

## Programa IT Academy – Processo Seletivo – Edição #18

Nome Completo: Natália Chies

E-mail: nataliachies@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

Eu desenvolvi um programa em Python para o sistema de transporte interestadual de cargas. O programa possui diversas classes que armazenam os dados e executam funções a partir do input do usuário.

Para resolver o problema, utilizei as bibliotecas CSV, itertools e os.

Para testar o programa, utilizei a biblioteca unittest.

No programa principal, o usuário pode escolher entre diversas funcionalidades, como consultar trechos e modalidades, cadastrar transporte, obter dados estatísticos e finalizar o programa.

Para resolver o algoritmo da opção 2, utilizei força bruta, pois ainda não domino a área de programação dinâmica.

**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

```
Bem-vindo(a) ao Sistema de Transporte Interestadual de Cargas!

Digite a opção desejada:
1) Consultar trechos x modalidade
2) Cadastrar transporte
3) Dados estatísticos
4) Finalizar o programa

Opção: 1
CONSULTAR TRECHOS X MODALIDADE
Digite a cidade: salvador
Digite a cidade: porto alegre
Tamanho do caminhão (PEQUENO/MEDIO/GRANDE): grande
Para ir da cidade SALVADOR até PORTO ALEGRE num caminhão de tamanho GRANDE, a distância é de 3090km, com um custo total de R$84789.60
```

Essa funcionalidade permite a consulta da distância entre duas cidades e o tipo de transporte mais indicado para essa rota. Ela utiliza a classe *Distances*, que é responsável por ler o arquivo CSV com as distâncias entre as cidades e retornar a distância entre dois pontos. Para identificar o tipo de transporte mais indicado para essa rota, a classe *Truck* é utilizada. A função *from\_input* recebe como parâmetro a entrada do usuário (para permitir que o mesmo selecione o tipo de caminhão desejado) e retorna a distância e o custo para o tipo de caminhão selecionado.

### FUNCIONALIDADE 2 – Cadastrar transporte

```
Bem-vindo(a) ao Sistema de Transporte Interestadual de Cargas!

Digite a opção desejada:
1) Consultar trechos x modalidade
2) Cadastrar transporte
3) Dados estatísticos
4) Finalizar o programa

Opção: 2
CADASTRAR TRANSPORTE
Forneça a lista, em ordem, da sua rota.
Insira o nome da cidade (ENTER para finalizar): sao paulo
Insira o nome da cidade (ENTER para finalizar): belem
Insira o nome da cidade (ENTER para finalizar):
Sua rota é: SAO PAULO - BELEM
Item a ser transportado (ENTER para finalizar): geladeira
Peso do item (em kg): 100
Item a ser transportado (ENTER para finalizar):
Destino para o item GELADEIRA
1. BELEM
Escolha o número do destino desejado: 1
Quantidade do item GELADEIRA: 10
10 GELADEIRA (100.0kg) adicionado(s) ao carregamento.
Cargo information:
Item 1: GELADEIRA, destino: BELEM, quantidade: 10
Rota: SAO PAULO - BELEM, peso total = 1000.0kg, caminhões necessários: {'PEQUENO': 1}, distância total = 2933km, preço total = R$14283.71, preço unitário = R$142.84
```

Essa funcionalidade permite o cadastro de um novo transporte. Ela utiliza a classe *Truck*, que possui as informações dos tipos de caminhão, como capacidade e preço por quilômetro. As funções *get\_list\_of\_cities* e *read\_items* são responsáveis por criar dados a partir da entrada do usuário.

### FUNCIONALIDADE 3 – Dados estatísticos

```
Bem-vindo(a) ao Sistema de Transporte Interestadual de Cargas!

Digite a opção desejada:
1) Consultar trechos x modalidade
2) Cadastrar transporte
3) Dados estatísticos
4) Finalizar o programa

Opção: 3
DADOS ESTATÍSTICOS
['ORIGEM', 'DESTINO', 'PESO TOTAL', 'CAMINHÕES', 'DISTÂNCIA TOTAL', 'PREÇO TOTAL']
['SAO PAULO', 'BELEM', '10000.0', '{"GRANDE": 1}', '2933', '80481.52']
['SALVADOR', 'BELO HORIZONTE', '1000.0', '{"PEQUENO": 1}', '1372', '6681.64']
['SAO PAULO', 'BELEM', '5600.0', '{"PEQUENO": 2, "MEDIO": 1}', '2933', '63528.78']
['SAO PAULO', 'BELEM', '5600.0', '{"PEQUENO": 2, "MEDIO": 1}', '2933', '63528.78']
['SAO PAULO', 'MACEIO', '124000.0', '{"MEDIO": 1, "GRANDE": 12}', '2453', '836963.6000000001']
['SAO PAULO', 'MANAUS', '124000.0', '{"MEDIO": 1, "GRANDE": 12}', '3971', '1354905.2000000002']
['SAO PAULO', 'SALVADOR', '2000.0', '{"PEQUENO": 2}', '1962', '19109.88']
['SAO PAULO', 'SALVADOR', '2000.0', '{"PEQUENO": 2}', '1962', '19109.88']
['SAO PAULO', 'BELEM', '4000.0', '{"MEDIO": 1}', '2933', '34961.36']
['SAO PAULO', 'BELEM', '1000.0', '{"PEQUENO": 1}', '2933', '14283.710000000001']
```

Essa funcionalidade fornece dados estatísticos sobre o transporte, como a quantidade de caminhões necessários, o custo total e o tipo de caminhão mais utilizado. Para calcular esses dados, a classe `Truck` é utilizada. A função `get_trucks_for_leg` é usada para obter a quantidade de caminhões necessários para cada rota. Em seguida, as informações são processadas para gerar estatísticas, armazenadas em `output.csv`.

## AUTOAVALIAÇÃO

Você concluiu a implementação de 100% das funcionalidades solicitadas?

( ) Sim      ( x ) 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

As principais dificuldades encontradas foram na parte da implementação do algoritmo da opção 2, já que a solução ideal envolveria programação dinâmica. No meu algoritmo, no entanto, é utilizada força bruta para encontrar as combinações possíveis a partir do maior número possível de caminhões (expressado pela quantidade de caminhões pequenos necessários para transportar o peso total).

Além disso, encontrei dificuldades para conseguir realizar a parte do algoritmo em que apenas uma parte da quantidade total de um tipo de item pode ser descarregada em um destino, enquanto o restante segue para o outro; portanto, o algoritmo permite que seja escolhido apenas um destino para cada tipo de item. Após a descarga em um destino, conforme o peso total decresce, o algoritmo também não permite que, se possível, caminhões sejam removidos para cortar custos.

## Desempenho Geral

Escolhi a linguagem python pois estou aprendendo sobre a mesma nas aulas da graduação e julgo ser um ótimo recurso para criar rápidas interfaces com o usuário através do terminal. Como o problema, para mim, é complexo, decidi focar nos algoritmos em vez de criar um design para a interface.

No geral, estou muito satisfeita com o meu resultado, levando em conta a falta de familiaridade com alguns recursos e o meu semestre atual no curso.

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