Title: DIET and MIXING Model

1-What is the meaning of Operations Research? What are the different phases for operations research?

Operations Research in the USA , south Africa and Australia and Operations Research in Europe and Canda.

It is interdisciplinary branch of applied mathematics and formal sciences that uses methods such as mathematical modeling,statistics and algorthims to arrive at optimal or near optimal solutions to complex problem .it is typically concerned with optimizing the maxima or minima of some objective function Operations Research helps management achieve its goals using scientific methods. The first formal activities of OR were initiated in England warII when a team of British scientists set out to make scientifically based decisions regarding the best utilization of war material After the war the ideas advanced in military operation were adapted to improve effecience and productivity in the civilian sector .

The different phases for operations research is :

1-Problem Definition (1-Decision variable .2-objective .3constrains)==(Formulate the problem).

2-Model Constrains (translate the problem definition into mathematical relationships .if the resulling model fits one of standard mathematical model such as LP .we can usually reach a solution by using available algorithm)==Develop a model.

3-Select appropriate data input.

4-Model Solution (it is far the simplest of all OR phase).

5-Model Validity (checks whether or not the proposed model does what it purport to do).

6-Implemention (the translation of the result into understandable operation instructions to be issued to the people who will administer the recommended system).

2- Formulate the Mathematical Linear Programming for this problem

1Decision Variable : $x1$**:The number of units of food of type 1.** $x2$**:The number of units of food of type 2.** $x3$**:The number of units of food of type 3.** $x4$**:The number of units of food of type 4. 2Objective :**

Min :z=45 x1+40 x2+85 x3+65 x4.

3-Constraint:

$3x1+4x2+8x3+6x4 \geq 800$, for proteins

$2x1+2x2+7x3+5x4 \geq 200$, for fats

$6x1+4x2+7x3+4x4 \geq 700$, for carbohydrates

where $x1,x2,x3,x4 \geq 0$ hidden.

3-Define the meaning of basic feasible solution of a general Linear Programming Problem.

In the theory of linear programming, a basic feasible solution (BFS) is a solution with a minimal set of non-zero variables. Geometrically, each BFS corresponds to a corner of the polyhedron of feasible solutions. If there exists an optimal solution, then there exists an optimal BFS.

4- Solve the problem using Regular Simplex method.

Modeling : z=45 x1+40 x2+85 x3+65 x4.

Subject to

$3x1+4x2+8x3+6x4 \geq 800$, for proteins

$2x1+2x2+7x3+5x4 \geq 200$, for fats

$6x1+4x2+7x3+4x4 \geq 700$, for carbohydrates

where $x1,x2,x3,x4 \geq 0$ hidden. Add 3 surplus and 3 artificial.

1. As the constraint-1 is of type '$\geq$' we should subtract surplus variable $S1$ and add artificial variable $A1$

2. As the constraint-2 is of type '$\geq$' we should subtract surplus variable $S2$ and add artificial variable $A2$

3. As the constraint-3 is of type '$\geq$' we should subtract surplus variable $S3$ and add artificial variable $A3$

After adding

Min $Z = 45\,x1 + 40\,x2 + 85\,x3 + 65\,x4 + 0\,S1 + 0\,S2 + 0\,S3 + M\,A1 + M\,A2 + M\,A3$ **subject to**

$3x1 + 4x2 + 8x3 + 6x4 - S1 + A1 = 800$

$2x1 + 2x2 + 7x3 + 5x4 - S2 + A2 = 200$

$6x1 + 4x2 + 7x3 + 4x4 - S3 + A3 = 700$ **1-tablon or table**

Iteration-1		***Cj***	**45**	**40**	**85**	**65**	**0**	**0**	**0**	***M***	***M***	***M***	
B	*CB*	*XB*	*x*1	*x*2	*x*3	*x*4	*S*1	*S*2	*S*3	*A*1	*A*2	*A*3	**MinRatio *XBx*3**
*A*1	*M*	800	3	4	8	6	-1	0	0	1	0	0	**8008=100**
*A*2	*M*	200	2	2	(7)	5	0	-1	0	0	1	0	**2007=28.5714→**
*A*3	*M*	700	6	4	7	4	0	0	-1	0	0	1	**7007=100**
Z=1700*M*		*Z*	11*M*	10*M*	22*M*	15*M*	-*M*	-*M*	-*M*	*M*	*M*	*M*	
		Z-*Cj*	11*M*-45	10*M*-40	22*M*85↑	15*M*-65	-*M*	-*M*	-*M*	0	0	0	

Enter =*x*3, Drop =*A*2.

R2=R2/7, R1=R1-8*R2, R3=R3-7R.

2-tablon or table

Iteration2		Cj	45	40	85	65	0	0	0	M	M	M	
B	CB	XB	x1	x2	x3	x4	S1	S2	S3	A1	A2	A3	MinRatio XBx1
A1	M	571.4286	0.7143	1.7143	0	0.2857	-1	1.1429	0	1	-1.1429	0	571.42860.7143 =800
x3	85	28.5714	(0.2857)	0.2857	1	0.7143	0	-0.1429	0	0	0.1429	0	28.57140.2857= 100→
A3	M	500	4	2	0	-1	0	1	-1	0	-1	1	5004=125
Z=1071.4286M+2428.5714		Zj	4.7143M+24.2857	3.7143M+24.2857	85	-0.7143M+60.7143	-M	2.1429M-12.1429	-M	M	-2.1429M+12.1429	M	
		Zj-Cj	4.7143M-20.7143↑	3.7143M-15.7143	0	-0.7143M-4.2857	-M	2.1429M-12.1429	-M	0	-3.1429M+12.1429	0	

Enter $=x1$, Drop $=x3$.

R2=R2/0,2857, R1=R1-0,7143R2 , R3=R3-4R2.

3-tablon or table

Iteration-3		Cj	45	40	85	65	0	0	0	M	M	M	
B	CB	XB	$x1$	$x2$	$x3$	$x4$	$S1$	$S2$	$S3$	$A1$	$A2$	$A3$	MinRatio $XBS2$
$A1$	M	500	0	1	-2.5	-1.5	-1	1.5	0	1	-1.5	0	5001.5=333.3333
$x1$	45	100	1	1	3.5	2.5	0	-0.5	0	0	0.5	0	---
$A3$	M	100	0	-2	-14	-11	0	(3)	-1	0	-3	1	1003=33.3333→
Z=600 M+4500		Zj	45	-M+45	-16.5M+157.5	-12.5M+112.5	-M	4.5M-22.5	-M	M	-4.5M+22.5	M	
		Zj Cj	0	-M+5	-16.5M+72.5	-12.5M+47.5	-M	4.5M-22.5↑	-M	0	-5.5M+22.5	0	

Enter =$S2$, Drop =$A3$.
R3=R3/3, R1=R1-1.5R3, R2=R2+.5R3. 4-tablon
or table

Iteration-4		Cj	45	40	85	65	0	0	0	M	M	M	

B	*CB*	*XB*	*x*1	*x*2	*x*3	*x*4	*S*1	*S*2	*S*3	*A*1	*A*2	*A*3	MinRatio *XBx*3
*A*1	*M*	450	0	2	(4.5)	4	-1	0	0.5	1	0	-0.5	4504.5=100 →
*x*1	45	116.6667	1	0.6667	1.1667	0.6667	0	0	-0.1667	0	0	0.1667	116.66671.1667=100
*S*2	0	33.3333	0	-0.6667	-4.6667	-3.6667	0	1	-0.3333	0	-1	0.3333	---
Z=450*M*+5250		*Zj*	45	2*M*+30	4.5*M*+52.5	4*M*+30	-*M*	0	0.5*M*-7.5	*M*	0	-0.5*M*+7.5	
		Zj-*Cj*	0	2*M*-10	4.5*M*-32.5 ↑	4*M*-35	-*M*	0	0.5*M*-7.5	0	-*M*	-1.5*M*+7.5	

Enter =*x*3, Drop =*A*1

R1=R1/4.5, R2=R2-1.1667R1, R3=R3-4.6667R1. 5-tablon or table

Iteration-5		*Cj*	45	40	85	65	0	0	0	*M*	*M*	*M*	
B	*CB*	*XB*	*x*1	*x*2	*x*3	*x*4	*S*1	*S*2	*S*3	*A*1	*A*2	*A*3	MinRatio *XBx*2

*x*3	85	100	0	0.4444	1	0.8889	-0.2222	0	0.1111	0.2222	0	-0.1111	1000.4444=225
*x*1	45	0	1	(0.1481)	0	-0.3704	0.2593	0	-0.2963	-0.2593	0	0.2963	00.1481=0 →
*S*2	0	500	0	1.4074	0	0.4815	-1.037	1	0.1852	1.037	-1	-0.1852	5001.4074=355.2632
Z=8500		*Zj*	45	44.4444	85	58.8889	-7.2222	0	-3.8889	7.2222	0	3.8889	
		Zj-*Cj*	0	4.4444↑	0	-6.1111	-7.2222	0	-3.8889	-*M*+7.2222	-*M*	-*M*+3.8889	

Enter =x2, Drop =x1.
R2=R2/.01481.
R1=R1-.4444R2.
R3=R3-1.4074R2.
6-tablon or table

Iteration6		*Cj*	45	40	85	65	0	0	0	*M*	*M*	*M*	
B	*C B*	*XB*	*x*1	*x* 2	*x* 3	*x*4	*S*1	*S* 2	*S* 3	*A*1	*A* 2	*A*3	MinRatio *XBx*4
*x*3	85	100	-3	0	1	(2)	-1	0	1	1	0	-1	1002=50 →
*x*2	40	0	6.75	1	0	-2.5	1.75	0	-2	-1.75	0	2	---

*S*2	0	500	-9.5	0	0	4	-3.5	1	3	3.5	-1	-3	5004=125
Z=8500		*Zj*	15	40	85	70	-15	0	5	15	0	-5	
		Zj-*Cj*	-30	0	0	5↑	-15	0	5	-*M*+15	-*M*	-*M*-5	

Enter =*x*4, Drop =*x*3.

R1=R1/2.

R2=R2+2.5R1.

R3=R3-4R1.

7-tablon or table

Iteration-7		***Cj***	**45**	**40**	**85**	**65**	**0**	**0**	**0**	***M***	***M***	***M***	
B	***C B***	***XB***	***x*1**	***x*2**	***x*3**	***x*4**	***S*1**	***S*2**	***S*3**	***A*1**	***A*2**	***A*3**	**MinRatio *XBS*3**
***x*4**	65	50	-1.5	0	0.5	1	-0.5	0	(0.5)	0.5	0	-0.5	500.5=100 →
***x*2**	40	125	3	1	1.25	0	0.5	0	0.75	-0.5	0	0.75	---
***S*2**	0	300	-3.5	0	-2	0	-1.5	1	1	1.5	-1	-1	3001=300

***Z*=8250**		***Zj***	**22.5**	**40**	**82.5**	**65**	**12.5**	**0**	**2.5**	**12.5**	**0**	**-2.5**	
		Zj*-*Cj	**22.5**	**0**	**-2.5**	**0**	**12.5**	**0**	**2.5** ↑	**-*M*+12.5**	**-*M***	**-*M*-2.5**	

Enter=*S*3, Drop =*x*4.
R1=R1/0.5
R2=R2-.75R1.
R3=R3-R1.
8-tablon or table

Iteration8		***Cj***	**45**	**40**	**85**	**65**	**0**	**0**	**0**	***M***	***M***	***M***	
B	***CB***	***XB***	***x*1**	***x*2**	***x*3**	***x*4**	***S*1**	***S*2**	***S*3**	***A*1**	***A*2**	***A*3**	**MinRatio**
***S*3**	**0**	**100**	**-3**	**0**	**1**	**2**	**-1**	**0**	**1**	**1**	**0**	**-1**	
***x*2**	**40**	**200**	**0.75**	**1**	**2**	**1.5**	**-0.25**	**0**	**0**	**0.25**	**0**	**0**	
***S*2**	**0**	**200**	**-0.5**	**0**	**-3**	**-2**	**-0.5**	**1**	**0**	**0.5**	**-1**	**0**	
***Z*=8000**		***Zj***	**30**	**40**	**80**	**60**	**-10**	**0**	**0**	**10**	**0**	**0**	
		Zj*-*Cj	**-15**	**0**	**-5**	**-5**	**-10**	**0**	**0**	**-*M*+10**	**-*M***	**-*M***	

X2=200,x1=0,x3=0,x4=0 then z=40*200=8000.
Final tablon .

5- Find the Dual Problem, and determine which is computationally advantageous to solve either the Primal or the Dual, then solve both problems

1-Dual problem :
Max=b^t y
A^ty<=c y>=0

max z= 800 y1+200 y2+700 y3.
Subject to
3 y1+2 y2+6 y3<=45,
4 y1+2 y2+4 y3<=40,
8 y1+7 y+7 y3<=85, 6
y1+5 y2+4 y3<=65.
2-Determine

CE=(constrains+1)(no of variable +constrains + artificial+1)(artificial+1).
CEP=(3+1)(4+3+3+1)(3+1)=176.
CED=(4+1)(3+4+0+1)(0+1)=40
Dual problem is better than primal problem . 3-
solve Dual problem
z= 800 y1+200 y2+700 y3+0s1+0s2+0s3+0s4.
3 y1+2 y2+6 y3+s1=45,
4 y1+2 y2+4 y3+s2=40,
8 y1+7 y+7 y3+s3=85, 6
y1+5 y2+4 y3+s4=65.

		Y1 800	Y2 200	Y3 700	S1 0	S2 0	S3 0	S4 0	
S1	0	3	2	6	1	0	0	0	45
S2	0	4	2	4	0	1	0	0	40
S3	0	8	7	7	0	0	1	0	85
S4	0	6	5	4	0	0	0	1	68
		-800	-200	-700	0	0	0	0	

Enter y1,Drop s2. R2=R2/4

R1=R1-3R2
R3=R3-8R2
R4=R4-6R2

		Y1 800	Y2 200	Y3 700	S1 0	S2 0	S3 0	S4 0	
S1	0	0	1/2	3	1	-3/4	0	0	15
Y1	800	1	1/2	1	0	1/4	0	0	10
S3	0	0	3	-1	0	-2	1	0	5
S4	0	0	2	2	0	-3/2	0	1	5
		0	200	100	0	200	0	0	

So
S1=x1=0,s2=x2=200,s3=x3=0,s4=x4=0; Z=200*40=8000.

4-solve primal with regular simplex methods (in number 4)

Modeling : z=45 x1+40 x2+85 x3+65 x4.

Subject to

$3x1+4x2+8x3+6x4 \geq 800$, for proteins

$2x1+2x2+7x3+5x4 \geq 200$, for fats

$6x1+4x2+7x3+4x4 \geq 700$, for carbohydrates where $x1,x2,x3,x4 \geq 0$ hidden.

Add 3 surplus and 3 artificial.

1. As the constraint-1 is of type '≥' we should subtract surplus variable $S1$ and add artificial variable $A1$

2. As the constraint-2 is of type '≥' we should subtract surplus variable $S2$ and add artificial variable $A2$

3. As the constraint-3 is of type '≥' we should subtract surplus variable $S3$ and add artificial variable $A3$

After adding

Min $Z = 45\,x1 + 40\,x2 + 85\,x3 + 65\,x4 + 0\,S1 + 0\,S2 + 0\,S3 + M\,A1 + M\,A2 + M\,A3$

subject to

$3x_1 + 4x_2 + 8x_3 + 6x_4 - S_1 \qquad + A_1 \qquad = 800$

$2x_1 + 2x_2 + 7x_3 + 5x_4 \quad - S_2 \qquad + A_2 \quad = 200$

$6x_1 + 4x_2 + 7x_3 + 4x_4 \qquad - S_3 \qquad + A_3 = 700$ **1-tablon or table**

Iteration-1		***Cj***	**45**	**40**	**85**	**65**	**0**	**0**	**0**	***M***	***M***	***M***	
B	CB	XB	$x1$	$x2$	$x3$	$x4$	$S1$	$S2$	$S3$	$A1$	$A2$	$A3$	**MinRatio *XBx*3**
$A1$	M	800	3	4	8	6	-1	0	0	1	0	0	**8008=100**
$A2$	M	200	2	2	(7)	5	0	-1	0	0	1	0	**2007=28.57 14→**
$A3$	M	700	6	4	7	4	0	0	-1	0	0	1	**7007=100**
Z=1700 M		Z	11M	10M	22M	15M	-M	-M	-M	M	M	M	
		Z-Cj	11M-45	10M-40	22M85↑	15M-65	-M	-M	-M	0	0	0	

Enter =x3, **Drop** =A2.

R2=R2/7, R1=R1-8*R2, R3=R3-7R.

2-tablon or table

Iteration2		***Cj***	**45**	**40**	**85**	**65**	**0**	**0**	**0**	***M***	***M***	***M***	

B	*CB*	*XB*	*x*1	*x*2	*x*3	*x*4	*S*1	*S*2	*S*3	*A*1	*A*2	*A*3	MinRatio *XBx*1
*A*1	*M*	571.4286	0.7143	1.7143	0	0.2857	-1	1.1429	0	1	-1.1429	0	571.42860.7143=800
*x*3	85	28.5714	(0.2857)	0.2857	1	0.7143	0	-0.1429	0	0	0.1429	0	28.57140.2857=100→
*A*3	*M*	500	4	2	0	-1	0	1	-1	0	-1	1	5004=125
Z=1071.4286*M*+2428.5714		*Zj*	4.7143*M*+24.2857	3.7143*M*+24.2857	85	-0.7143*M*+60.7143	-*M*	2.1429*M*-12.1429	-*M*	*M*	-2.1429*M*+12.1429	*M*	
		Zj-*Cj*	4.7143*M*-20.7143↑	3.7143*M*-15.7143	0	-0.7143*M*-4.2857	-*M*	2.1429*M*-12.1429	-*M*	0	-3.1429*M*+12.1429	0	

Enter =*x*1, Drop =*x*3.
R2=R2/0,2857, R1=R1-0,7143R2 , R3=R3-4R2.

3-tablon or table

Iteration-3		*Cj*	45	40	85	65	0	0	0	*M*	*M*	*M*	

B	CB	XB	x1	x2	x3	x4	S1	S2	S3	A1	A2	A3	MinRatio XBS2
A1	M	500	0	1	-2.5	-1.5	-1	1.5	0	1	-1.5	0	5001.5=333.3333
x1	45	100	1	1	3.5	2.5	0	-0.5	0	0	0.5	0	---
A3	M	100	0	-2	-14	-11	0	(3)	-1	0	-3	1	1003=33.3333→
Z=600M+4500		Zj	45	-M+45	-16.5M+157.5	-12.5M+112.5	-M	4.5M-22.5	-M	M	-4.5M+22.5	M	
		Zj Cj	0	-M+5	-16.5M+72.5	-12.5M+47.5	-M	4.5M-22.5↑	-M	0	-5.5M+22.5	0	

Enter =S2, Drop =A3.
R3=R3/3, R1=R1-1.5R3, R2=R2+.5R3. 4-tablon or table

Iteration-4		Cj	45	40	85	65	0	0	0	M	M	M	

B	CB	XB	x1	x2	x3	x4	S1	S2	S3	A1	A2	A3	MinRatio XBx3
A1	M	450	0	2	(4.5)	4	-1	0	0.5	1	0	-0.5	4504.5=100 →
x1	45	116.6667	1	0.6667	1.1667	0.6667	0	0	-0.1667	0	0	0.1667	116.66671.1667=100
S2	0	33.3333	0	-0.6667	-4.6667	-3.6667	0	1	-0.3333	0	-1	0.3333	---
Z=450M+5250		Zj	45	2M+30	4.5M+52.5	4M+30	-M	0	0.5M-7.5	M	0	-0.5M+7.5	
		Zj-Cj	0	2M-10	4.5M-32.5 ↑	4M-35	-M	0	0.5M-7.5	0	-M	-1.5M+7.5	

Enter =x3, Drop =A1

R1=R1/4.5, R2=R2-1.1667R1, R3=R3-4.6667R1. 5-tablon or table

Iteration-5		Cj	45	40	85	65	0	0	0	M	M	M	
B	CB	XB	x1	x2	x3	x4	S1	S2	S3	A1	A2	A3	MinRatio XBx2

*x*3	85	100	0	0.4444	1	0.8889	-0.2222	0	0.1111	0.2222	0	-0.1111	1000.4444=225
*x*1	45	0	1	(0.1481)	0	-0.3704	0.2593	0	-0.2963	-0.2593	0	0.2963	00.1481=0 →
*S*2	0	500	0	1.4074	0	0.4815	-1.037	1	0.1852	1.037	-1	-0.1852	5001.4074=355.2632
Z=8500		*Zj*	45	44.4444	85	58.8889	-7.2222	0	-3.8889	7.2222	0	3.8889	
		Zj-*Cj*	0	4.4444↑	0	-6.1111	-7.2222	0	-3.8889	-*M*+7.2222	-*M*	-*M*+3.8889	

Enter =x2, Drop =x1.
R2=R2/.01481.
R1=R1-.4444R2.
R3=R3-1.4074R2.
6-tablon or table

Iteration6		*Cj*	45	40	85	65	0	0	0	*M*	*M*	*M*	
B	*C B*	*XB*	*x*1	*x*2	*x*3	*x*4	*S*1	*S*2	*S*3	*A*1	*A*2	*A*3	MinRatio *XBx*4
*x*3	85	100	-3	0	1	(2)	-1	0	1	1	0	-1	1002=50 →
*x*2	40	0	6.75	1	0	-2.5	1.75	0	-2	-1.75	0	2	---

*S*2	0	500	-9.5	0	0	4	-3.5	1	3	3.5	-1	-3	5004=125
Z=8500		*Zj*	15	40	85	70	-15	0	5	15	0	-5	
		Zj-*Cj*	-30	0	0	5↑	-15	0	5	-*M*+15	-*M*	-*M*-5	

Enter =*x*4, **Drop** =*x*3.

R1=R1/2.

R2=R2+2.5R1.

R3=R3-4R1.

7-tablon or table

Iteration-7		***Cj***	**45**	**40**	**85**	**65**	**0**	**0**	**0**	***M***	***M***	***M***	
B	***C B***	***XB***	***x*1**	***x*2**	***x*3**	***x*4**	***S*1**	***S*2**	***S*3**	***A*1**	***A*2**	***A*3**	**MinRatio *XBS*3**
*x*4	65	50	-1.5	0	0.5	1	-0.5	0	(0.5)	0.5	0	-0.5	500.5=100 →
*x*2	40	125	3	1	1.25	0	0.5	0	0.75	-0.5	0	0.75	---
*S*2	0	300	-3.5	0	-2	0	-1.5	1	1	1.5	-1	-1	3001=300

Z=8250		*Zj*	22.5	40	82.5	65	12.5	0	2.5	12.5	0	-2.5	
		Zj-*Cj*	22.5	0	-2.5	0	12.5	0	2.5 ↑	-*M*+12.5	-*M*	-*M*-2.5	

Enter=*S*3, Drop =*x*4.
R1=R1/0.5
R2=R2-.75R1.
R3=R3-R1.
8-tablon or table

Iteration8		*Cj*	45	40	85	65	0	0	0	*M*	*M*	*M*	
B	*CB*	*XB*	*x*1	*x*2	*x*3	*x*4	*S*1	*S*2	*S*3	*A*1	*A*2	*A*3	MinRatio
*S*3	0	100	-3	0	1	2	-1	0	1	1	0	-1	
*x*2	40	200	0.75	1	2	1.5	-0.25	0	0	0.25	0	0	
*S*2	0	200	-0.5	0	-3	-2	-0.5	1	0	0.5	-1	0	
Z=8000		*Zj*	30	40	80	60	-10	0	0	10	0	0	
		Zj-*Cj*	-15	0	-5	-5	-10	0	0	-*M*+10	-*M*	-*M*	

X2=200,x1=0,x3=0,x4=0 then z=40*200=8000.
Final tablon .

6- Solve the Problem Using Dual Simplex Method, if possible

The condition is met min+ so can solve it but Need to adjust the constrains

1-constrains hit *-1 to <=
Then add slacks

Min :z=45 x1+40 x2+85 x3+65 x4+0s1+0s2+0s3

-3*x*1-4*x*2-8*x*3-6*x*4+s1=-800, for proteins
-2*x*1-2*x*2-7*x*3-5*x*4-s2=-200, for fats
-6*x*1-4*x*2-7*x*3-4*x*4+s3=-700, for carbohydrates where
***x*1,*x*2,*x*3,*x*4 ,s1,s2,s3 ≥0 hidden.**

		X1	X2	X3	X4	S1	S2	S3	
		45	40	85	65	0	0	0	
S1	0	-3	-4	-8	-6	1	0	0	-800
S2	0	-2	-2	-7	-5	0	1	0	-200
S3	0	-6	-4	-7	-4	0	0	1	-700
		45	40	85	65	0	0	0	

Enter x2,Drop s1
R1=R1*-1THEN R1=R1/4
R2=R2-2R1
R3=R3-4R1

		X1	X2	X3	X4	S1	S2	S3	
		45	40	85	65	0	0	0	
X2	40	3/4	1	2	3/2	-1/4	0	0	200
S2	0	-1/2	0	-3	-2	-1/2	1	0	200

S3	**0**	**-3**	**0**	**1**	**2**	**-1**	**0**	**1**	**100**
		15	**0**	**5**	**5**	**10**	**0**	**0**	

X2=200,x1=0,x3=0,x4=0 then z=40*200=8000.

7- By using any computer software (e.g. TORA, MATLAB, …) solve this optimization problem, or write down a program that solve the Problem using any programming language you know (program one method ONLY; Regular method, Dual Simplex method, or Dual Problem)
(Reference 1)
https://cbom.atozmath.com/CBOM/Simplex.aspx?q=sm&q1=4%603%60MIN%60Z%60x1%2Cx2%2Cx3%2Cx4%6045%2C40%2C85%2C65%603%2C4%2C8%2C6%3B2%2C2%2C7%2C5%3B6%2C4%2C7%2C4%60%3E%3D%2C%3E%3D%2C%3E%3D%60800%2C200%2C700%60%60D%60false%60true%60false%60false%60false%60false%60true&do=1#tblSolution

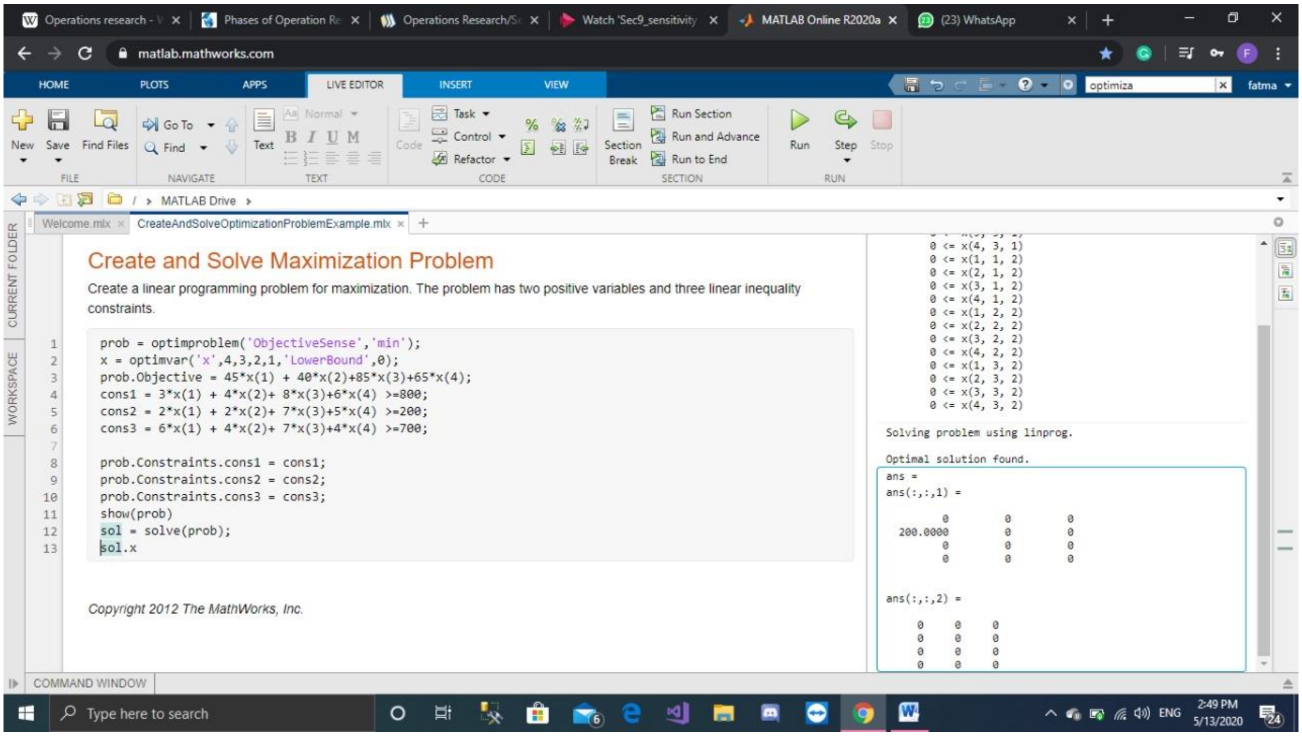
Or

From this link the code from matlab program :
https://matlab.mathworks.com/
the code:

```
prob = optimproblem('ObjectiveSense','min');
x = optimvar('x',4,3,2,1,'LowerBound',0);
prob.Objective = 45*x(1) + 40*x(2)+85*x(3)+65*x(4);
cons1 = 3*x(1) + 4*x(2)+ 8*x(3)+6*x(4) >=800;
cons2 = 2*x(1) + 2*x(2)+ 7*x(3)+5*x(4) >=200;
cons3 = 6*x(1) + 4*x(2)+ 7*x(3)+4*x(4) >=700;

prob.Constraints.cons1 = cons1;
prob.Constraints.cons2 = cons2;
prob.Constraints.cons3 = cons3;
show(prob)
sol = solve(prob);
```

sol.x



To show the out put on the program .