MAT3007 HW #1

Xue Zhongkai 122090b36

1.(a)

	90h Assembly	80h Testing	Raw	Value
1st	$\frac{1}{4}$h	$\frac{1}{8}$h	\$1.2	\$9
2nd	$\frac{1}{3}$h	$\frac{1}{3}$h	\$0.9	\$8

Profit for the 1st product $z_1 = 9-1.2 = \$7.8$, No. of 1st n_1
for the 2nd product $z_2 = 8-0.9 = \$7.1$, No. of 2nd n_2

Formulate the optimization

$$\max_{n_1, n_2} \quad 7.8n_1 + 7.1n_2 \quad \text{s.t.} \quad \begin{aligned} &\tfrac{1}{4}n_1 + \tfrac{1}{3}n_2 \le 90 \\ &\tfrac{1}{8}n_1 + \tfrac{1}{3}n_2 \le 80 \\ &n_1, n_2 \ge 0 \end{aligned}$$

(b) As the standard form.

$$\min_{\substack{n_1, n_2 \\ S_1, S_2}} \quad -7.8n_1 - 7.1n_2 \quad \text{s.t.} \quad \begin{aligned} &\tfrac{1}{4}n_1 + \tfrac{1}{3}n_2 + S_1 = 90 \\ &\tfrac{1}{8}n_1 + \tfrac{1}{3}n_2 \qquad + S_2 = 80 \\ &n_1, n_2, S_1, S_2 \ge 0 \end{aligned}$$

(c) Let n_3 be the overtime labor used.
Re-formulate the optimization as

$$\max_{n_1, n_2, n_3} \quad 7.8n_1 + 7.1n_2 - 7n_3 \quad \text{s.t.} \quad \begin{aligned} &\tfrac{1}{4}n_1 + \tfrac{1}{3}n_2 + n_3 \le 140 \\ &\tfrac{1}{8}n_1 + \tfrac{1}{3}n_2 \le 80 \\ &n_3 \le 50 \\ &n_1, n_2, n_3 \ge 0 \end{aligned}$$

(d) Refer to the Python file attached as 1d.py
Produce 360 units of the 1st product and 0 for the 2nd.
The optimal daily product is roughly \$2,808.

2. Introduce u, v, w s.t. $u \ge |x_1 - x_3|$
$v \ge |x_1 + 2|$
$w \ge |x_2|$

Reformulate the problem as

$$\min_{\substack{x_1, x_2, x_3 \\ u, v, w}} \quad 2x_2 + u \quad \text{s.t.} \quad \begin{aligned} &u \ge x_1 - x_3 \\ &u \ge x_3 - x_1 \\ &v \ge x_1 + 2 \\ &v \ge -x_1 - 2 \\ &w \ge x_2 \\ &w \ge -x_2 \\ &v + w \le 5 \\ &x_3 \ge -1 \\ &x_3 \le 1 \end{aligned}$$

3. Set x_{ijg} to be the No. of students assigned from neighbourhood i to school j's grade g.

The objective could be formulated as

$$\sum_{i\in I}\sum_{j\in J}\sum_{g\in G} d_{ij}x_{ijg}$$

For the constraints, we mainly consider three:

$$\forall j\in J,\ g\in G,\ \sum_{i\in I} x_{ijg} \le C_{jg}$$

$$\forall i\in I,\ g\in G,\ \sum_{j\in J} x_{ijg} = S_{ig}$$

$x_{ijg} \ge 0$ of course

As a result, we formulate it as

$$\min \sum_{i\in I}\sum_{j\in J}\sum_{g\in G} d_{ij}x_{ijg} \quad \text{s.t.} \quad \sum_{i\in I} x_{ijg} \le C_{jg}$$

$$\sum_{j\in J} x_{ijg} = S_{ig}$$

$$x_{ijg} \ge 0$$

4. Set x_{ij} to be the No. of cars moving from i to j ($i \ne j$, cannot move to itself)

$$\min \sum_{i=1}^{5}\sum_{j=1, j\ne i}^{5} Cost_{ij}\cdot x_{ij}$$

$$\text{s.t.} \quad \sum_{j=1}^{5} x_{ij} \le Crt_i$$ (For each region j, No. of cars moving out cannot exceed current ones)

$$\sum_{\substack{i=1\\ i\ne j}}^{5} x_{ij} - \sum_{\substack{i=1\\ i\ne j}}^{5} x_{ji} + Crt_j \ge Needed_j$$ (For each region i, the difference between No. of cars moved in and out, combined with current ones should satisfy the needs)

Refer to the file attached 4.py

From the matrix, we move 115 cars from 4 to 2;
165 cars from 4 to 3;
85 cars from 5 to 1;
225 cars from 5 to 3.

The optimal cost is \$11,370 for the movement.

5. Refer to the file attached 5_raw.py, 5.py

If constraints are released to [0,1] (5.py)
then the sol. returns as [0.5, 0.5 ... 0.5]; value returns as 5.00

If constraints remain $x\in\{0,1\}$ (5_raw.py)
the sol. should be [0,1,0,1,1,1,1,1,0,0]; value returns as 6

Thus we could not remove integrality constraints.

END OF HW#1 122090636