



Regras para Entrega:

1. A data da entrega é exclusivamente através do moodle, até 30/11/2019;
2. A nota do trabalho considera:
 1. **Legibilidade do código:** código indentado, com nomes de variáveis simples e significativos e devidamente comentado. Nos comentários iniciais deve constar o nome dos autores.
 2. **Correção do código:** o código deve ser em C e admitir compilação através do CodeBlocks na plataforma Windows, sem alterações e sem erros. No caso de necessitar de bibliotecas específicas, as informações necessárias sobre onde conseguir e como instalar devem estar devidamente documentadas, conforme o item anterior.
 3. **Originalidade do código:** cópia de código implica em nota zero para todas as cópias. A nota máxima para trabalhos corretos, dentro do esperado é 9.0. Nota 10.0 apenas para trabalhos que apresentem soluções diferenciadas, pela criatividade, elegância e eficiência acima do esperado.
 4. **Pontualidade da entrega:** atrasos na entrega acarretam descontos cumulativos na nota, assim, entrega em 30/11/2019, após o horário da aula (11:20) acarreta desconto de 1 ponto na nota do trabalho, nos dias seguintes, mais 3 pontos de desconto por dia de atraso.
3. Os grupos para execução do trabalho devem ter no máximo 03 (três) alunos.

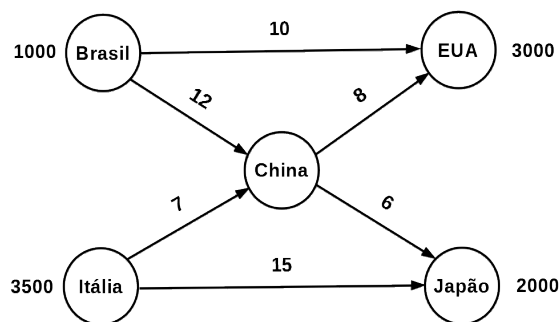
Problema:

Elaborar um programa em C, conforme as regras acima, que leia um conjunto de rotas com seus respectivos custos unitários de transporte, identifique os nós de origem, destino e transbordo e leia as respectivas ofertas para os nós de origem e as demandas para os nós de destino, considerando que nós de origem possuem apenas rotas de saída, nós de destino possuem apenas rotas de chegada, enquanto que nós de transbordo possuem tanto rotas de saída, quanto de chegada.

O programa deve:

- Identificar automaticamente o tipo de cada nó;
- Balancear oferta e demanda, acrescentando nó de origem ou destino fictício, com a respectiva oferta ou demanda, automaticamente;
- Atribuir custos proibitivos para rotas inexistentes, caso necessário;
- Calcular a solução inicial tanto pelo canto noroeste, quanto pelo menor custo e escolher automaticamente a melhor solução inicial;
- Calcular a solução ótima do problema através do método da divisão de custos;
- Durante as interações, se for necessário, o programa deve acrescentar as rotas para ajustar as equações do problema (número de origens + número de destinos = número de rotas + 1);
- Mostrar as soluções parciais e o custo para cada interação.

Para ilustrar o que se espera do programa, considere o seguinte exemplo:



e as seguintes mensagens para as entradas e saídas do programa:



origem: Brasil
destino: EUA
custo: 10
mais rotas? (S/N): s

origem: Brasil
destino: China
custo: 12
mais rotas? (S/N): s

origem: Itália
destino: China
custo: 7
mais rotas? (S/N): s

origem: Itália
destino: Japão
custo: 15
mais rotas? (S/N): s

origem: China
destino: EUA
custo: 8
mais rotas? (S/N): s

origem: China
destino: Japão
custo: 6
mais rotas? (S/N): n

Origens
oferta Brasil: 1000
oferta Italia: 3500

Destinos
demanda EUA: 3000
demanda Japão: 2000

Quadro de Custos Unitários

	Destino			
Origem	EUA	Japão	China	Oferta
Brasil	10	150	12	1000
Itália	150	15	7	3500
China	8	6	0	5000
Outros	0	0	0	500
Demanda	3000	2000	5000	

Solução Inicial Canto Noroeste

	Destino			
Origem	EUA	Japão	China	Oferta
Brasil	1000			1000
Itália	2000	1500		3500



China		500	4500	5000
Outros			500	500
Demanda	3000	2000	5000	

Custo Inicial: 335500.00

Solução Inicial Menor Custo

		Destino		
Origem	EUA	Japão	China	Oferta
Brasil	1000			1000
Itália	1500	2000		3500
China			5000	5000
Outros	500			500
Demanda	3000	2000	5000	

Custo Inicial: 265000.00

Interação 1

		Destino		
Origem	EUA	Japão	China	Oferta
Brasil	1000			1000
Itália		2000	1500	3500
China	1500		3500	5000
Outros	500			500
Demanda	3000	2000	5000	

Custo: 62500.00

Interação 2

		Destino		
Origem	EUA	Japão	China	Oferta
Brasil	1000			1000
Itália			3500	3500
China	1500	2000	1500	5000
Outros	500			500
Demanda	3000	2000	5000	

Custo: 58500.00

A otimização terminou.