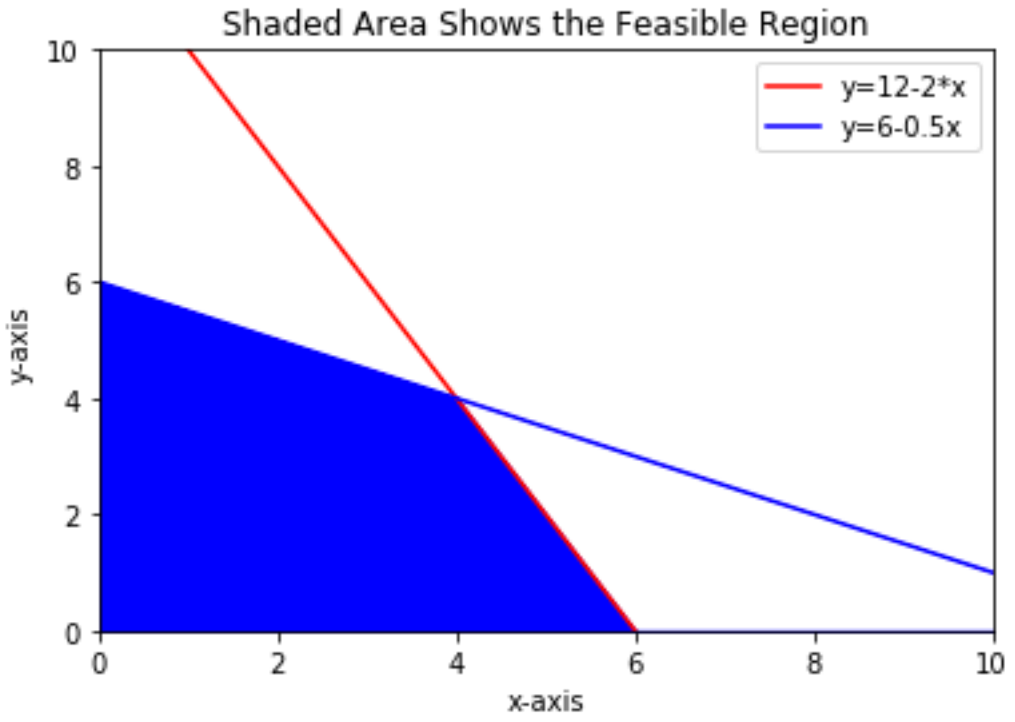
Module 3 Answers

Module 3 Practice 1

Exercise 1: Refer to Lial Section 3.1 page 118. Write the code to reproduce Figure 10. Compare your code and the resulting plot to the answer sheet.
It is possible to fill the feasible region by specifying the corner points of the polygon. The solution shown below uses the fill_between() command with the where= statement to specify the plotting region. Also, the xlim() and ylim() commands give horizontal and vertical boundaries.

```
import matplotlib.pyplot
from matplotlib.pyplot import *
import numpy
from numpy import arange
figure()
x= arange(0,10.1,0.1)
y1= 12.0 - 2.0*x
y2= 6.0 - 0.5*x
xlim(0,10)
ylim(0,10)
xlabel('x-axis')
ylabel('y-axis')
plot(x,y1, c='r', label='y=12-2x')
plot(x,y2, c='b', label='y=6-0.5x')
legend(['y=12-2*x','y=6-0.5x'],loc='best')
fill_between(x,y2,where=(y2<=y1), color= 'b')
fill_between(x,y1,where=(y1<=y2), color= 'b')
title ('Shaded Area Shows the Feasible Region')
show()
```



Module 3 Practice 3

Exercise 1: Refer to Lial Section 3.2 Example 3. Using matrix methods, evaluate the objective function at each corner point and determine both the maximum and the minimum. Compare your code and solutions with the answer sheet.

```
import numpy
from numpy import *
from numpy.linalg import *

x= [6,6,4,0]
y= [6,3,2,3]
obj= matrix([1.0,10.0])
obj= transpose(obj)
corners= matrix([x,y])
corners= transpose(corners)
result= dot(corners,obj)
print ('Value of Objective Function at Each Corner Point:\n', result)
```

```
Value of Objective Function at Each Corner Point:
 [[ 66.]
 [ 36.]
 [ 24.]
 [ 30.]]
```

Module 3 Practice 4

Exercise 1: Using the matrix methods, verify the calculations in Lial Section 4.3 Example 1. Compare your code to the answer sheet.

```
import numpy
from numpy import *
from numpy.linalg import *

print ('\nMinimization Problem')
x=[0,4,9]
y=[5,1,0]
obj= matrix([8.0,16.0])
obj= transpose(obj)
corners= matrix([x,y])
corners= transpose(corners)
result= dot(corners,obj)
print ('Value of Objective Function at Each Corner Point\n', result)
print ('\nMaximization Problem')
x=[0,2,3.2]
y=[4,3,0]
obj= matrix([9.0,10.0])
obj= transpose(obj)
corners= matrix([x,y])
corners= transpose(corners)
result= dot(corners,obj)
print ('Value of Objective Function at Each Corner Point\n', result)
```

```
Minimization Problem
Value of Objective Function at Each Corner Point
 [[ 80.]
 [ 48.]
 [ 72.]]

Maximization Problem
Value of Objective Function at Each Corner Point
 [[ 40. ]
 [ 48. ]
 [ 28.8]]
```