isso é, em minha implementação todos os itens podem ser subdivididos. Fiz isso pois estava relativamente sem tempo e, ao analisar o problema da subdivisão ótima, percebi que ele possuía semelhanças com o problema da mochila binária. E, sabendo que o problema da mochila binária é um problema NP-Completo, preferi considerar que todos os itens podem ser fracionados, para aproximar o problema ao da mochila fracionária, que é polinomial. Assim facilitando meu trabalho, afinal estava com pouco tempo.

Espero que entendam, e desde já agradeço a oportunidade de ter participado do processo seletivo. Ele, com certeza, foi bem divertido e desafiador de se implementar!

TESTES (aqui você deverá colar capturas de tela de todas as funcionalidades desenvolvidas e realizar comentários, use o espaço que julgar necessário)

FUNCIONALIDADE 1 – Consultar trechos x modalidade

```
=====
Digite as cidades que irá passar, separadas por vírgula.
=====
> porto alegre, sao paulo
Digite o porte do caminhão utilizado para a viagem
> medio
de PORTO ALEGRE para SAO PAULO, utilizando um caminhão de medio porte, a distância é de 1109 km e o custo será de R$ 13219.28.
```

Exemplo do código funcionando.

```
=====
Digite as cidades que irá passar, separadas por vírgula.
=====
> porto alegre
Entrada não reconhecida!
```

Exemplo de erro, apenas 1 cidade.

```
=====
Digite as cidades que irá passar, separadas por vírgula.
=====
> porto alegre, teresina, vitoria
Entradas de mais (limite 2 cidades)
```

Exemplo de erro, mais de 2 cidades.

```
=====
Digite as cidades que irá passar, separadas por vírgula.
=====
> joao pessoa, cacimbinhas
Entrada não reconhecida!
```

Exemplo de erro, cidade inválida.

FUNCIONALIDADE 2 – Cadastrar transporte

```
=====
Digite as cidades que irá passar, separadas por vírgula.
=====
> porto alegre, salvador, rio de janeiro
Digite os itens deseja levar na viagem: (item1_nome, item1_peso, item1_quantidade / ...)
> grama, 1, 20 / anao de jardim, 1, 30 / mangueira, 0.5, 50
O que irá deixar em SALVADOR? (item1_nome, item1_quantidade / ...)
> grama, 10 / anao de jardim, 7 / mangueira, 14
de PORTO ALEGRE para RIO DE JANEIRO, a distância a ser percorrida é de 4739 km, para transporte dos produtos (GRAMA, ANAO DE JARDIM e MANGUEIRA) s
erá necessário utilizar 1 caminhões pequenos, 0 caminhões médios e 0 caminhões grandes, de forma a resultar no menor custo de transporte por km ro
dado. O valor total do transporte dos itens é R$ 23078.93, sendo R$ 230.79 o custo unitário médio.
```

Exemplo do código funcionando.

```
=====
Digite as cidades que irá passar, separadas por vírgula.
=====
> belem, porto alegre
Digite os itens deseja levar na viagem: (item1_nome, item1_peso, item1_quantidade / ...)
> casa, 5
Erro na entrada de itens
```

Exemplo de erro, entrada de itens não segue o padrão pedido.

****OBS:** Erros na entrada da cidade seguem a implementação da funcionalidade 1, pois são geridos pela mesma função.

FUNCIONALIDADE 3 – Dados estatísticos

```
=====
                        Consultas Realizadas
=====
[01] Custo Total: 33393.59
[01] Custo para o Trecho 01: 18525.48
[01] Custo para o Trecho 02: 2172.02
[01] Custo para o Trecho 03: 12696.09
[01] Custo medio por KM: 4.87
[01] Custo medio para o Produto macas: 667.87
[01] Produtos Transportados: 82
=====
```

Exemplo do histórico com 1 cadastro.

```
=====
                        Consultas Realizadas
=====
[01] Custo Total: 33393.59
[01] Custo para o Trecho 01: 18525.48
[01] Custo para o Trecho 02: 2172.02
[01] Custo para o Trecho 03: 12696.09
[01] Custo medio por KM: 4.87
[01] Custo medio para o Produto macas: 667.87
[01] Custo medio para o Produto geladeiras: 1113.12
[01] Custo medio para o Produto sofas: 16696.79
[01] Custo para o Caminhão Pequeno: 18525.48
[01] Custo para o Caminhão Medio: 0
[01] Custo para o Caminhão Grande: 0
[01] Veiculos deslocados: 1
[01] Produtos Transportados: 82
[02] Custo Total: 23078.93
[02] Custo para o Trecho 01: 15048.3
[02] Custo para o Trecho 02: 8030.63
[02] Custo medio por KM: 4.87
[02] Custo medio para o Produto grama: 1153.95
[02] Custo medio para o Produto anao de jardim: 769.3
[02] Custo medio para o Produto mangueira: 461.58
[02] Custo para o Caminhão Pequeno: 15048.31
[02] Custo para o Caminhão Medio: 0
[02] Custo para o Caminhão Grande: 0
[02] Veiculos deslocados: 1
[02] Produtos Transportados: 100
=====
```

Exemplo do histórico com mais de 1 cadastro.

AUTOAVALIAÇÃO

Você concluiu a implementação de 100% das funcionalidades solicitadas?

(X) Sim () Não

Para as 3 principais funcionalidades solicitadas, como você avalia a sua solução?

Marque um 'X'.

	Inexistente/ Insuficiente	Pouco satisfeito(a)	Satisfeito(a)	Muito satisfeito(a)
Funcionalidade 1				X
Funcionalidade 2			X	
Funcionalidade 3				X

Principais dificuldades

A principal dificuldade que tive foi, de longe, o tempo. Infelizmente a data do processo seletivo convergiu justamente na semana em que já possuía uma viagem programada. E, por isso, não pude mexer no programa por 4 dias, o que significa que infelizmente perdi mais da metade do tempo que poderia ter dedicado ao cumprimento da tarefa. Além disso, tive alguns problemas com a sintaxe do JS, pois já fazia um tempo que não trabalhava puramente com ele, já que atualmente ando aprendendo C++.

Desempenho Geral

Acredito ter conseguido desempenhar bem no geral, apesar dos contratempos com a viagem. Contudo gostaria de ter tido mais tempo para automatizar os testes e desenvolver uma interface melhor. Contudo, estou relativamente satisfeito quanto às funcionalidades, já que todas foram implementadas sem grandes problemas.

Obrigado por participar deste processo seletivo.
Salve o documento em PDF com o seu nome completo.