Nome: Luís Felipe de Melo Costa Silva
Número USP: 9297961

Lista de Exercícios 1 - MAC0315

1 Exercícios sobre modelagem

1.1

Com os dados apresentados, fiz uma tabela:

Modelo	Valor	Demanda		Placas de RAM	Disco Rígido	Placa de vídeo
		min	max			
A	R$ 800,00	300	700	1	1	1
B	R$ 1.000,00	500	1200	2	2	1
C	R$ 1.200,00	400	600	2	1	2

A empresa possui 3000 placas de RAM, 2500 discos rígidos e 2250 placas de vídeo.
Definindo como x a quantidade de modelos A a serem produzidos, como y a de modelos B e como z a de modelos C ($x, y, z \in \mathbb{Z}$), teremos o seguinte programa linear:

$$
\begin{aligned}
\max\ & u = 800x + 1000y + 1200z \\
\text{suj}\ & 300 \leq x \leq 700 \\
& 500 \leq y \leq 1200 \\
& 400 \leq z \leq 600 \\
& x + 2y + 2z \leq 3000 \\
& x + 2y + z \leq 2500 \\
& x + y + 2z \leq 2250
\end{aligned}
$$

onde u é o lucro, que queremos maximizar.

1.2

Vamos definir como x_i a variável que indica se a antena i foi instalada, onde:

$$
x_i = \begin{cases} 1, & \text{se a antena } i \text{ foi instalada} \\ 0, & \text{c.c.} \end{cases} \tag{1}
$$

Então, teremos o seguinte programa linear:

$$
\begin{aligned}
\min\ u = \sum_{i=1}^{n} x_i c_i \\
\text{suj}\ \sum_{i=1}^{n} x_i : x_i \text{ atende a região } 1 \geq 1 \\
\vdots \\
\sum_{i=1}^{n} x_i : x_i \text{ atende a região } m \geq 1
\end{aligned}
$$

onde u é o custo, que queremos minimizar.

1.3

Definindo como x_{ij} a váriavel que indica se o caixeiro-viajante foi da cidade i direto para a cidade j, onde:

$$
x_{ij} = \begin{cases} 1, & \text{o caminho vai da cidade } i \text{ para a cidade } j \\ 0, & \text{c.c.} \end{cases} \tag{2}
$$

O modelo de programação linear usado para a resolução será:

$$
\begin{aligned}
\min\ u = \sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} x_{ij} c_{ij} \\
\text{suj}\ \sum_{i=1, i \neq j}^{n} x_{ij} = 1 \qquad & , j = 1, ..., n \\
\sum_{j=1, j \neq i}^{n} x_{ij} = 1 \qquad & , i = 1, ..., n \\
u_i - u_j + n x_{ij} \leq n - 1, & 2 \leq i \neq j \leq n
\end{aligned}
$$

Queremos minimizar o custo u. As primeiras duas restrições são para garantir que apenas um caminho sai de cada cidade, e apenas um entra na cidade. A última é para garantir que teremos apenas 1 caminho ligando as cidades, e não 2 ou mais caminhos disjuntos fazendo isso, com o uso das variáveis $u_i \in \mathbb{Z}$.

1.4

No período dos n meses, para cada período t, nós queremos minimizar o custo de produção u, que será dado por:

$$\begin{aligned}
\max\ u &= f_t + (p_t + h_t) \cdot (\sum_i^n x_i) \\
\text{suj } &\sum_i^n x_i \leq d_t \\
&\sum_i^n x_i \leq M_t
\end{aligned}$$

onde x_i é a variável que representa quantas unidades foram produzidas no mês i.

2 Exercícios sobre a estrutura de programação linear

2.1

Vamos aplicar a definição de conjunto convexo aqui, que é:

$$A \subseteq \mathbb{R}^n \text{ é convexo se } x, y \in A \implies \alpha x + (1-\alpha)y \in A$$

Se $a^T x \geq b$, então $\alpha a^T x \geq \alpha b$ (I), $\forall \alpha \in [0,1]$.
Seja $y \in \mathbb{R}^n$. Se $a^T y \geq b$, então $\alpha a^T y \geq \alpha b$, e também $(1-\alpha)a^T y \geq (1-\alpha)b$, (II) $\forall \alpha \in [0,1]$.
Somando as duas retas, temos que $\alpha a^T x + (1-\alpha)a^T y \geq b$, que mostra que o conjunto viável é convexo.

2.2

Seja S_i um conjunto convexo para $i = 1, ..., n$. Para todo $x, y \in \bigcap_{i=1}^n S_i$ e um $\alpha \in [0,1]$ implica que $\alpha x + (1-\alpha)y \in S_i$, já que S_i é convexo. Portanto, $\alpha x + (1-\alpha)y \in \bigcap_{i=1}^n S_i$, e então, $\bigcap_{i=1}^n S_i$ é convexo.

2.3

2.4

Na forma canônica, o programa fica assim (omitindo x_4, x_5 e x_6 pois não estão na função objetiva e não interferem no resultado e trocando x_1 por x, x_2 por y e x_3 por z, por conveniência):

$$
\begin{aligned}
\max\ u &= 10x + 12y + 12z \\
\text{suj}\ x + 2y + 2z - 20 &= -r \\
2x + y + 2z - 20 &= -s \\
2x + 2y + z - 20 &= -t \\
x, y, z &\geq 0
\end{aligned}
$$

Seguem os tableaux:

	x	y	z	1
-r =	1	2*	2	-20
-s =	2	1	2	-20
-t =	2	2	1	-20
	10	12	12	u = 0

	x	r	z	1
-y =	0,5	0,5	1	-10
-s =	1,5	-0,5	1	-10
-t =	1*	-1	-1	0
	4	-6	0	u = 120

	t	r	z	1
-y =	-0,5	1	1,5	-10
-s =	-1,5	0	1,5*	-10
-x =	1	-1	-1	0
	-4	-2	4	u = 120

	t	r	s	1
-y =	1*	1	-1	0
-z =	-1	0	2/3	- 20/3
-x =	0	-1	2/3	- 20/3
	0	-2	- 8/3	u = 440/3