

# Implementação N.04 - Problema de Transporte (Método Dual)

Iniciar tarefa

- Vencimento Domingo por 23:59
- Pontos 3
- Enviando um upload de arquivo
- Disponível até 1 dez em 23:59

## Descrição da tarefa

Nessa tarefa iremos explorar resolução do problema de transporte por meio de uma abordagem alternativa. Os métodos primais como o Simplex para problema de transporte. Necessitam de uma solução básica viável inicial, antes de poderem ser aplicados. Além disso, é comum dependerem do balanceamento entre a oferta total e a demanda total para obterem tais soluções iniciais.

Por outro lado, em uma abordagem dual para o problema de transporte, apenas a solução dual que consiste nas variáveis duais (ou preços/valores marginais) precisa ser viável. Já isto é muito mais simples de se obter. Além disso, basta que a oferta total seja maior ou igual a demanda total (para se garantir que há solução), tornando o balanceamento desnecessário.

Uma abordagem dual matricial (de tableau ou quadro) para o problema de transporte foi proposta no seguinte artigo:

[JI, P.; CHU, K. F. A dual-matrix approach to the transportation problem. Asia-Pacific Journal of Operational Research, v. 19, n. 1, p. 35-45, 2002.](#)

[https://pucminas.instructure.com/courses/213168/files/12031968/download?download\\_frd=1](https://pucminas.instructure.com/courses/213168/files/12031968/download?download_frd=1). 

[https://pucminas.instructure.com/courses/213168/files/12031968/download?download\\_frd=1](https://pucminas.instructure.com/courses/213168/files/12031968/download?download_frd=1)

Portanto, você deve implementar (na linguagem C, C#, C++ ou Java) um programa que receba do usuário o nome do arquivo contendo as informações/dados sobre um problema de transporte. Seu programa deverá ler o conteúdo do arquivo e aplicar o método dual descrito no artigo mencionado acima. Ao final, seu programa deve exibir o valor da solução ótima (custo total de transporte), juntamente com os valores das quantidades transportadas entre os pontos de oferta e os pontos de demanda.

Você deverá apresentar testes realizados com **02 (dois) tipos diferentes de problemas de transporte** (balanceados e desbalanceados) sendo que para cada tipo você deve apresentar **resultados para instâncias de tamanhos diferentes (pelo menos 2 tamanhos)**. Os problemas desbalanceados podem ser apenas com excesso de oferta.

OBS<sub>1</sub>: Além de entregar as instâncias testadas, você deverá entregar os resultados dos testes sob forma de um arquivo PDF.

OBS<sub>2</sub>: Você pode considerar que as maiores instâncias que seu programa irá receber possuem **10 (dez) pontos de oferta e 10 (dez) pontos de demanda**.

## Formato do arquivo contendo os dados do grafo

Seu programa deverá ler as informações sobre o grafo a partir de um arquivo texto. A primeira linha desse arquivo contém o número ***m*** de pontos de oferta seguido do número ***n*** de pontos de demanda. Você deve considerar que os pontos de ofertas são numerados (rotulados) de 1 a ***m***, enquanto os pontos de demanda são numerados de 1 a ***n***.

Depois disso, o arquivo contém uma lista com os ***m*** valores de oferta (sendo um valor por linha), seguida por uma lista com os ***n*** valores de demanda (sendo um valor por linha).

Ao final do arquivo, haverá uma matriz ***m* x *n*** contendo os custo de transporte entre cada uma dos pontos de oferta para cada um dos pontos de demanda.

Abaixo, você pode observa um esquema que representa a estrutura que deve ser esperada do arquivo:

<b><i>m</i></b>	<b><i>n</i></b>		
<Valor da Oferta 1>			
<Valor da Oferta 2>			
⋮			
<Valor da Oferta <b><i>m</i></b> >			
<Valor da Demanda 1>			
<Valor da Demanda 2>			
⋮			
<Valor da Demanda <b><i>n</i></b> >			
<Custo de 1 para 1>	<Custo de 1 para 2>		<Custo de 1 para <b><i>n</i></b> >
<Custo de 2 para 1>	<Custo de 2 para 2>		<Custo de 2 para <b><i>n</i></b> >
<Custo de <b><i>m</i></b> para 1>	<Custo de <b><i>m</i></b> para 1>		<Custo de <b><i>m</i></b> para <b><i>n</i></b> >