CHAPTER 15

15.1 (a) Define x_a = amount of product A produced, and x_b = amount of product B produced. The objective function is to maximize profit,

$$P = 45x_a + 20x_b$$

Subject to the following constraints

$$\begin{aligned} &20x_a + 5x_b \le 9500 && \{\text{raw materials}\} \\ &0.04x_a + 0.12x_b \le 40 && \{\text{production time}\} \\ &x_a + x_b \le 550 && \{\text{storage}\} \\ &x_a, x_b \ge 0 && \{\text{positivity}\} \end{aligned}$$

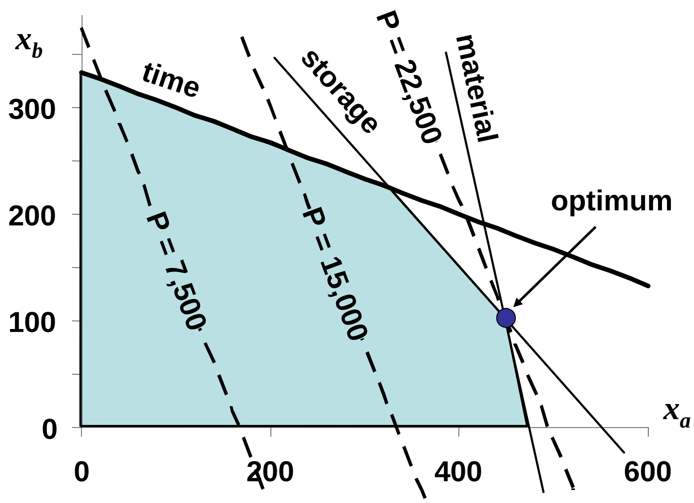
(b) To solve graphically, the constraints can be reformulated as the following straight lines

$$\begin{aligned} &x_b = 1900 - 4x_a && \{\text{raw materials}\} \\ &x_b = 333.3333 - 0.333333x_a && \{\text{production time}\} \\ &x_b = 550 - x_a && \{\text{storage}\} \end{aligned}$$

The objective function can be reformulated as

$$x_b = (1/20)P - 2.25x_a$$

The constraint lines can be plotted on the x_a-x_b plane to define the feasible space. Then the objective function line can be superimposed for various values of P until it reaches the boundary. The result is $P \cong 22{,}250$ with $x_a \cong 450$ and $x_b \cong 100$. Notice also that material and storage are the binding constraints and that there is some slack in the time constraint.



(c) The simplex tableau for the problem can be set up and solved as

Basis	P	x_a	x_b	S_1	S_2	S_3	Solution	Intercept
P	1	-45	-20	0	0	0	0	
S_1	0	20	5	1	0	0	9500	475

S_2	0	0.04	0.12	0	1	0	40	1000
S_3	0	1	1	0	0	1	550	550

Basis	***P***	x_a	x_b	S_1	S_2	S_3	**Solution**	**Intercept**
P	1	0	-8.75	2.25	0	0	21375	
x_a	0	1	0.25	0.05	0	0	475	1900
S_2	0	0	0.11	-0.002	1	0	21	190.9091
S_3	0	0	0.75	-0.05	0	1	75	100

Basis	***P***	x_a	x_b	S_1	S_2	S_3	**Solution**	**Intercept**
P	1	0	0	1.666667	0	11.66667	22250	
x_a	0	1	0	0.066667	0	-0.33333	450	
S_2	0	0	0	0.005333	1	-0.14667	10	
x_b	0	0	1	-0.06667	0	1.333333	100	

(d) An Excel spreadsheet can be set up to solve the problem as

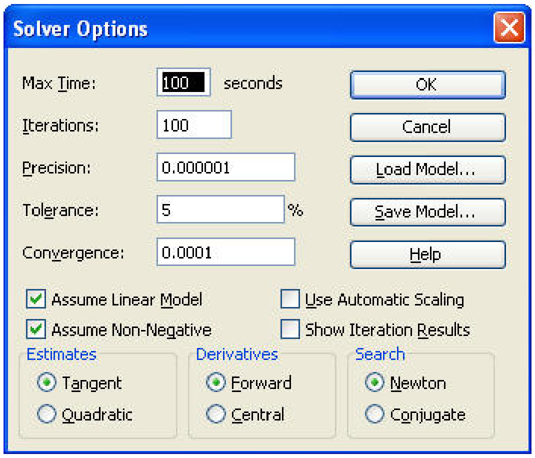
	A	B	C	D	E
1		xA	xB	total	constraint
2	amount	0	0		
3	time	0.04	0.12	0	40
4	storage	1	1	0	550
5	raw material	20	5	0	9500
6	profit	45	20	0	

The formulas in column D are

	A	B	C	D	E
1		xA	xB	total	constraint
2	amount	0	0		
3	time	0.04	0.12	=B3*B$2+C3*C$2	40
4	storage	1	1	=B4*B$2+C4*C$2	550
5	raw material	20	5	=B5*B$2+C5*C$2	9500
6	profit	45	20	=B6*B$2+C6*C$2	

The Solver can be called and set up as

Before depressing the Solve button, depress the Options button and check the boxes to “Assume Linear Model” and “Assume Non-Negative.”



The resulting solution is

	A	B	C	D	E
1		xA	xB	total	constraint
2	amount	450	100		
3	time	0.04	0.12	30	40
4	storage	1	1	550	550
5	raw material	20	5	9500	9500
6	profit	45	20	22250	

In addition, a sensitivity report can be generated as

	A	B	C	D	E	F	G	H
1	**Microsoft Excel 11.0 Sensitivity Report**							
2	**Worksheet: [prob1501.xls]Graphical**							
3	**Report Created: 6/30/2005 3:19:02 PM**							
4								
5								
6	Adjustable Cells							
7				**Final**	**Reduced**	**Objective**	**Allowable**	**Allowable**
8		**Cell**	**Name**	**Value**	**Cost**	**Coefficient**	**Increase**	**Decrease**
9		B2	amount xA	450	0	45	35	25
10		C2	amount xB	100	0	20	25	8.75
11								
12	Constraints							
13				**Final**	**Shadow**	**Constraint**	**Allowable**	**Allowable**
14		**Cell**	**Name**	**Value**	**Price**	**R.H. Side**	**Increase**	**Decrease**
15		D3	time total	30	0	40	1E+30	10
16		D4	storage total	550	11.66666667	550	68.18181818	75
17		D5	raw material total	9500	1.666666667	9500	1500	1875

(e) The high shadow price for storage from the sensitivity analysis from **(d)** suggests that increasing storage will result in the best increase in profit.

15.2 (a) The LP formulation is given by

$$\text{Maximize } Z = 150x_1 + 175x_2 + 250x_3 \qquad \{\text{Maximize profit}\}$$

subject to

$7x_1 + 11x_2 + 15x_3 \le 154$	{Material constraint}
$10x_1 + 8x_2 + 12x_3 \le 80$	{Time constraint}
$x_1 \le 9$	{"Regular" storage constraint}
$x_2 \le 6$	{"Premium" storage constraint}
$x_3 \le 5$	{"Supreme" storage constraint}
$x_1, x_2, x_3 \ge 0$	{Positivity constraints}

(b) The simplex tableau for the problem can be set up and solved as

Basis	Z	x_1	x_2	x_3	S_1	S_2	S_3	S_4	S_5	Solution	Intercept
Z	1	-150	-175	-250	0	0	0	0	0	0	
S1	0	7	11	15	1	0	0	0	0	154	10.2667
S2	0	10	8	12	0	1	0	0	0	80	6.66667
S3	0	1	0	0	0	0	1	0	0	9	∞
S4	0	0	1	0	0	0	0	1	0	6	∞
S5	0	0	0	1	0	0	0	0	1	5	5

Basis	Z	x_1	x_2	x_3	S_1	S_2	S_3	S_4	S_5	Solution	Intercept
Z	1	-150	-175	0	0	0	0	0	250	1250	
S1	0	7	11	0	1	0	0	0	-15	79	7.18182
S2	0	10	8	0	0	1	0	0	-12	20	2.5
S3	0	1	0	0	0	0	1	0	0	9	∞
S4	0	0	1	0	0	0	0	1	0	6	6
x3	0	0	0	1	0	0	0	0	1	5	∞

Basis	Z	x_1	x_2	x_3	S_1	S_2	S_3	S_4	S_5	Solution	Intercept
Z	1	68.75	0	0	0	21.88	0	0	-12.5	1687.5	
S1	0	-6.75	0	0	1	-1.375	0	0	1.5	51.5	34.3333
x2	0	1.25	1	0	0	0.125	0	0	-1.5	2.5	-1.66667
S3	0	1	0	0	0	0	1	0	0	9	∞
S4	0	-1.25	0	0	0	-0.125	0	1	1.5	3.5	2.33333
x3	0	0	0	1	0	0	0	0	1	5	5

Basis	Z	x_1	x_2	x_3	S_1	S_2	S_3	S_4	S_5	Solution
Z	1	58.3333	0	0	0	20.83	0	8.33	0	1716.7
S1	0	-5.5	0	0	1	-1.25	0	-1	0	48
x2	0	0	1	0	0	0	0	1	0	6
S3	0	1	0	0	0	0	1	0	0	9
S5	0	-0.8333	0	0	0	-0.083	0	0.67	1	2.3333
x3	0	0.83333	0	1	0	0.083	0	-0.67	0	2.6667

(c) An Excel spreadsheet can be set up to solve the problem as

	A	B	C	D	E	F
1		regular	premium	supreme	total	constraint
2	amount	0	0	0		
3	material	7	11	15	0	154
4	time	10	8	12	0	80
5	reg stor	1	0	0	0	9
6	prem stor	0	1	0	0	6
7	sup stor	0	0	1	0	5
8	profit	150	175	250	0	

The formulas in column E are

	A	B	C	D	E	F
1		regular	premium	supreme	total	constraint
2	amount	0	0	0		
3	material	7	11	15	=B3*B$2+C3*C$2+D3*D$2	154
4	time	10	8	12	=B4*B$2+C4*C$2+D4*D$2	80
5	reg stor	1	0	0	=B5*B$2+C5*C$2+D5*D$2	9
6	prem stor	0	1	0	=B6*B$2+C6*C$2+D6*D$2	6
7	sup stor	0	0	1	=B7*B$2+C7*C$2+D7*D$2	5
8	profit	150	175	250	=B8*B$2+C8*C$2+D8*D$2	

The Solver can be called and set up as

The resulting solution is

	A	B	C	D	E	F
1		regular	premium	supreme	total	constraint
2	amount	0	6	2.666667		
3	material	7	11	15	106	154
4	time	10	8	12	80	80
5	reg stor	1	0	0	0	9
6	prem stor	0	1	0	6	6
7	sup stor	0	0	1	2.666667	5
8	profit	150	175	250	1716.667	

In addition, a sensitivity report can be generated as

Microsoft Excel 11.0 Sensitivity Report
Worksheet: [Book1]Sheet1
Report Created: 6/24/2005 2:41:55 PM

Adjustable Cells

Cell	Name	Final Value	Reduced Cost	Objective Coefficient	Allowable Increase	Allowable Decrease
B2	amount regular	0	-58.33333333	150	58.33333333	1E+30
C2	amount premium	6	0	175	1E+30	8.333333334
D2	amount supreme	2.666666667	0	250	12.5	70

Constraints

Cell	Name	Final Value	Shadow Price	Constraint R.H. Side	Allowable Increase	Allowable Decrease
E3	material total	106	0	154	1E+30	48
E4	time total	80	20.83333333	80	28	32
E5	reg stor total	0	0	9	1E+30	9
E6	prem stor total	6	8.333333334	6	4	3.5
E7	sup stor total	2.666666667	0	5	1E+30	2.333333333

(d) The high shadow price for time from the sensitivity analysis from **(c)** suggests that increasing time will result in the best increase in profit.

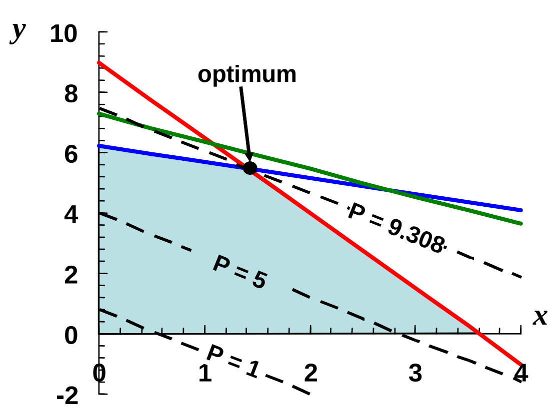
15.3 (a) To solve graphically, the constraints can be reformulated as the following straight lines

$$y = 6.22222 - 0.53333x$$
$$y = 7.2727 - 0.90909x$$
$$y = 9 - 2.5x$$

The objective function can be reformulated as

$$y = 0.8P - 1.4x$$

The constraint lines can be plotted on the *x*-*y* plane to define the feasible space. Then the objective function line can be superimposed for various values of *P* until it reaches the boundary. The result is $P \cong 9.30791$ with $x \cong 1.4$ and $y \cong 5.5$.



(b) The simplex tableau for the problem can be set up and solved as

Basis	P	x	y	S_1	S_2	S_3	Solution	Intercept
P	1	-1.75	-1.25	0	0	0	0	
S_1	0	1.2	2.25	1	0	0	14	11.66667
S_2	0	1	1.1	0	1	0	8	8
S_3	0	2.5	1	0	0	1	9	3.6

Basis	P	x	y	S_1	S_2	S_3	Solution	Intercept
P	1	0	-0.55	0	0	0.7	6.3	
S_1	0	0	1.77	1	0	-0.48	9.68	5.468927
S_2	0	0	0.7	0	1	-0.4	4.4	6.285714
x	0	1	0.4	0	0	0.4	3.6	9

Basis	P	x	y	S_1	S_2	S_3	Solution	Intercept
P	1	0	0	0.310734	0	0.550847	9.30791	
y	0	0	1	0.564972	0	-0.27119	5.468927	
S_2	0	0	0	-0.39548	1	-0.21017	0.571751	
x	0	1	0	-0.22599	0	0.508475	1.412429	

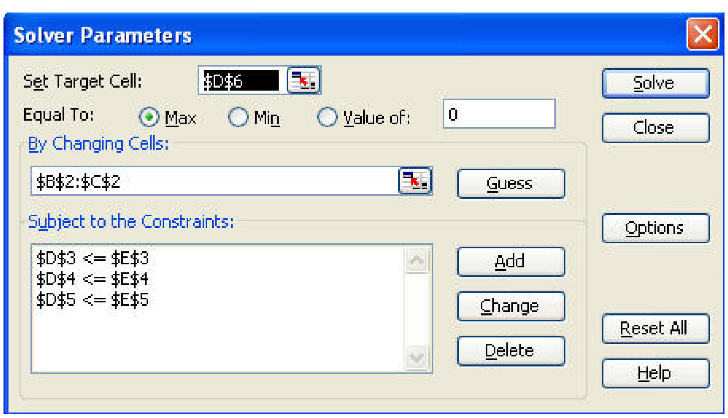
(c) An Excel spreadsheet can be set up to solve the problem as

	A	B	C	D	E
1		x	y	total	constraint
2	amount	0	0		
3	constraint 1	1.2	2.25	0	14
4	constraint 2	1	1.1	0	8
5	constraint 3	2.5	1	0	9
6	profit	1.75	1.25	0	

The formulas in column D are

	A	B	C	D	E
1		x	y	total	constraint
2	amount	0	0		
3	constraint 1	1.2	2.25	=B3*B$2+C3*C$2	14
4	constraint 2	1	1.1	=B4*B$2+C4*C$2	8
5	constraint 3	2.5	1	=B5*B$2+C5*C$2	9
6	profit	1.75	1.25	=B6*B$2+C6*C$2	

The Solver can be called and set up as



The resulting solution is

	A	B	C	D	E
1		x	y	total	constraint
2	amount	1.412429	5.468927		
3	constraint 1	1.2	2.25	14	14
4	constraint 2	1	1.1	7.428249	8
5	constraint 3	2.5	1	9	9
6	profit	1.75	1.25	9.30791	

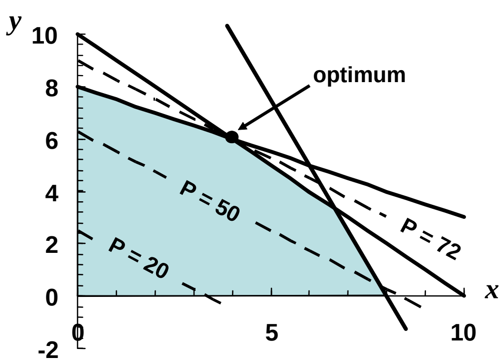
15.4 (a) To solve graphically, the constraints can be reformulated as the following straight lines

$$y = 20 - 2.5x$$
$$y = 10 - 10x$$
$$y = 9 - 2.5x$$

The objective function can be reformulated as

$$y = 0.125P - 0.75x$$

The constraint lines can be plotted on the x-y plane to define the feasible space. Then the objective function line can be superimposed for various values of P until it reaches the boundary. The result is $P \cong 72$ with $x \cong 4$ and $y \cong 6$.



(b) The simplex tableau for the problem can be set up and solved as

Basis	P	x	y	S_1	S_2	S_3	Solution	Intercept
P	1	-6	-8	0	0	0	0	
S_1	0	5	2	1	0	0	40	20
S_2	0	6	6	0	1	0	60	10
S_3	0	2	4	0	0	1	32	8

Basis	P	x	y	S_1	S_2	S_3	Solution	Intercept
P	1	-2	0	0	0	2	64	
S_1	0	4	0	1	0	-0.5	24	6
S_2	0	3	0	0	1	-1.5	12	4
y	0	0.5	1	0	0	0.25	8	16

Basis	P	x	y	S_1	S_2	S_3	Solution	Intercept
P	1	0	0	0	0.666667	1	72	
S_1	0	0	0	1	-1.33333	1.5	8	
x	0	1	0	0	0.333333	-0.5	4	
y	0	0	1	0	-0.16667	0.5	6	

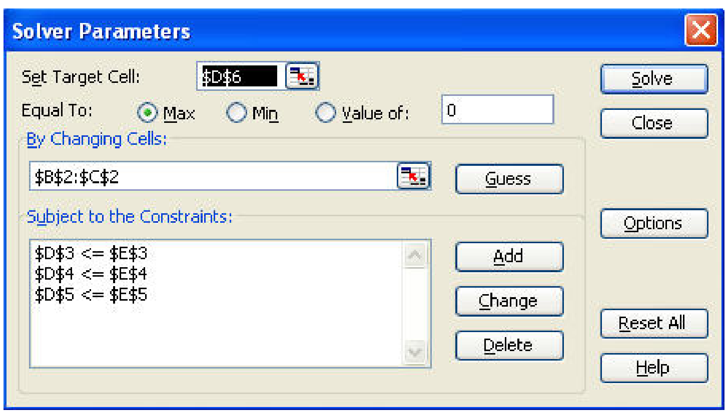
(c) An Excel spreadsheet can be set up to solve the problem as

	A	B	C	D	E
1		x	y	total	constraint
2	amount	0	0		
3	constraint 1	5	2	0	40
4	constraint 2	6	6	0	60
5	constraint 3	2	4	0	32
6	profit	6	8	0	

The formulas in column D are

	A	B	C	D	E
1		x	y	total	constraint
2	amount	0	0		
3	constraint 1	5	2	=B3*B$2+C3*C$2	40
4	constraint 2	6	6	=B4*B$2+C4*C$2	60
5	constraint 3	2	4	=B5*B$2+C5*C$2	32
6	profit	6	8	=B6*B$2+C6*C$2	

The Solver can be called and set up as



The resulting solution is

	A	B	C	D	E
1		x	y	total	constraint
2	amount	4	6		
3	constraint 1	5	2	32	40
4	constraint 2	6	6	60	60
5	constraint 3	2	4	32	32
6	profit	6	8	72	

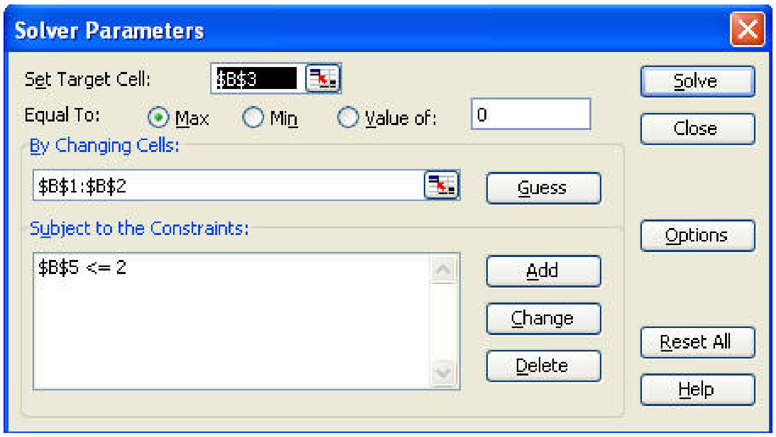
15.5 An Excel spreadsheet can be set up to solve the problem as

	A	B
1	x	0
2	y	0
3	f(x,y)	0
4	Constraint:	
5	2x+y=	0

The formulas are

	A	B
1	x	0
2	y	0
3	f(x,y)	=1.2*B1+2*B2-B2^3
4	Constraint:	
5	2x+y=	=2*B1+B2

The Solver can be called and set up as



The resulting solution is

	A	B
1	x	0.658435
2	y	0.68313
3	f(x,y)	1.837588
4	Constraint:	
5	2x+y=	2

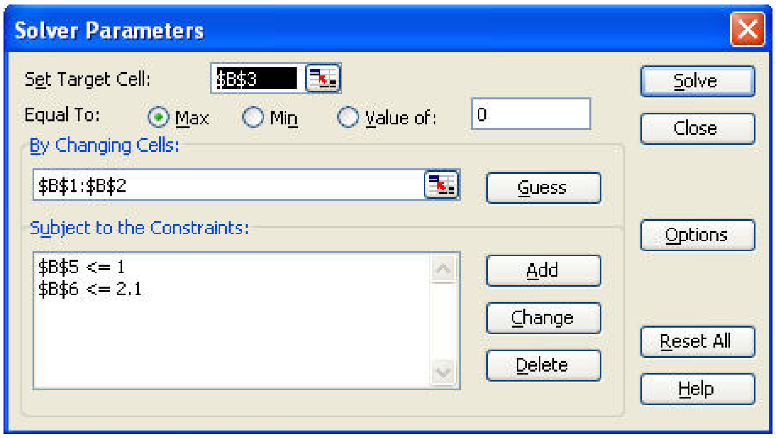
15.6 An Excel spreadsheet can be set up to solve the problem as

	A	B
1	x	0
2	y	0
3	f(x,y)	0
4	Constraints:	
5	x^2+y^2	0
6	x+2y	0

The formulas are

	A	B
1	x	0
2	y	0
3	f(x,y)	=15*B1+15*B2
4	Constraints:	
5	x^2+y^2	=B1^2+B2^2
6	x+2y	=B1+2*B2

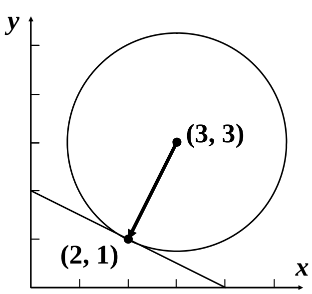
The Solver can be called and set up as



The resulting solution is

	A	B
1	x	0.727247
2	y	0.686377
3	f(x,y)	21.20435
4	Constraints:	
5	x^2+y^2	1.000001
6	x+2y	2.1

15.7 (a) The function and the constraint can be plotted and as shown indicate a solution of $x = 2$ and $y = 1$.



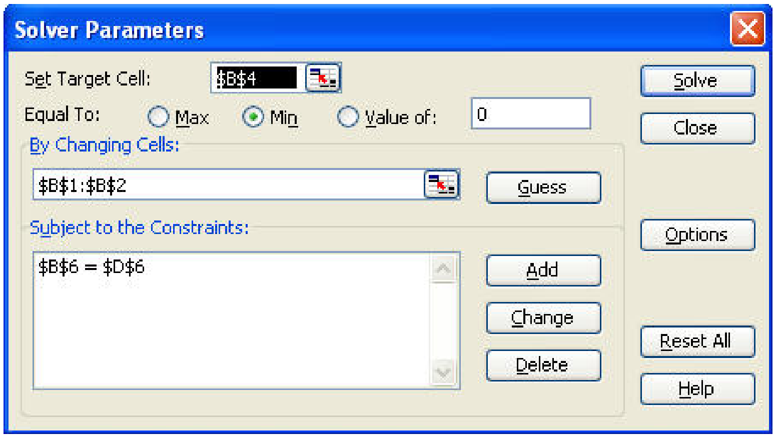
(b) An Excel spreadsheet can be set up to solve the problem as

	A	B	C	D
1	x	0		
2	y	0		
3	Minimize			
4	f(x,y)	18		
5	Subject to			
6	x+2y =	0	=	4

The formulas are

	A	B	C	D
1	x	0		
2	y	0		
3	Minimize			
4	f(x,y)	=(B1-3)^2+(B2-3)^2		
5	Subject to			
6	x+2y =	=B1+2*B2	=	4

The Solver can be called and set up as



The resulting solution is

	A	B	C	D
1	x	2		
2	y	1		
3	Minimize			
4	f(x,y)	5		
5	Subject to			
6	x+2y =	4	=	4

15.8 This problem can be solved with a variety of software tools.

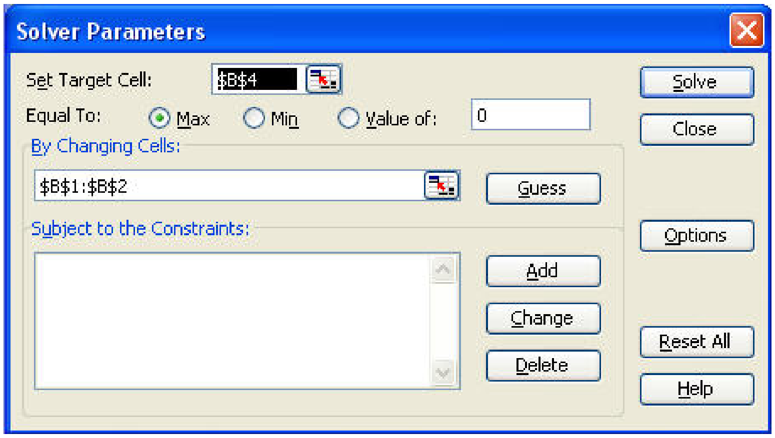
<u>Excel:</u> An Excel spreadsheet can be set up to solve the problem as

	A	B
1	x	0
2	y	0
3	Maximize	
4	f(x,y)	0

The formulas are

	A	B
1	x	0
2	y	0
3	Maximize	
4	f(x,y)	=2.25*B1*B2+1.75*B2-1.5*B1^2-2*B2^2

The Solver can be called and set up as



The resulting solution is

	A	B
1	x	0.567568
2	y	0.756757
3	Maximize	
4	f(x,y)	0.662162

MATLAB: Set up an M-file to hold the negative of the function

```
function f=fxy(x)
f = -(2.25*x(1)*x(2)+1.75*x(2)-1.5*x(1)^2-2*x(2)^2);
```

Then, the MATLAB function `fminsearch` can be used to determine the maximum:

```
>> x=fminsearch(@fxy,[0,0])
x =
    0.5676    0.7568
>> fopt=-fxy(x)
fopt =
    0.6622
```

15.9 This problem can be solved with a variety of software tools.

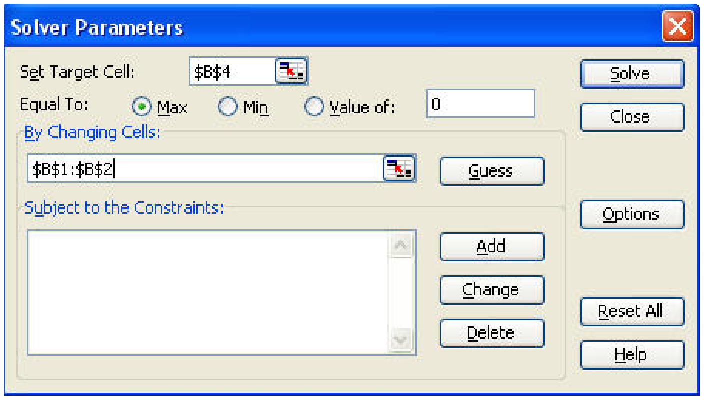
Excel: An Excel spreadsheet can be set up to solve the problem as

	A	B
1	x	0
2	y	0
3	Maximize	
4	f(x,y)	0

The formulas are

	A	B
1	x	0
2	y	0
3	Maximize	
4	f(x,y)	=4*B1+2*B2+B1^2-2*B1^4+2*B1*B2-3*B2^2

The Solver can be called and set up as



The resulting solution is

	A	B
1	x	0.96758
2	y	0.65586
3	Maximize	
4	f(x,y)	4.344006

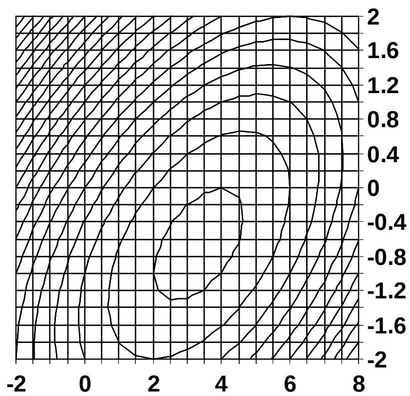
<u>MATLAB:</u> Set up an M-file to hold the negative of the function

```
function f=fxy(x)
f = -(4*x(1)+2*x(2)+x(1)^2-2*x(1)^4+2*x(1)*x(2)-3*x(2)^2);
```

Then, the MATLAB function `fminsearch` can be used to determine the maximum:

```
>> x=fminsearch(@fxy,[1,1])
x =
    0.9676    0.6559
>> fopt=-fxy(x)
fopt =
    4.3440
```

15.10 (a) This problem can be solved graphically by using a software package to generate a contour plot of the function. For example, the following plot can be developed with Excel. As can be seen, a minimum occurs at approximately $x = 3.3$ and $y = –0.7$.



(b) We can use a software package like MATLAB to determine the minimum by first setting up an M-file to hold the function as

```
function f=fxy(x)
f = -8*x(1)+x(1)^2+12*x(2)+4*x(2)^2-2*x(1)*x(2);
```

Then, the MATLAB function `fminsearch` can be used to determine the location of the minimum as:

```
>> x=fminsearch(@fxy,[0,0])
x =
    3.3333   -0.6666
```

Thus, $x = 3.3333$ and $y = -0.6666$.

(c) A software package like MATLAB can then be used to evaluate the function value at the minimum as in

```
>> fopt=fxy(x)
fopt =
  -17.3333
```

(d) We can verify that this is a minimum as follows

$$\frac{\partial^2 f}{\partial x^2} = 2 \qquad \frac{\partial^2 f}{\partial y^2} = 8 \qquad \frac{\partial}{\partial y}\left(\frac{\partial f}{\partial x}\right) = \frac{\partial}{\partial x}\left(\frac{\partial f}{\partial y}\right) = -2$$

$$H = \begin{bmatrix} 2 & -2 \\ -2 & 8 \end{bmatrix} \qquad |H| = 2 \times 8 - (-2)(-2) = 12$$

Therefore the result is a minimum because $|H| > 0$ and $\partial^2 f / \partial x^2 > 0$.

15.11 The volume of a right circular cone can be computed as

$$V = \frac{\pi r^2 h}{3}$$

where r = the radius and h = the height. The area of the cone's side is computed as

$A_s = \pi r s$

where s = the length of the side which can be computed as

$s = \sqrt{r^2 + h^2}$

The area of the circular cover is computed as

$A_c = \pi r^2$

(a) Therefore, the optimization problem with no side slope constraint can be formulated as

minimize $\quad C = 100V + 50A_s + 25A_c$

subject to $\quad V \geq 50$

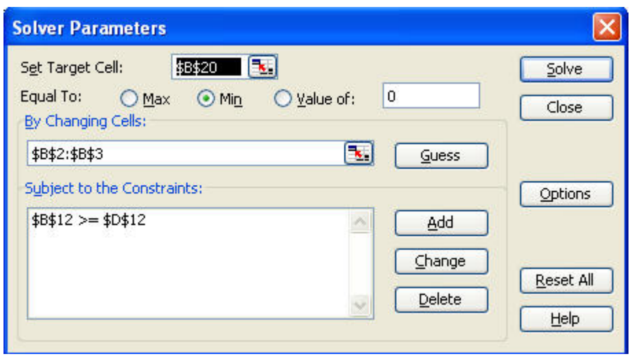
A solution can be generated in a number of different ways. For example, using Excel

	A	B	C	D	E
1	Decision variables:				
2	rad	1			
3	h	1			
4					
5	Computed values:				
6	s	1.414213562			
7	slope	0.785398163	radians	45	degrees
8	Side area	4.442882938			
9	Lid area	3.141592654			
10					
11	Constraints:				
12	Volume	1.047197551	>=	50	
13					
14					
15	Objective function:				
16	Area cost	$ 50.00			
17	Volume cost	$ 100.00			
18	Lid cost	$ 25.00			
19					
20	Total cost	$ 405.40			

The underlying formulas can be displayed as

	A	B	C	D	E
1	Decision variables:				
2	rad	1			
3	h	1			
4					
5	Computed values:				
6	s	=SQRT(B2^2+B3^2)			
7	slope	=ATAN(B3/B2)	radians	=B7*180/PI()	degrees
8	Side area	=PI()*B2*B6			
9	Lid area	=PI()*B2^2			
10					
11	Constraints:				
12	Volume	=PI()*B2^2*B3/3	>=	50	
13					
14					
15	Objective function:				
16	Area cost	50			
17	Volume cost	100			
18	Lid cost	25			
19					
20	Total cost	=B16*B8+B17*B12+B18*B9			

The Solver can be implemented as



The result is

	A	B	C	D	E
1	Decision variables:				
2	rad	2.844611637			
3	h	5.900589766			
4					
5	Computed values:				
6	s	6.550478986			
7	slope	1.121579609	radians	64.26178	degrees
8	Side area	58.5390827			
9	Lid area	25.4211877			
10					
11	Constraints:				
12	Volume	50	>=	50	
13					
14					
15	Objective function:				
16	Area cost	$ 50.00			
17	Volume cost	$ 100.00			
18	Lid cost	$ 25.00			
19					
20	Total cost	$ 8,562.48			

(b) The optimization problem with the side slope constraint can be formulated as minimize $C = 100V + 50A_s + 25A_c$

subject to

$$V \geq 50$$

$$\frac{h}{r} \leq 1$$

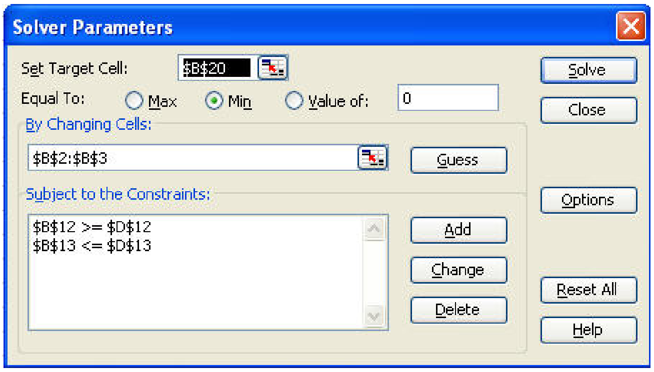
A solution can be generated in a number of different ways. For example, using Excel

	A	B	C	D	E
1	Decision variables:				
2	rad	1			
3	h	1			
4					
5	Computed values:				
6	s	1.414213562			
7	slope	0.785398163	radians	45	degrees
8	Side area	4.442882938			
9	Lid area	3.141592654			
10					
11	Constraints:				
12	Volume	1.047197551	>=	50	
13	slope	45	<=	45	
14					
15	Objective function:				
16	Area cost	$ 50.00			
17	Volume cost	$ 100.00			
18	Lid cost	$ 25.00			
19					
20	Total cost	$ 405.40			

The underlying formulas can be displayed as

	A	B	C	D	E
1	Decision variables:				
2	rad	1			
3	h	1			
4					
5	Computed values:				
6	s	=SQRT(B2^2+B3^2)			
7	slope	=ATAN(B3/B2)	radians	=B7*180/PI()	degrees
8	Side area	=PI()*B2*B6			
9	Lid area	=PI()*B2^2			
10					
11	Constraints:				
12	Volume	=PI()*B2^2*B3/3	>=	50	
13	slope	=D7	<=	45	
14					
15	Objective function:				
16	Area cost	50			
17	Volume cost	100			
18	Lid cost	25			
19					
20	Total cost	=B16*B8+B17*B12+B18*B9			

The Solver can be implemented as



The result is

	A	B	C	D	E
1	Decision variables:				
2	rad	3.627831676			
3	h	3.627831676			
4					
5	Computed values:				
6	s	5.130528758			
7	slope	0.785398163	radians	45	degrees
8	Side area	58.47350507			
9	Lid area	41.34701196			
10					
11	Constraints:				
12	Volume	49.9999999	>=	50	
13	slope	45	<=	45	
14					
15	Objective function:				
16	Area cost	$ 50.00			
17	Volume cost	$ 100.00			
18	Lid cost	$ 25.00			
19					
20	Total cost	$ 8,957.35			

15.12 Assuming that the amounts of the two-door and four-door models are x_1 and x_2, respectively, the linear programming problem can be formulated as

Maximize: $z = 13,500x_1 + 15,000x_2$

subject to

$$15x_1 + 20x_2 \le 8,000$$
$$700x_1 + 500x_2 \le 240,000$$
$$x_1 \le 400$$
$$x_2 \le 350$$
$$x_1, x_2 \ge 0$$

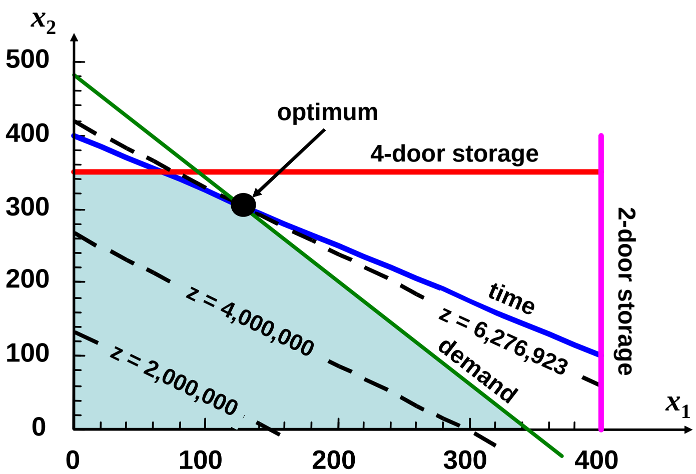
(a) To solve graphically, the constraints can be reformulated as the following straight lines

$$x_2 = 400 - 0.75x_1$$
$$x_2 = 480 - 1.4x_1$$
$$x_1 = 400$$
$$x_2 = 350$$

The objective function can be reformulated as

$$x_2 = (1/15,000)z - 0.9x_1$$

The constraint lines can be plotted on the x_1-x_2 plane to define the feasible space. Then the objective function line can be superimposed for various values of z until it reaches the boundary. The result is $z \cong$ \$6,276,923 with $x_1 \cong 123.08$ and $x_2 \cong 307.69$.



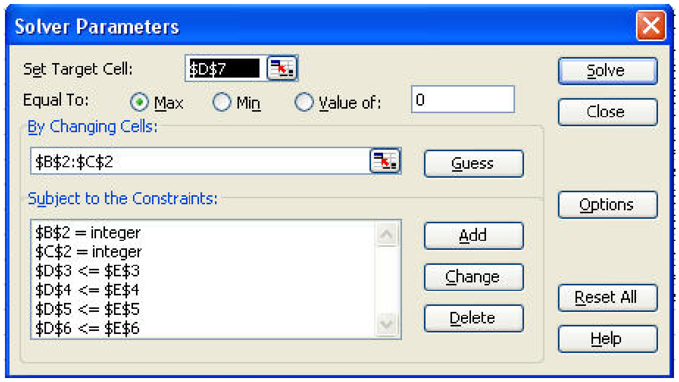
(b) The solution can be generated with Excel as in the following worksheet

	A	B	C	D	E
1		x1	x2	total	constraint
2	amount	0	0		
3	time	15	20	0	8000
4	demand	700	500	0	240000
5	Storage	1		0	400
6	Storage		1	0	350
7	profit	13500	15000	0	

The underlying formulas can be displayed as

	A	B	C	D	E
1		x1	x2	total	constraint
2	amount	122	308		
3	time	15	20	=B3*B$2+C3*C$2	8000
4	demand	700	500	=B4*B$2+C4*C$2	240000
5	Storage	1		=B5*B$2+C5*C$2	400
6	Storage		1	=B6*B$2+C6*C$2	350
7	profit	13500	15000	=B7*B$2+C7*C$2	

The Solver can be implemented as



Notice how, along with the other constraints, we have specified that the decision variables must be integers. The result of running Solver is

	A	B	C	D	E
1		x1	x2	total	constraint
2	amount	122	308		
3	time	15	20	7990	8000
4	demand	700	500	239400	240000
5	Storage	1		122	400
6	Storage		1	308	350
7	profit	13500	15000	6267000	

Thus, because we have constrained the decision variables to be integers, the maximum profit is slightly smaller than that obtained graphically in part **(a)**.

15.13 (a) First, we define the decision variables as

x_1 = number of clubs produced
x_2 = number of axes produced

The damages can be parameterized as

damage/club = 2(0.45) + 1(0.65) = 1.55 maim equivalents
damage/axe = 2(0.70) + 1(0.35) = 1.75 maim equivalents

The linear programming problem can then be formulated as

$$\text{maximize} \quad Z = 1.55x_1 + 1.75x_2$$

subject to

$$5.1x_1 + 3.2x_2 \le 240 \qquad \text{(materials)}$$
$$2.1x_1 + 4.3x_2 \le 200 \qquad \text{(time)}$$
$$x_1, x_2 \ge 0 \qquad \text{(positivity)}$$

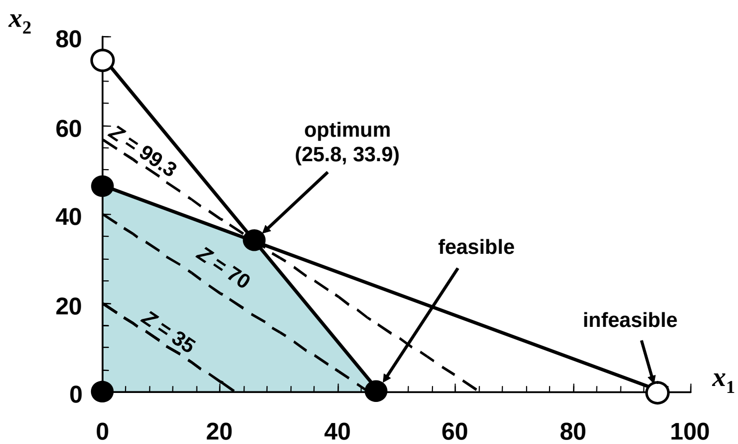
(b) and **(c)** To solve graphically, the constraints can be reformulated as the following straight lines

$$x_2 = 75 - 1.59375x_1$$
$$x_2 = 46.51163 - 0.488372x_1$$

The objective function can be reformulated as

$x_2 = (1/1.75)Z - 0.885714x_1$

The constraint lines can be plotted on the x_1-x_2 plane to define the feasible space. Then the objective function line can be superimposed for various values of Z until it reaches the boundary. The result is $Z \cong 99.3$ with $x_1 \cong 25.8$ and $x_2 \cong 33.9$.



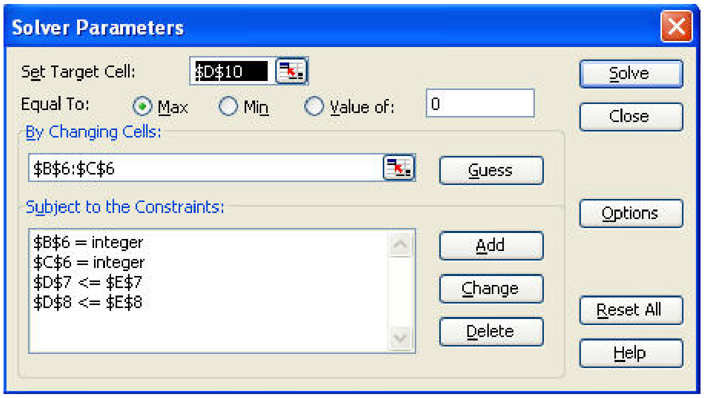
(d) The solution can be generated with Excel as in the following worksheet

	A	B	C	D	E
1		club	axe	value	
2	kills	0.45	0.7	2	
3	maims	0.65	0.35	1	
4					
5		x1	x2	total	constraint
6	quantity	0	0		
7	materials	5.1	3.2	0	240
8	time	2.1	4.3	0	200
9					
10	damage	1.55	1.75	0	

The underlying formulas can be displayed as

	A	B	C	D	E
1		club	axe	value	
2	kills	0.45	0.7	2	
3	maims	0.65	0.35	1	
4					
5		x1	x2	total	constraint
6	quantity	25	34		
7	materials	5.1	3.2	=B7*B6+C7*C6	240
8	time	2.1	4.3	=B8*B6+C8*C6	200
9					
10	damage	=D2*B2+D3*B3	=D2*C2+D3*C3	=B10*B6+C10*C6	

The Solver can be implemented as



Notice how, along with the other constraints, we have specified that the decision variables must be integers. The result of running Solver is

	A	B	C	D	E
1		club	axe	value	
2	kills	0.45	0.7	2	
3	maims	0.65	0.35	1	
4					
5		x1	x2	total	constraint
6	quantity	25	34		
7	materials	5.1	3.2	236.3	240
8	time	2.1	4.3	198.7	200
9					
10	damage	1.55	1.75	98.25	

Thus, because we have constrained the decision variables to be integers, the maximum damage is slightly smaller than that obtained graphically in part **(c)**.

15.14

```
function [x,fx,ea,iter]=goldmax(f,xl,xu,es,maxit,varargin)
% goldmax: maximization golden section search
%   [xopt,fopt,ea,iter]=goldmax(f,xl,xu,es,maxit,p1,p2,...):
%      uses golden section search to find the maximum of f
% input:
%   f = name of function
%   xl, xu = lower and upper guesses
%   es = desired relative error (default = 0.0001%)
%   maxit = maximum allowable iterations (default = 50)
%   p1,p2,... = additional parameters used by f
% output:
%   x = location of maximum
%   fx = maximum function value
%   ea = approximate relative error (%)
%   iter = number of iterations

if nargin<3,error('at least 3 input arguments required'),end
if nargin<4|isempty(es), es=0.0001;end
if nargin<5|isempty(maxit), maxit=50;end
phi=(1+sqrt(5))/2;
iter=0;
while(1)
  d = (phi-1)*(xu - xl);
  x1 = xl + d;
  x2 = xu - d;
```

```
  if f(x1,varargin{:}) > f(x2,varargin{:})
    xopt = x1;
    xl = x2;
  else
    xopt = x2;
    xu = x1;
  end
  iter=iter+1;
  if xopt~=0, ea = (2 - phi) * abs((xu - xl) / xopt) * 100;end
  if ea <= es | iter >= maxit,break,end
end
x=xopt;fx=f(xopt,varargin{:});
```

An application to solve Example 13.1 can be developed as

```
>> f=@(x) 2*sin(x) - x^2/10;
>> [xmax,fmax,ea,iter]=goldmax(f,0,4)
xmax =
    1.4276
fmax =
    1.7757
ea =
  9.3079e-005
iter =
    29
```

15.15 Each iteration of the golden-section search reduces the interval by a factor of $1/\phi$. Hence, if we start with an interval, $E_a^0 = \Delta x_0 = x_u - x_l$, the interval after the nth iteration will be

$$E_a^n = \frac{\Delta x^0}{\phi^n}$$

Each iteration of the golden-section search reduces the interval by a factor of $1/\phi$. Hence, if we start with an interval, $\Delta x^0 = x_u - x_l$, the interval after the nth iteration will be

$$\Delta x^n = \frac{\Delta x^0}{\phi^n}$$

In addition, the maximum error is $(2 - \phi)$ times the interval width. If we define the desired error as $E_{a,d}$,

$$E_{a,d} = (2-\phi)\frac{\Delta x^0}{\phi^n}$$

This equation can be solved for

$$n = \frac{\log_2\left[(2-\phi)\dfrac{\Delta x^0}{E_{a,d}}\right]}{\log_2 \phi}$$

or through further simplification,

$$n = \frac{\log_2\left(\dfrac{\Delta x^0}{E_{a,d}}\right)}{\log_2 \phi} - 2$$

The following function implements n iterations of the golden section search:

```
function [x,fx]=prob1515(f,xl,xu,Ead,varargin)
% prob1515: minimization golden section search
%   [x,fx]=prob1515(f,xl,xu,Ead,p1,p2,...):
%      uses golden section search to find the minimum of f
%      within a prescribed tolerance
% input:
%   f = name of function
%   xl, xu = lower and upper guesses
%   Ead = desired absolute error (default = 0.000001)
%   p1,p2,... = additional parameters used by f
% output:
%   x = location of minimum
%   fx = minimum function value

if nargin<3,error('at least 3 input arguments required'),end
if nargin<4|isempty(Ead), Ead=0.000001;end
phi=(1+sqrt(5))/2;
n=ceil(log2((xu-xl)/Ead)/log2(phi)-2);
if n<1,n=1;end
for i = 1:n
  d = (phi-1)*(xu - xl);
  x1 = xl + d;
  x2 = xu - d;
  if f(x1,varargin{:}) < f(x2,varargin{:})
    xopt = x1;
    xl = x2;
  else
    xopt = x2;
    xu = x1;
  end
end
x=xopt;fx=f(xopt,varargin{:});
```

Here is a session that uses this function to solve the minimization from Example 13.2.

```
>> format long
>> f=@(x) x^2/10-2*sin(x);
>> [x,fx]=prob1515(f,0,4,0.0001)
x =
   1.42752749558362
fx =
  -1.77572565250482
```

15.16

```
function [xopt,fopt]=prob1516(func,xlow,xhigh,es,maxit,varargin)
% prob1516: minimization parabolic interpolation
%   [xopt,fopt]=prob1516(func,xlow,xhigh,es,maxit,p1,p2,...):
%      uses parabolic interpolation to find the minimum of f
% input:
%   f = name of function
%   xl, xu = lower and upper guesses
%   es = desired relative error (default = 0.0001%)
```

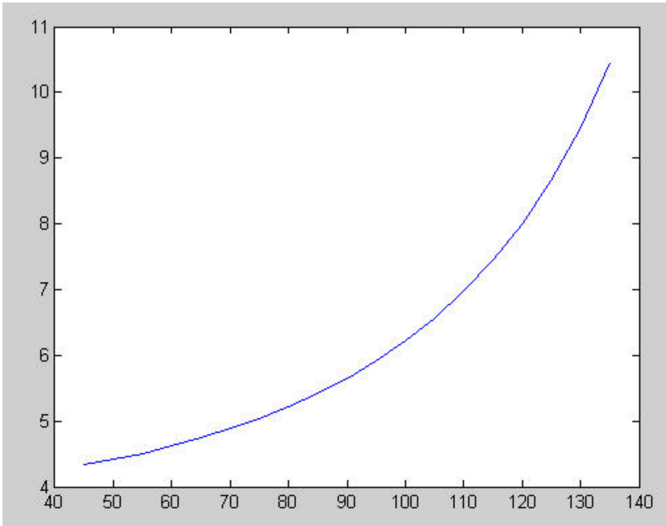
```
%   maxit = maximum allowable iterations (default = 50)
%   p1,p2,... = additional parameters used by f
% output:
%   xopt = location of minimum
%   fopt = minimum function value

if nargin<3,error('at least 3 input arguments required'),end
if nargin<4|isempty(es), es=0.0001;end
if nargin<5|isempty(maxit), maxit=50;end
iter = 0;
x1 = xlow;
x3 = xhigh;
x2 = (x1 + x3) / 2;
f1 = func(x1,varargin{:});
f2 = func(x2,varargin{:});
f3 = func(x3,varargin{:});
if f2<f1 & f2<f3
  xoptold = x2;
  while(1)
    xopt=x2-0.5*((x2-x1)^2*(f2-f3)-(x2-x3)^2*(f2-f1))/((x2-x1)...
                                         *(f2-f3)-(x2-x3)*(f2-f1));
    fopt = func(xopt,varargin{:});
    iter = iter + 1;
    if xopt > x2
      x1 = x2;
      f1 = f2;
      x2 = xopt;
      f2 = fopt;
    else
      x3 = x2;
      f3 = f2;
      x2 = xopt;
      f2 = fopt;
    end
    if xopt~=0,ea=abs((xopt - xoptold) / xopt) * 100;end
    xoptold = xopt;
    if ea<=es | iter>=maxit,break,end
  end
else
  error('bracket does not contain minimum')
end
```

```
>> f=@(x) x^2/10-2*sin(x);
>> [x,fx]=prob1516(f,0,4)
x =
    1.4276
fx =
   -1.7757
```

15.17 The following M-file generates the plot of L versus a range of α's:

```
function prob1517
alphad=[45:5:135];
alpha=alphad*pi/180;
for i=1:length(alpha)
  [t,Lmin(i)]=fminsearch(@(x) 2/sin(x)+2/sin(pi-alpha(i)-x),0.3);
end
plot(alphad,Lmin)
```


11
10
9
8
7
6
5
4
40
50
60
70
80
90
100
110
120
130
140