

NAME: SHUBHAM SHARMA

ROLL NO: 18i190002

MSC PHD (OR)

Exercise 1):

Q1) [R]

We have to find an optimal location for the new facility ,We have given the following table of the Feeders with their co-ordinates and weights.

FEEDER	CORDINATE	QUANTITY(w_i)
A	(0.5,0,0)	6
B	(10,10,0)	9
C	(0,0,5)	3
D	(2,10,0)	12
E	(3,0,15)	5
F	(8,0,8)	8
G	(0,12,0)	8
H	(15,1,0)	5

We have to formulate a lpp with the given data so that we can set up new facility.

Let the coordinates of the new facility are (a,b,c) , so our objective is to minimize the weighted distance of this new

facility from every other facility, we will take the Euclidian distance to solve this.

DEFINING THE OBJECTIVE FUNCTION:

$$\begin{aligned} \text{MINIMIZE dist: } & 6\sqrt{(0.5 - a)^2 + (0 - b)^2 + (0 - c)^2} \\ & + 9\sqrt{(10 - a)^2 + (10 - b)^2 + (0 - c)^2} \\ & + 3\sqrt{(0 - a)^2 + (0 - b)^2 + (5 - c)^2} + \\ & 12\sqrt{(2 - a)^2 + (10 - b)^2 + (0 - c)^2} + \\ & 5\sqrt{(3 - a)^2 + (0 - b)^2 + (15 - c)^2} + \\ & 8\sqrt{(8 - a)^2 + (0 - b)^2 + (8 - c)^2} \\ & + 8\sqrt{(0 - a)^2 + (12 - b)^2 + (0 - c)^2} \\ & + 5\sqrt{(15 - a)^2 + (1 - b)^2 + (0 - c)^2} \end{aligned}$$

Q3) [R]

The lpp formulated above has been solved by AMPL and the code files(ex1.mod,ex2.run,ex1.dat) has been attached in the zipped file.

Console

AMPL

```
ampl: include ex1.run;  
SNOPT 7.5-1.2 : Optimal solution found.  
13 iterations, objective 457.3728012  
Nonlin evals: obj = 13, grad = 12.  
dist = 457.373  
  
a = 3.88991  
b = 7.39618  
c = 1.46601  
  
ampl:
```

1) The objective value came out to be:457.373(taken the Euclidian distance)

2) The values are variables at optimal solution is:

a(x coordinate of the new facility)=3.88991

b(y coordinate of the new facility)=7.39618

c(z coordinate of the new facility)=1.46601

3)The number of iterations solver took to solve it=13 iterations(highlighted in the screenshot)

Q4) [R]

The following question has been solved on the last page...

Exercise 2):

Q1) [R]

As we have to formulate a mathematical program , we will define some sets and parameters that will help us defining the lpp....

DEFINING SETS 'S', 'CORD':

Let $S=\{1,2,3,4,5\}$ (where each $i \in S, i \neq 5$ denotes the corner i as per given in the question and 5 represents the center of the lake)

Let $CORD=\{x,y\}$ (that will help us defining the coordinate of every corner and the center of the lake)

DEFINING THE PARAMETERS:-

Let us define a parameter $LOCATIONS_{S*CORD}$ that will define the coordinates of every coordinates.

CORNER	x-coordinate	y-coordinate
1	0	0
2	-100	-100
3	0	-200
4	150	-50

5(center of lake)	200	200
-------------------	-----	-----

(These coordinates are calculated as per the data given is the question and the unit of measure is meters.)

Now let us define an another parameter radius that will tell us the radius of the given lake:-

Radius = 30meters.

NOW LET US DEFINE THE DECISION VARIABLES:-

(As per defined is the question itself)

Let x_f = x coordinate of the end point of the farm.

Let y_f = y coordinate of the end point of the farm.

Let x_l = x coordinate of the end point of the lake.

Let y_l = y coordinate of the end point of the lake.

DEFINING THE OBJECTIVE FUNCTION:-

MINIMIZE dist: $\sqrt{(x_l - x_f)^2 + (y_l - y_f)^2}$

Subject to constrains:-

(As we are given that (x_f, y_f) must lie inside the farm, we will define four constrains such that these coordinates must lie in each line of the polyhedron)

(we are defining the equation of line in terms of parameters as if someone change the corners and want to see the changes in the optimal point then we dont have to find the equation every time)

a) (defining the first constraint):

$$y_f - \text{LOCATION}_{1,y} \leq ((\text{LOCATION}_{1,y} - \text{LOCATION}_{2,y}) / (\text{LOCATION}_{1,x} - \text{LOCATION}_{2,x})) * (x_f - \text{LOCATION}_{1,x})$$

(i.e, it will denote the line between corner 1 and 2)

b) (defining the second constraint):

$$yf-LOCATION_{1,y} \leq ((LOCATION_{1,y} - LOCATION_{4,y}) / (LOCATION_{1,x} - LOCATION_{4,x})) * (xf - LOCATION_{1,x})$$

(i.e, it will denote the line between corner 1 and 4)

c) (defining the third constraint):

$$yf-LOCATION_{3,y} \geq ((LOCATION_{3,y} - LOCATION_{4,y}) / (LOCATION_{3,x} - LOCATION_{4,x})) * (xf - LOCATION_{3,x})$$

(i.e, it will denote the line between corner 3 and 4)

d) (defining the forth constraint):

$$yf-LOCATION_{3,y} \geq ((LOCATION_{3,y} - LOCATION_{2,y}) / (LOCATION_{3,x} - LOCATION_{2,x})) * (xf - LOCATION_{3,x})$$

(i.e, it will denote the line between corner 2 and 3)

(Defining the last constraint as we are given that (xl,yl) must inside the lake):

e) (defining the fifth constraint):

$$(xl-LOCATION_{5,x})^2 + (yl-LOCATION_{5,y})^2 \leq RADIUS^2$$

Q3) [R]

The following question has been solved using AMPL and the code files(ex2.run,ex2.mod,ex2.dat) has been attached in the zipped file:-

```

AMPL
ampl: include ex2.run;
SNOPT 7.5-1.2 : Error evaluating objective function: can't evaluate sqrt'(0).
SNOPT 7.5-1.2 : Bug: "scream" called.
-11111 iterations
Nonlin evals: obj = 1, grad = 1, constra = 1, Jac = 1.
SNOPT 7.5-1.2 : Optimal solution found.
20 iterations, objective 222.9822128
Nonlin evals: obj = 16, grad = 15, constra = 16, Jac = 15.
x1 = 190.513
y1 = 171.54
xf = 120
yf = -40

dist = 222.982

ampl:

```

a)The value of the objective value(taken the Euclidian distance) is=222.982

b)The location of two ends are:-

x1(x coordinate of the lake)=190.513

y1(y coordinate of the lake)=171.54

xf(x coordinate of the farm)=120

yf(y coordinate of the farm)=-40

Q4) [R]

As we have to formulate a mathematical program , we will define some sets and parameters that will help us defining the lpp....

DEFINING SETS 'S', 'CORD':

Let $S=\{1,2,3,4,5\}$ (where each $i \in S$, $i \neq 5$ denotes the corner i as per given in the question and 5 represents the center of the lake)

Let $CORD=\{x,y\}$ (that will help us defining the coordinate of every corner and the center of the lake)

DEFINING THE PARAMETERS:-

Let us define a parameter $LOCATIONS_{S*CORD}$ that will define the coordinates of every coordinates.

CORNER	x-coordinate	y-coordinate
1	0	0
2	-100	-100
3	0	-200
4	150	-50
5(center of lake)	200	200

(These coordinates are calculated as per the data given is the question and the unit of measure is meters.)

Now let us define an another parameter radius that will tell us the radius of the given lake:-

Radius = 30meters.

NOW LET US DEFINE THE DECISION VARIABLES:-

(As per defined is the question itself)

Let x_f = x coordinate of the end point of the farm.

Let y_f = y coordinate of the end point of the farm.

Let x_l = x coordinate of the end point of the lake.

Let y_l = y coordinate of the end point of the lake.

Let y_0 = y coordinate of the junction.

Let x_0 = x coordinate of the junction.

(The mathematical program will be same as that we described in question 1 but the only changes will be the new

decision variables x_0 and y_0 and the some slightest changes in the objective function)

WLOG , Let the unit cost of pipe from lake to junction and from junction to the farm is Rs 100 ,then it is given is the question that The pipe from the junction to the store-house is thinner and its unit-cost is only 20% of that of the pipe from lake to junction and that from the junction to the farm. So the unit cost of the pipe from junction to store house is 20% of 100 i.e, 20.

(We can let it to be any positive value as it wont make any difference in the solution(i.e, the coordinates))

DEFINING THE OBJECTIVE FUNCTION:-

$$\begin{aligned} \text{MINIMIZE wtdist: } & 20 \sqrt{(x_0 - 15)^2 + (y_0 - 20)^2} + \\ & 100 \sqrt{(x_0 - x_l)^2 + (y_0 - y_l)^2} + \\ & 100 \sqrt{(x_0 - x_f)^2 + (y_0 - y_f)^2} \end{aligned}$$

(i.e, it is the sum of three **weighted** distances, one is from the junction to the point on the lake , other is the from the junction the point in the farm and the last one is from the junction the store-house)

Where (15,20) are the x and y coordinates of the store-house as per the data given in the question.

(As we are given that (x_f, y_f) must lie inside the farm, we will define four constraints such that these coordinates must lie in each line of the polyhedron)

(we are defining the equation of line in terms of parameters as if someone change the corners and want to see the changes in the optimal point then we don't have to find the equation every time)

a) (defining the first constraint):

$$y_f - \text{LOCATION}_{1,y} \leq ((\text{LOCATION}_{1,y} - \text{LOCATION}_{2,y}) / (\text{LOCATION}_{1,x} - \text{LOCATION}_{2,x})) * (x_f - \text{LOCATION}_{1,x})$$

(i.e, it will denote the line between corner 1 and 2)

b) (defining the second constraint):

$$y_f - \text{LOCATION}_{1,y} \leq ((\text{LOCATION}_{1,y} - \text{LOCATION}_{4,y}) / (\text{LOCATION}_{1,x} - \text{LOCATION}_{4,x})) * (x_f - \text{LOCATION}_{1,x})$$

(i.e, it will denote the line between corner 1 and 4)

c) (defining the third constraint):

$$y_f - \text{LOCATION}_{3,y} \geq ((\text{LOCATION}_{3,y} - \text{LOCATION}_{4,y}) / (\text{LOCATION}_{3,x} - \text{LOCATION}_{4,x})) * (x_f - \text{LOCATION}_{3,x})$$

(i.e, it will denote the line between corner 3 and 4)

d) (defining the fourth constraint):

$$y_f - \text{LOCATION}_{3,y} \geq ((\text{LOCATION}_{3,y} - \text{LOCATION}_{2,y}) / (\text{LOCATION}_{3,x} - \text{LOCATION}_{2,x})) * (x_f - \text{LOCATION}_{3,x})$$

(i.e, it will denote the line between corner 2 and 3)

(Defining the last constraint as we are given that (x_l, y_l) must inside the lake):

e) (defining the fifth constraint):

$$(x_l - \text{LOCATION}_{5,x})^2 + (y_l - \text{LOCATION}_{5,y})^2 \leq \text{RADIUS}^2$$

Q6) [R]

The following question has been solved using AMPL and the code files(ex2b.run,ex2b.mod,ex2b.dat) has been attached in the zipped file:-

```
Console
AMPL
ampl: include ex2b.run;
SNOPT 7.5-1.2 : Error evaluating objective function: can't evaluate sqrt'(0).
SNOPT 7.5-1.2 : Bug: "scream" called.
-11111 iterations
Nonlin evals: obj = 1, grad = 1, constrs = 1, Jac = 1.
SNOPT 7.5-1.2 : Optimal solution found.
28 iterations, objective 24199.51093
Nonlin evals: obj = 23, grad = 22, constrs = 23, Jac = 22.
x1 = 185.039
y1 = 173.997
xf = 74.468
yf = -24.8227
x0 = 79.7245
y0 = -9.05328

dist = 24199.5

ampl:
```

a)The value of the objective value(taken the Euclidian distance) is=24199.5

b)The location of two ends are:-

x_l (x coordinate of the lake)=185.039

y_l (y coordinate of the lake)=173.997

x_f (x coordinate of the farm)=74.468

y_f (x coordinate of the fram)=-24.8227

c)The location of the junction is:

x_0 (x coordinate of the junction)=79.7245

y_0 (y coordinate of the junction)=-9.05328

Exercise 3):

Q1) [R]

In this question we are given a right angled triangle with perpendicular and base as 5 inches each. One way to approach this question is that we can take the vertex with right angle as (0,0). Then, as per the data given in the question, the coordinates of the vertex of the triangle will be A(0,0), B(0,5), C(5,0).

Therefore, we are dealing with three lines:

- 1) $x=0$
- 2) $y=0$
- 3) $x+y=5$

Given that each coin has a circle of radius $2/3 \approx 0.67$ inches and each coin must lie inside the board(triangle).

Let (x_i, y_i) be the coordinate of the i^{th} coordinate.

Then, The linear constraints to force each coin to lie inside the triangular box are :-

(For this to happen the distance of each points from these lines should be greater than or equal to radius)

- a) $x_i \geq 0.67$
- b) $y_i \geq 0.67$
- c) $((x_i + y_i - 5) / \sqrt{2}) \leq -0.67$

Q2) [R]

In this question, we have to write the non-linear constraints that forbids the coins from overlapping each other, we can do this by putting the constraints:-

$$\sqrt{(x[i] - x[j])^2 + (y[i] - y[j])^2} \geq 0.67 * 2$$

for all i,j and $i \neq j$.

Here the constraints means that distance between the centres of any two coins is greater than or equal to twice the radius.

Q6) [R]

As it is given in the question to try different starting values of x, y in the model to obtain a feasible solution, we have got the solution after putting these values $x[1]=0.67, y[1]=0.67$, $x[2]=1.34, y[2]=0.67$.

The following question has been solved by AMPL and the commands are attached(ex3.run,ex3.dat,ex3.mod) in the zipped file.

Console

```
AMPL
ampl: include ex3.run;
SNOPT 7.5-1.2 : Error evaluating constraint 1: can't evaluate sqrt'(0).
SNOPT 7.5-1.2 : Bug: "scream" called.
-11111 iterations
Nonlin evals: constrs = 1, Jac = 1.
Objective = find a feasible point.
SNOPT 7.5-1.2 : Feasible point found.
1 iterations, objective 0
Nonlin evals: constrs = 3, Jac = 2.
Objective = find a feasible point.
:      x      y      :=
1  0.67  0.67
2  2.01  0.67
;

:      con1.ub  con1.lb  con1.body      :=
1 2  Infinity   1.34     1.34
2 1  Infinity   1.34     1.34
;

:      con2.ub  con2.lb  con2.body      :=
1  Infinity    0.67     0.67
2  Infinity    0.67     2.01
;

:      con3.ub  con3.lb  con3.body      :=
1  Infinity    0.67     0.67
2  Infinity    0.67     0.67
;

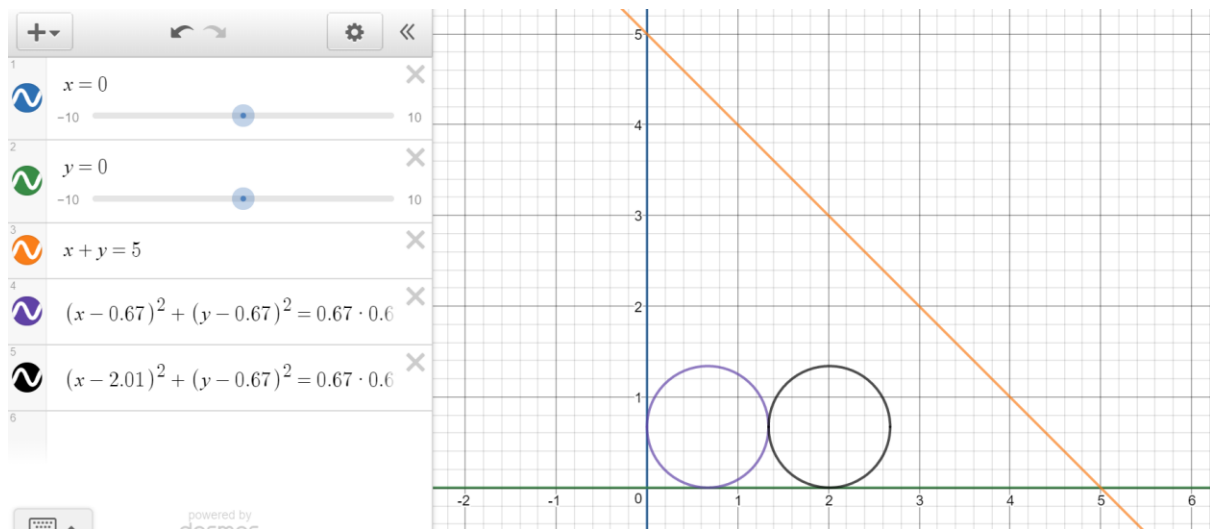
:      con4.ub   con4.lb  con4.body      :=
1  4.05248     -Infinity  1.34
2  4.05248     -Infinity  2.68
;
```

The feasible point has been found in 1 iterations.

The coordinates of the first coin are: $x[1]=0.67, y[1]=0.67$

The coordinates of the second coin are: $x[2]=2.01, y[2]=0.67$

It can be easily seen that these points are feasible as the each constraint is satisfied and for our sake we can also see it graphically by plotting these coins:



Q7) [R]

The codes have been attached in the zipped file(ex3b.run,ex3b.mod,ex3b.dat)with the # used to ensure that AMPL don't read some things in some cases.

SOLVING PROBLEM FOR 4 COINS:

We have solved the question through AMPL by giving some initial values to the variables as shown in the snapshot:

the initial values are:

let $x[1]:=0.67;$

let $y[1]:=0.67;$

let $x[2]:=1.34;$

let $y[2]:=0.67;$

```
let x[3]:=0.67;
let y[3]:=3.35;
let x[4]:=0.67*3;

let y[4]:=0.67*3;
```

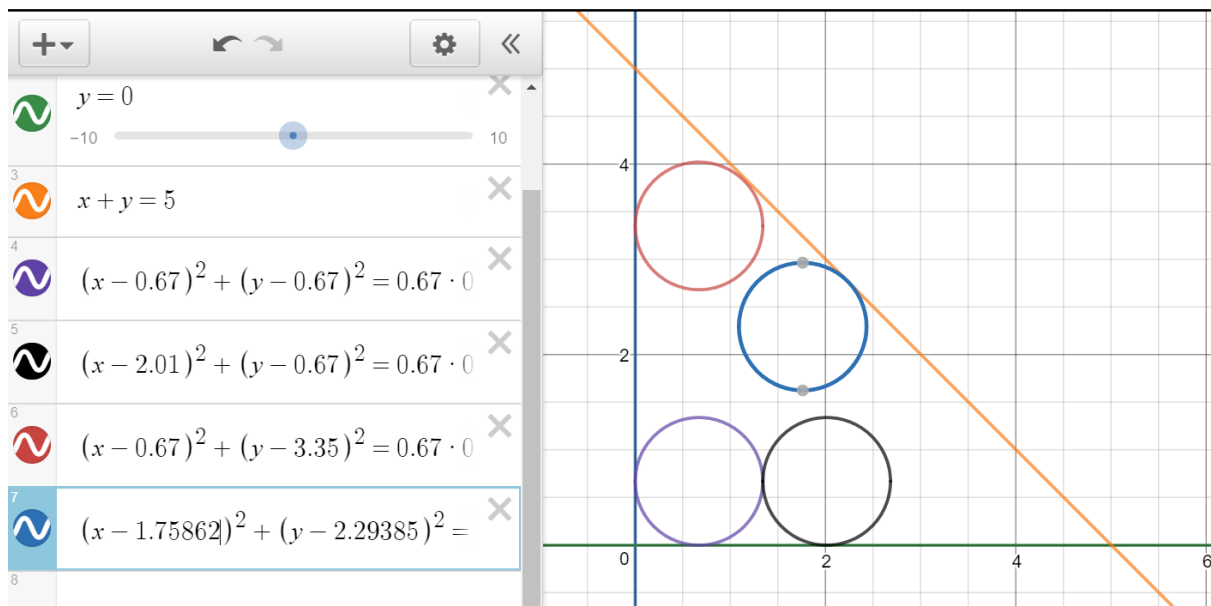
Now solving it in ampl , we get a feasible solution as highlighted in the solution below:

```
Console
AMPL
ampl: include ex3b.run;
SNOPT 7.5-1.2 : Error evaluating constraint 1: can't evaluate sqrt'(0).
SNOPT 7.5-1.2 : Bug: "scream" called.
-11111 iterations
Nonlin evals: constrs = 1, Jac = 1.
Objective = find a feasible point.
SNOPT 7.5-1.2 : Feasible point found.
5 iterations, objective 0
Nonlin evals: constrs = 3, Jac = 2.
Objective = find a feasible point.
:      x      y      :=
1   0.67    0.67
2   2.01    0.67
3   0.67    3.35
4   1.75862 2.29385
;

ampl:
```

The value of x and y coordinates for 4 coins are :

i	x	Y
1	0.67	0.67
2	2.01	0.67
3	0.67	3.35
4	1.75862	2.29385



NOW,

SOLVING PROBLEM FOR 6 COINS:

We have solved the question through AMPL by giving some initial values to the variables as shown in the snapshot:

the initial values are:

```

let x[1]:=0.67;
let y[1]:=0.67;
let x[2]:=1.34;
let y[2]:=0.67;
let x[3]:=0.67;
let y[3]:=3.35;
let x[4]:=0.67*3;
let y[4]:=0.67*3;
let x[5]:=0.7;
let y[5]:=0.67*3;
let x[6]:=0.67*5;

```

let y[6]:=0.67;

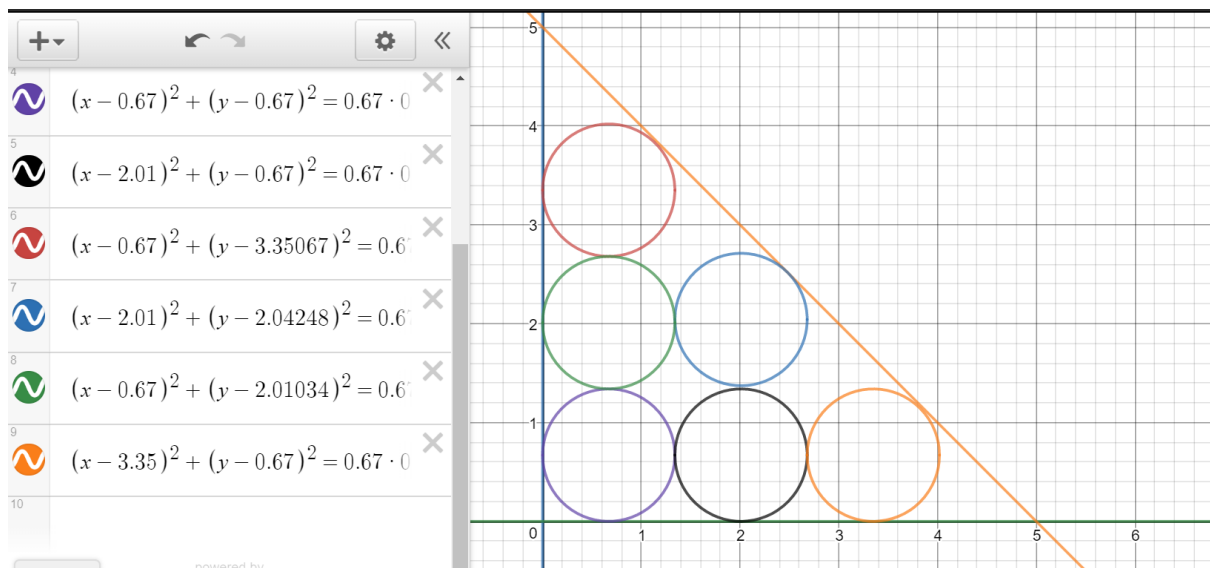
Now solving it in ampl , we get a feasible solution as highlighted in the solution below:

```
Console
AMPL
ampl: include ex3b.run;
SNOPT 7.5-1.2 : Error evaluating constraint 1: can't evaluate sqrt'(0).
SNOPT 7.5-1.2 : Bug: "scream" called.
-11111 iterations
Nonlin evals: constrs = 1, Jac = 1.
Objective = find a feasible point.
SNOPT 7.5-1.2 : Feasible point found.
16 iterations, objective 0
Nonlin evals: constrs = 3, Jac = 2.
Objective = find a feasible point.
:      x      y      :=
1    0.67    0.67
2    2.01    0.67
3    0.67    3.35067
4    2.01    2.04248
5    0.67    2.01034
6    3.35    0.67
;

ampl:
```

The value of x and y coordinates for 4 coins are :

i	x	Y
1	0.67	0.67
2	2.01	0.67
3	0.67	3.35067
4	2.01	2.04248
5	0.67	2.01034
6	3.35	0.67



SOLVING PROBLEM FOR 8 COINS:

While solving for 8 coins it will always give infeasible solution as Area of a coin * 8 is > the area of the board.

But we have tried to give different values to x and y for different i^s and the **solution is always infeasible**, hence the solution is always infeasible for $i=8$. One of such examples is the snapshot below:-

```
Console
AMPL
ampl: include ex3b.run;
SNOPT 7.5-1.2 : Error evaluating constraint 1: can't evaluate sqrt'(0).
SNOPT 7.5-1.2 : Bug: "scream" called.
-11111 iterations
Nonlin evals: constrs = 1, Jac = 1.
Objective = find a feasible point.
SNOPT 7.5-1.2 : Nonlinear infeasibilities minimized.
23 iterations, objective 0
Nonlin evals: constrs = 9, Jac = 8.
Objective = find a feasible point.
:      x      y      :=
1   0.67    0.67
2   1.552   0.67
3   0.67    3.38248
4   2.222   1.83047
5   1.61752 2.43495
6   3.38248 0.67
7   0.67    2.01
8   2.892   0.67
;

ampl:
```

SOLVING PROBLEM FOR 10 COINS:

While solving for 10 coins it will always give infeasible solution as Area of a coin * 10 is > the area of the board.

But we have tried to give different values to x and y for different i 's and the **solution is always infeasible**, hence the solution is always infeasible for $i=10$. One of such examples is the snapshot below:-

```

,
ampl: include ex3b.run;
SNOPT 7.5-1.2 : Error evaluating constraint 1: can't evaluate sqrt'(0).
SNOPT 7.5-1.2 : Bug: "scream" called.
-11111 iterations
Nonlin evals: constrs = 1, Jac = 1.
Objective = find a feasible point.
SNOPT 7.5-1.2 : Error evaluating constraint 18: can't evaluate sqrt'(0).
SNOPT 7.5-1.2 : Bug: "scream" called.
-11111 iterations
Nonlin evals: constrs = 1, Jac = 1.
Objective = find a feasible point.
:      x      y      :=
1      0.67    0.67
2      1.34    0.67
3      0.67    3.35
4      2.01    2.01
5      0.7     2.01
6      3.35    0.67
7      0.67    1.71
8      2.85    1.14
9      0.67    3.38248
10     1.34    0.67
;

ampl: |

```

SOLVING PROBLEM FOR 12 COINS:

While solving for 12 coins it will always give infeasible solution as Area of a coin * 12 is > the area of the board.

But we have tried to give different values to x and y for different i 's and the **solution is always infeasible**, hence the solution is always infeasible for $i=12$. One of such examples is the snapshot below:-

```
Console
AMPL
ampl: include ex3b.run;
SNOPT 7.5-1.2 : Error evaluating constraint 1: can't evaluate sqrt'(0).
SNOPT 7.5-1.2 : Bug: "scream" called.
-11111 iterations
Nonlin evals: constrs = 1, Jac = 1.
Objective = find a feasible point.
SNOPT 7.5-1.2 : Error evaluating constraint 8: can't evaluate sqrt'(0).
SNOPT 7.5-1.2 : Bug: "scream" called.
-11111 iterations
Nonlin evals: constrs = 3, Jac = 2.
Objective = find a feasible point.
:      x      y      :=
1      0.67      0.67
2      1.34      0.67
3      0.67      3.35
4      2.01      2.01
5      0.7       2.01
6      3.35      0.67
7      0.67      1.71
8      2.85      1.14
9      0.7       0.7
10     1.4       0.7
11     1.05248   3
12     2         1
;

ampl: |
```

(Q5) [R] is on the last page)

Exercise 4):

Q1) [R]

The following question has been solved by AMPL and the code files (ex4.mod,ex4.run) has been attached in the zipped file.

Console

AMPL

```
ampl: include ex4.run;  
SNOPT 7.5-1.2 : Optimal solution found.  
41 iterations, objective -15.30409721  
Nonlin evals: obj = 105, grad = 104, constrs = 105, Jac = 104.  
objective = -15.3041  
  
x1 = -20.4772  
x2 = -17.8155  
  
con.ub = 0  
con.body = 4.06813e-08  
con.lb = -Infinity  
  
ampl: |
```

The Optimal value of the objective function is
: -15.3041

The Solution is :

$x_1 = -20.4772$

$x_2 = -17.8115$

Q2) [R]

The solution has given 8 different starting values and the result is as follows:

```
AMPL
ampl: include ex4.run;
SNOPT 7.5-1.2 : Optimal solution found.
41 iterations, objective -15.30409721
Nonlin evals: obj = 105, grad = 104, constra = 105, Jac = 104.
objective = -15.3041

x1 = -20.4772
x2 = -17.8155

con.ub = 0
con.body = 4.06813e-08
con.lb = -Infinity

objective = 16.0638

x1 = 1
x2 = 34

con.ub = 0
con.body = 1860360
con.lb = -Infinity

objective = 2.65984

x1 = 3
x2 = 4

con.ub = 0
con.body = 597300
con.lb = -Infinity

objective = 23.0443

x1 = 16
x2 = 40
AMPL
con.ub = 0
con.body = 2024870
con.lb = -Infinity

objective = 9.70828

x1 = 18
x2 = 4

con.ub = 0
con.body = 519870
con.lb = -Infinity

objective = 5.97399

x1 = 15
x2 = 7

con.ub = 0
con.body = 374208
con.lb = -Infinity

objective = 11.9004

x1 = 19
x2 = 20

con.ub = 0
con.body = -3116
con.lb = -Infinity

objective = 7.19356

x1 = 10
x2 = 12
con.lb = -Infinity

objective = 7.19356

x1 = 10
x2 = 12

con.ub = 0
con.body = 310878
con.lb = -Infinity

ampl: |
```

The values that we have supposed are in the snapshot and every solutions is satisfying the constraints so every solution is optimal.

Q3) [R]

We are observing different optimal solutions as The function is highly convex as we can think of sine function which has infinite maxima when defined on \mathbb{R} .

Exercise 3):

Q5) [R]

The feasible region is not convex because of the non linear constraints. If the constraints were linear then the feasible region would be convex. As if the constraints in a linear program are linear, they will always produce a convex body. With finitely many constraints, it will in fact be a convex polytope.

Exercise 1):

Q4) [R]

We have to find the minimum quantity that should flow from feeder-B so that the optimal location of the main reactor is within 4 units of feeder B.

We can do this by playing with the quantity flowing from the optimal location to the Feeder-B. We can do this by giving different values to quantity from B. We have done this in AMPL and the files(ex1b.mod,ex1b.run,ex1b.dat) are attached in the zipped file.

AMPL

```
ampl: include ex1b.run;
SNOPT 7.5-1.2 : Optimal solution found.
13 iterations, objective 457.3728012
Nonlin evals: obj = 13, grad = 12.
sqrt((COORDINATE[2,1] - a)^2 + (COORDINATE[2,2] - b)^2 + (COORDINATE[2,3] - c)^2) =
6.80164
```

QUANTITY[2] = 9

```
ampl: include ex1b.run;
SNOPT 7.5-1.2 : Optimal solution found.
12 iterations, objective 477.0473186
Nonlin evals: obj = 12, grad = 11.
sqrt((COORDINATE[2,1] - a)^2 + (COORDINATE[2,2] - b)^2 + (COORDINATE[2,3] - c)^2) =
6.30757
```

QUANTITY[2] = 12

```
ampl: include ex1b.run;
SNOPT 7.5-1.2 : Optimal solution found.
13 iterations, objective 495.1661253
Nonlin evals: obj = 13, grad = 12.
sqrt((COORDINATE[2,1] - a)^2 + (COORDINATE[2,2] - b)^2 + (COORDINATE[2,3] - c)^2) =
5.76156
```

QUANTITY[2] = 15

```
ampl: include ex1b.run;
SNOPT 7.5-1.2 : Optimal solution found.
13 iterations, objective 511.5486224
Nonlin evals: obj = 13, grad = 12.
sqrt((COORDINATE[2,1] - a)^2 + (COORDINATE[2,2] - b)^2 + (COORDINATE[2,3] - c)^2) =
5.14747
```

QUANTITY[2] = 18

QUANTITY[2] = 18

```
ampl: include ex1b.run;
SNOPT 7.5-1.2 : Optimal solution found.
14 iterations, objective 525.9668547
Nonlin evals: obj = 14, grad = 13.
sqrt((CORDINATE[2,1] - a)^2 + (CORDINATE[2,2] - b)^2 + (CORDINATE[2,3] - c)^2) =
4.44915
```

QUANTITY[2] = 21

```
ampl: include ex1b.run;
SNOPT 7.5-1.2 : Optimal solution found.
14 iterations, objective 530.2896206
Nonlin evals: obj = 14, grad = 13.
sqrt((CORDINATE[2,1] - a)^2 + (CORDINATE[2,2] - b)^2 + (CORDINATE[2,3] - c)^2) =
4.19439
```

QUANTITY[2] = 22

```
ampl: include ex1b.run;
SNOPT 7.5-1.2 : Optimal solution found.
14 iterations, objective 534.3515149
Nonlin evals: obj = 14, grad = 13.
sqrt((CORDINATE[2,1] - a)^2 + (CORDINATE[2,2] - b)^2 + (CORDINATE[2,3] - c)^2) =
3.92725
```

QUANTITY[2] = 23

ampl:

We have taken many values for Quantity flowing from feeder 2 and the distance of feeder B from the optimal location is coming as follows:-

DIFFERENT VALUE FOR QUANTITY FLOWING FROM B	THE DISTANCE OF FEEDER B FROM THE OPTIMAL LOCATION
9	6.80164
12	6.30757
15	5.76156
18	5.14747
21	4.44915
22	4.19439
23	3.92725

As we can see in the table above , the value of the distance of feeder B from the optimal location is decreasing as we are increasing the values of Quantity flowing from B. Then , When **23 units** are flowing from FEEDER B to the new facility then we get the minimum quantity that should flow from feeder-B so that the optimal location of the main reactor is within 4 units of feeder-B.
