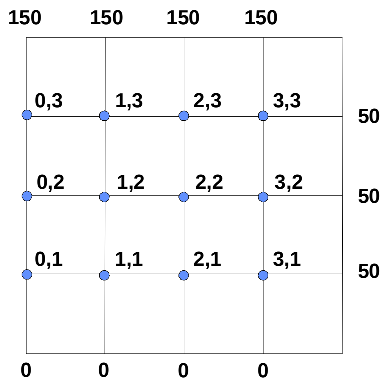
CHAPTER 29

29.1 The new representation of the plate is



Because the left edge is insulated, the finite-difference equations for the nodes on that edge are written as

$$(0,\,3)\text{:}\;\; 4T_{0,3} - 2T_{1,3} - T_{0,2} = 150$$
$$(0,\,2)\text{:}\;\; 4T_{0,2} - T_{0,3} - 2T_{1,2} - T_{0,1} = 0$$
$$(0,\,1)\text{:}\;\; 4T_{0,1} - 2T_{1,1} - T_{0,2} = 0$$

All the other nodes are represented by Eq. (29.11). The first two iterations of Liebmann's method are

150	150	150	150	
45	58.5	62.55	84.615	50
0	0	0	19.5	50
0	0	0	15	50
0	0	0	0	

150	150	150	150	
75.15	81.09	91.935	86.2509	50
13.5	21.6	32.445	51.978	50
0	0	4.5	19.2	50
0	0	0	0	

After 10 iterations, the maximum approximate error is 0.754% with the result

150	150	150	150	
109.8519	108.8637	104.6117	91.54351	50
71.75664	70.97995	68.0271	61.55182	50
35.49834	35.32963	34.98164	36.63082	50
0	0	0	0	

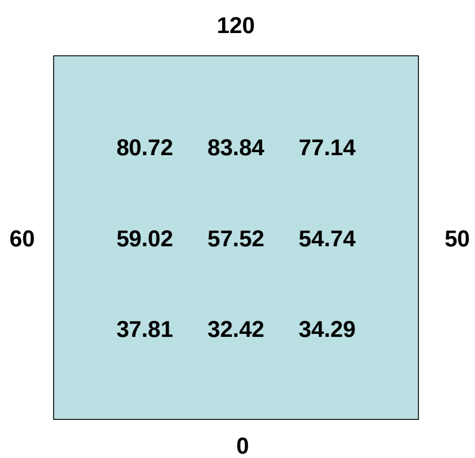
Note that the ultimate result is

150	150	150	150	
109.9655	108.9359	104.6497	91.55766	50
71.9903	71.12847	68.10531	61.58093	50
35.73874	35.48233	35.0621	36.66076	50
0	0	0	0	

29.2 Here are the results of using Liebmann's method to obtain the solution. Notice that after 6 iterations all the relative error estimates have fallen below 1% and the computation is terminated.

```
iteration = 1
       18                5.4                 16.62
       23.4              8.64                22.578
       61.02             56.898              74.8428
ea:
       100               100                 100
       100               100                 100
       100               100                 100
iteration = 2
       23.04             13.41               22.4724
       41.13             38.4768             51.222
       71.2044           79.9776             75.39132
ea:
       21.875            59.73154362         26.04261227
       43.10722101       77.54491018         55.92128382
       14.30304869       28.85758012         0.727563863
•
•
•
iteration = 6
       37.80666066       32.41632148         34.29437766
       59.01952801       57.51727394         54.73884006
       80.71921628       83.84433834         77.14262488
ea:
       0.688181746       0.130204454         0.019913358
       0.010933694       0.020874012         0.049145839
       0.033767672       0.037176168         0.02465349
```

Therefore, the results are:



29.3 The fluxes for Prob. 29.2 can be calculated as in Example 29.2. For example, for $i = j = 1$,

$$q_x = -0.49\frac{32.41632148 - 60}{2(10)} = 0.675800124$$

$$q_y = -0.49\frac{59.01952801 - 0}{2(10)} = -1.445978436$$

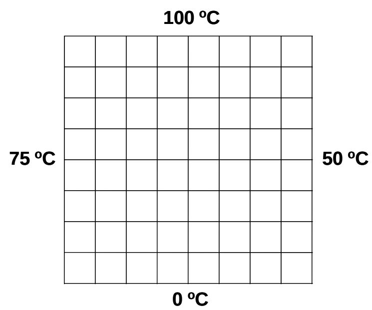
$$q_n = \sqrt{0.675800124^2 + (-1.445978436)^2} = 1.596107592$$

$$\theta = \tan^{-1}\left(\frac{-1.445978436}{0.675800124}\right) \times \frac{180°}{\pi} = -64.95024572°$$

All the results are summarized below:

```
qx:
0.675800124         0.086050934       -0.430800124
0.060826788         0.104876855       0.184173212
-0.584186289        0.087626489       0.829186289
qy:
-1.445978436        -1.409173212      -1.341101582
-1.051357613        -1.259986413      -1.049782057
-1.494021564        -1.530826788      -1.598898418
qn:
1.596107592         1.41179811        1.408595825
1.053115724         1.26434367        1.065815246
1.604173947         1.533332664       1.801118001
theta:
-64.95024572        -86.50558167      -107.8085014
-86.68881693        -85.24186843      -80.04932455
-111.356292         -86.72389108      -62.588814
```

29.4 The plate is redrawn below



After 15 iterations of the Liebmann method, the result is

	100	100	100	100	100	100	100	
75	85.32617	88.19118	88.54443	87.79909	86.06219	82.39736	73.69545	50
75	78.10995	78.88691	78.1834	76.58771	74.05069	69.82967	62.38146	50
75	73.23512	71.06672	68.71675	66.32057	63.72554	60.48986	55.99875	50
75	68.75568	63.42793	59.30269	56.25934	54.04625	52.40787	51.1222	50
75	63.33804	54.57569	48.80562	45.37425	43.79945	43.97646	46.08048	50
75	54.995	42.71618	35.95756	32.62971	31.80514	33.62176	39.22063	50
75	38.86852	25.31308	19.66293	17.3681	17.16645	19.48972	27.17735	50
	0	0	0	0	0	0	0	

with percent approximate errors of

	0	0	0	0	0	0	0	
0	0.012%	0.011%	0.007%	0.005%	0.004%	0.004%	0.003%	0
0	0.011%	0.012%	0.008%	0.005%	0.006%	0.006%	0.005%	0
0	0.024%	0.010%	0.001%	0.001%	0.004%	0.007%	0.006%	0
0	0.054%	0.016%	0.007%	0.011%	0.002%	0.007%	0.008%	0
0	0.101%	0.040%	0.008%	0.007%	0.003%	0.008%	0.011%	0

0	0.234%	0.119%	0.063%	0.033%	0.012%	0.010%	0.015%	0
0	0.712%	0.292%	0.219%	0.126%	0.030%	0.001%	0.014%	0
	0	0	0	0	0	0	0	

29.5 The solution is identical to Prob. 29.4, except that now the bottom edge must be modeled. This means that the nodes along the bottom edge are simulated with equations of the form

$$4T_{i,j} - T_{i-1,j} - T_{i+1,j} - 2T_{i,j+1} = 0$$

The resulting simulation (after 15 iterations) yields

	100	100	100	100	100	100	100	
75	86.4529	90.2627	91.2337	90.6948	88.7323	84.4436	74.8055	50
75	80.5554	83.3649	83.9691	82.8006	79.7785	74.2277	64.7742	50
75	77.4205	78.6771	78.4736	76.7481	73.3375	67.8999	60.0537	50
75	75.5117	75.4794	74.5080	72.3743	68.9073	63.9608	57.5247	50
75	74.2631	73.2996	71.7406	69.3405	65.9436	61.4870	56.0597	50
75	73.4348	71.8433	69.8853	67.3320	64.0357	59.9572	55.1934	50
75	72.8998	70.9218	68.7359	66.1171	62.9167	59.0892	54.7153	50
75	72.5345	70.4401	68.1797	65.5685	62.4467	58.7477	54.5354	50

with percent approximate errors of

	0	0	0	0	0	0	0	
0	0.009%	0.018%	0.024%	0.024%	0.018%	0.011%	0.004%	0
0	0.042%	0.066%	0.074%	0.069%	0.053%	0.034%	0.015%	0
0	0.079%	0.140%	0.155%	0.140%	0.110%	0.071%	0.032%	0
0	0.113%	0.224%	0.260%	0.239%	0.187%	0.124%	0.057%	0
0	0.133%	0.327%	0.388%	0.359%	0.284%	0.190%	0.089%	0
0	0.173%	0.471%	0.544%	0.502%	0.395%	0.261%	0.124%	0
0	0.289%	0.628%	0.709%	0.651%	0.508%	0.330%	0.153%	0
0	0.220%	0.665%	0.779%	0.756%	0.620%	0.407%	0.180%	0

29.6 The solution is identical to Examples 29.1 and 29.3, except that now heat balances must be developed for the three interior nodes on the bottom edge. For example, using the control-volume approach, node 1,0 can be modeled as

$$-0.49(5)\frac{T_{10} - T_{00}}{10} + 0.49(5)\frac{T_{20} - T_{10}}{10} + 0.49(10)\frac{T_{11} - T_{10}}{10} - 2(10) = 0$$

$$4T_{10} - T_{00} - T_{20} - 2T_{11} = -81.6327$$

Using Liebmann's method and iterating to a high level of precision, the results are

	100	100	100	
75	**79.9046**	**77.7153**	**70.7550**	50
75	**66.9027**	**60.2010**	**55.3045**	50
75	**52.5044**	**40.8810**	**40.2619**	50
75	**27.2332**	**10.5561**	**14.8618**	50

The fluxes for the computed nodes can be determined as

q_x:

	-0.0665	0.2242	0.6790	
	0.3626	0.2842	0.2499	

	0.8359	0.2999	-0.2234	
	1.5789	0.3031	-0.9664	

q_y:

	-0.8109	-0.9751	-1.0950	
	-0.6713	-0.9024	-0.7471	
	-0.9719	-1.2163	-0.9908	
	-1.2383	-1.4859	-1.2446	

q_n:

	0.8136	1.0005	1.2885	
	0.7630	0.9461	0.7878	
	1.2819	1.2527	1.0157	
	2.0066	1.5165	1.5757	

θ (degrees):

	-94.6893	-77.0532	-58.1978	
	-61.6256	-72.5225	-71.5041	
	-49.3013	-76.1474	-102.7074	
	-38.1063	-78.4711	-127.8281	

29.7 The solution is identical to Example 29.4, except that now heat balances must be developed for the interior nodes at the lower left and the upper right edges. The balances for nodes 1,1 and 3,3 can be written as

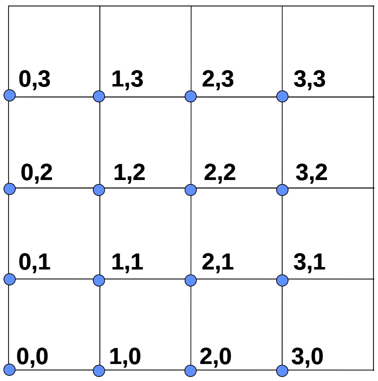
$$-4T_{11} + 0.8453T_{21} + 0.8453T_{12} = -1.154701(T_{01} + T_{10})$$

$$-4T_{33} + 0.8453T_{32} + 0.8453T_{23} = -1.154701(T_{34} + T_{43})$$

Using the appropriate boundary conditions, simple Laplacians can be used for the remaining interior nodes. The resulting simulation yields

75	100	100	100	
50	**75**	**86.02317**	**94.09269**	100
50	**63.97683**	**75**	**86.02317**	100
50	**55.90731**	**63.97683**	**75**	100
	50	50	50	75

29.8 The nodes to be simulated are



Simple Laplacians are used for all interior nodes. Balances for the edges must take insulation into account. For example, node 1,0 is modeled as

$$4T_{1,0} - T_{0,0} - T_{2,0} - 2T_{1,1} = 0$$

The corner node, 0,0 would be modeled as

$$4T_{0,0} - 2T_{1,0} - 2T_{0,1} = 0$$

The resulting set of equations can be solved for

0	25	50	75	100
23.89706	**32.16912**	**45.58824**	**60.29412**	75
31.25	**34.19118**	**39.88971**	**45.58824**	50
32.72059	**33.45588**	**34.19118**	**32.16912**	25
32.72059	**32.72059**	**31.25**	**23.89706**	0

The fluxes can be computed as

J

J_x

-1.225	-1.225	-1.225	-1.225	-1.225
-0.40533	-0.53143	-0.68906	-0.72059	-0.72059
-0.14412	-0.21167	-0.27923	-0.2477	-0.21618
-0.03603	-0.03603	0.031526	0.225184	0.351287
0	0.036029	0.216176	0.765625	1.170956

J_y

1.170956	0.351287	-0.21618	-0.72059	-1.225
0.765625	0.225184	-0.2477	-0.72059	-1.225
0.216176	0.031526	-0.27923	-0.68906	-1.225
0.036029	-0.03603	-0.21167	-0.53143	-1.225
0	-0.03603	-0.14412	-0.40533	-1.225

J_n

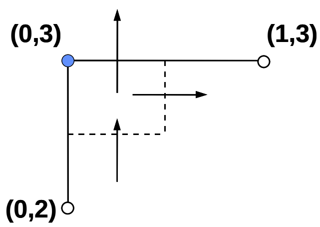
1.694628	1.274373	1.243928	1.421222	1.732412
0.866299	0.577174	0.732232	1.019066	1.421222
0.259812	0.214008	0.394888	0.732232	1.243928
0.050953	0.050953	0.214008	0.577174	1.274373
0	0.050953	0.259812	0.866299	1.694628

θ (degrees)

136.2922	163.999	-169.992	-149.534	-135
117.8973	157.0362	-160.228	-135	-120.466
123.6901	171.5289	-135	-109.772	-100.008
135	-135	-81.5289	-67.0362	-73.999
0	-45	-33.6901	-27.8973	-46.2922

29.9 Node 0,3:

There are two approaches for modeling this node. One would be to consider it a Dirichlet node and not model it at all (i.e., set its temperature at 50°C). The second alternative is to use a heat balance to model it as shown here



$$0 = 0.75(15)(0.5)\frac{T_{1,3} - T_{0,3}}{40} - 0.75(20)(0.5)\frac{T_{0,3} - T_{0,2}}{30} + 0.015(20)(0.5)(10 - T_{0,3})$$

$$-1.04046T_{1,3} + 4T_{0,3} - 1.84971T_{0,2} = 11.09827$$

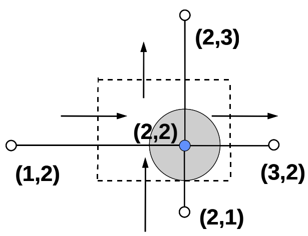
Node 2,3:

$$0 = -0.75(15)(0.5)\frac{T_{2,3} - T_{1,3}}{40} + 0.75(15)(0.5)\frac{T_{3,3} - T_{2,3}}{20} - 0.75(30)(0.5)\frac{T_{2,3} - T_{2,2}}{30}$$

$$+ 0.015(30)(0.5)(10 - T_{2,3})$$

$$4T_{2,3} - 0.55046T_{1,3} - 1.10092T_{3,3} - 1.46789T_{2,2} = 8.807339$$

Node 2,2:

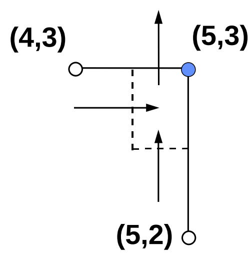


$$0 = -0.75(22.5)(0.5)\frac{T_{2,2} - T_{1,2}}{40} + 0.75(22.5)(0.5)\frac{T_{3,2} - T_{2,2}}{20} - 0.75(30)(0.5)\frac{T_{2,2} - T_{2,1}}{15}$$

$$+ 0.75(30)(0.5)\frac{T_{2,3} - T_{2,2}}{30} + 10\pi(7.5)^2$$

$$4T_{2,2} - 0.48T_{1,2} - 0.96T_{3,2} - 1.70667T_{2,1} - 0.85333T_{2,3} = 4021.239$$

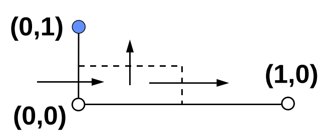
Node 5,3:



$$0 = -0.75(15)(1)\frac{T_{5,3} - T_{4,3}}{20} - 0.75(10)(1)\frac{T_{5,3} - T_{5,2}}{30} + 0.01(10)(0.5)(10 - T_{5,3})$$

$$4T_{5,3} - 2.33766T_{4,3} - 1.03896T_{5,2} = 6.23377$$

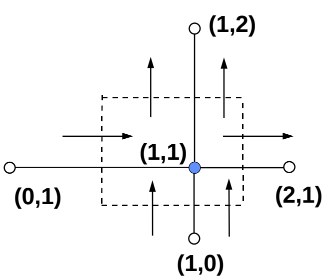
29.10 Node 0,0 :



$$0 = 0.01(7.5)(2)(20 - T_{0,0}) + 0.7(7.5)(2)\frac{T_{1,0} - T_{0,0}}{40} + 0.7(20)(2)\frac{T_{0,1} - T_{0,0}}{15}$$

$$4T_{0,0} - 0.460695T_{1,0} - 3.276051T_{0,1} = 5.265082$$

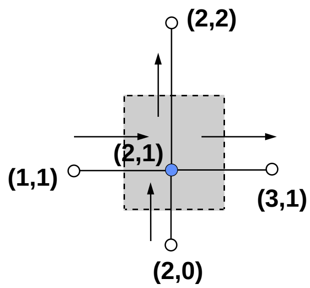
Node 1,1:



$$0 = -0.7(22.5)(1)\frac{T_{1,1} - T_{0,1}}{40} + 0.5(22.5)(2)\frac{T_{2,1} - T_{1,1}}{20} - 0.7(20)(2)\frac{T_{1,1} - T_{1,0}}{15}$$
$$- 0.5(10)(2)\frac{T_{1,1} - T_{1,0}}{15} + 0.7(20)(2)\frac{T_{1,2} - T_{1,1}}{30} + 0.5(10)(2)\frac{T_{1,2} - T_{1,1}}{30}$$

$$4T_{1,1} - 0.78775T_{2,1} - 1.77389T_{1,0} - 0.88694T_{1,2} - 0.55142T_{0,1} = 0$$

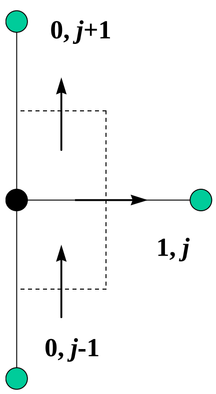
Node 2,1:



$$0 = -0.5(22.5)(2)\frac{T_{2,1}-T_{1,1}}{20} + 0.5(22.5)(2)\frac{T_{3,1}-T_{2,1}}{20} - 0.5(20)(2)\frac{T_{2,1}-T_{2,0}}{15} + 0.5(20)(2)\frac{T_{2,2}-T_{2,1}}{30} + 10(22.5)(20)$$

$$4T_{2,1} - 1.05882T_{1,1} - 1.05882T_{3,1} - 1.2549T_{2,0} - 0.62745T_{2,2} = 4{,}235.294$$

29.11 The control volume is drawn as in



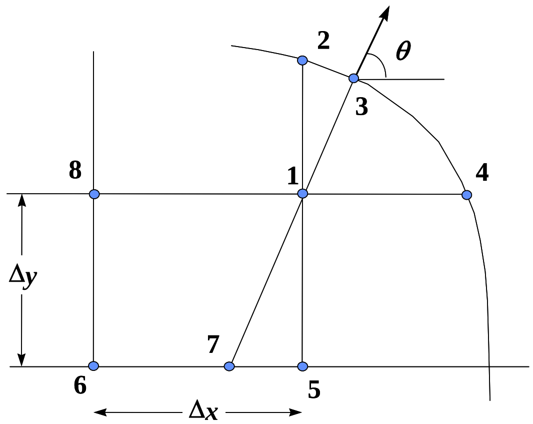
A flux balance around the node can be written as (note $\Delta x = \Delta y = h$)

$$-kh\Delta z\frac{T_{1,j}-T_{0,j}}{h} + k(h/2)\Delta z\frac{T_{0,j}-T_{0,j-1}}{h} - k(h/2)\Delta z\frac{T_{0,j+1}-T_{0,j}}{h} = 0$$

Collecting and canceling terms gives

$$4T_{0,j} - T_{0,j-1} - T_{0,j+1} - 2T_{1,j} = 0$$

29.12 A setup similar to Fig. 29.11, but with $\theta > 45^{\circ}$ can be drawn as in



The normal derivative at node 3 can be approximated by the gradient between nodes 1 and 7,

$$\left.\frac{\partial T}{\partial \eta}\right|_3 = \frac{T_1 - T_7}{L_{17}}$$

When θ is greater than 45^o as shown, the distance from node 5 to 7 is $\Delta y \cot\theta$, and linear interpolation can be used to estimate

$$T_7 = T_5 + (T_6 - T_5)\frac{\Delta y \cot\theta}{\Delta x}$$

The length L_{17} is equal to $\Delta y/\sin\theta$. This length, along with the approximation for T_7 can be substituted into the gradient equation to give

$$T_1 = \left(\frac{\Delta y}{\sin\theta}\right)\left.\frac{\partial T}{\partial \eta}\right|_3 + T_6\frac{\Delta y \cot\theta}{\Delta x} + T_5\left(1 - \frac{\Delta y \cot\theta}{\Delta x}\right)$$

29.13 The following VBA program implements Liebmann’s method with relaxation.

```
Option Explicit
Sub Liebmann()
Dim nx As Integer, ny As Integer, l As Integer
Dim i As Integer, j As Integer
Dim T(20, 20) As Double, ea(20, 20) As Double, Told(20, 20) As Double
Dim qy(20, 20) As Double, qx(20, 20) As Double, qn(20, 20) As Double
Dim th(20, 20) As Double
Dim Trit As Double, Tlef As Double, Ttop As Double, Tbot As Double
Dim lam As Double, emax As Double, es As Double
Dim pi As Double
Dim k As Double, x As Double, y As Double, dx As Double, dy As Double
nx = 4
ny = 4
pi = 4 * Atn(1)
x = 40
y = 40
k = 0.49
lam = 1.5
es = 1
dx = x / nx
dy = y / ny
```

```
Tbot = 0
Tlef = 75
Trit = 50
Ttop = 100
For i = 1 To nx - 1
  T(i, 0) = Tbot
Next i
For i = 1 To nx - 1
  T(i, ny) = Ttop
Next i
For j = 1 To ny - 1
  T(0, j) = Tlef
Next j
For j = 1 To ny - 1
  T(nx, j) = Trit
Next j
l = 0
Sheets("sheet1").Select
Range("a5:z5000").ClearContents
Range("a5").Select
Do
  l = l + 1
  emax = 0
  For j = 1 To ny - 1
    For i = 1 To nx - 1
      Told(i, j) = T(i, j)
      T(i, j) = (T(i + 1, j) + T(i - 1, j) + T(i, j + 1) + T(i, j - 1)) / 4
      T(i, j) = lam * T(i, j) + (1 - lam) * Told(i, j)
      ea(i, j) = Abs((T(i, j) - Told(i, j)) / T(i, j)) * 100
      If (ea(i, j) > emax) Then emax = ea(i, j)
    Next i
  Next j
  ActiveCell.Value = "iteration = "
  ActiveCell.Offset(0, 1).Select
  ActiveCell.Value = l
  ActiveCell.Offset(1, -1).Select
  For j = 1 To ny - 1
    For i = 1 To nx - 1
      ActiveCell.Value = T(i, j)
      ActiveCell.Offset(0, 1).Select
    Next i
    ActiveCell.Offset(1, -(nx - 1)).Select
  Next j
  ActiveCell.Value = "ea = "
  ActiveCell.Offset(1, 0).Select
  For j = 1 To ny - 1
    For i = 1 To nx - 1
      ActiveCell.Value = ea(i, j)
      ActiveCell.Offset(0, 1).Select
    Next i
    ActiveCell.Offset(1, -(nx - 1)).Select
  Next j
  If emax <= es Then Exit Do
Loop
For j = 1 To ny - 1
  For i = 1 To nx - 1
    qy(i, j) = -k * (T(i, j + 1) - T(i, j - 1)) / 2 / dy
    qx(i, j) = -k * (T(i + 1, j) - T(i - 1, j)) / 2 / dx
    qn(i, j) = Sqr(qy(i, j) ^ 2 + qx(i, j) ^ 2)
    th(i, j) = Application.WorksheetFunction.Atan2(qx(i, j),qy(i, j))*180/pi
  Next i
Next j
```

```
ActiveCell.Offset(1, 0).Select
ActiveCell.Value = "qx = "
ActiveCell.Offset(1, 0).Select
For j = 1 To ny - 1
  For i = 1 To nx - 1
    ActiveCell.Value = qx(i, j)
    ActiveCell.Offset(0, 1).Select
  Next i
  ActiveCell.Offset(1, -(nx - 1)).Select
Next j
ActiveCell.Offset(1, 0).Select
ActiveCell.Value = "qy = "
ActiveCell.Offset(1, 0).Select
For j = 1 To ny - 1
  For i = 1 To nx - 1
    ActiveCell.Value = qy(i, j)
    ActiveCell.Offset(0, 1).Select
  Next i
  ActiveCell.Offset(1, -(nx - 1)).Select
Next j
ActiveCell.Offset(1, 0).Select
ActiveCell.Value = "qn = "
ActiveCell.Offset(1, 0).Select
For j = 1 To ny - 1
  For i = 1 To nx - 1
    ActiveCell.Value = qn(i, j)
    ActiveCell.Offset(0, 1).Select
  Next i
  ActiveCell.Offset(1, -(nx - 1)).Select
Next j
ActiveCell.Offset(1, 0).Select
ActiveCell.Value = "theta = "
ActiveCell.Offset(1, 0).Select
For j = 1 To ny - 1
  For i = 1 To nx - 1
    ActiveCell.Value = th(i, j)
    ActiveCell.Offset(0, 1).Select
  Next i
  ActiveCell.Offset(1, -(nx - 1)).Select
Next j
End Sub
```

When the program is run, the result of the last iteration is shown below. Note that for simplicity in programming, the sense of the *y* dimension is reversed from Fig. 29.3. That is, node 1,1 is at the upper left rather than the lower left corner of the domain.

iteration =		9	
	43.0006	33.29754	33.88506
	63.21151	56.11237	52.33998
	78.58717	76.06401	69.71051
ea =			
	0.711643	0.342925	0.247647
	0.045667	0.464174	0.027927
	0.194747	0.17208	0.47127
qx =			
	1.02171	0.223331	-0.40921
	0.462747	0.266352	0.149753
	-0.02607	0.217478	0.638568
qy =			

-1.54868	-1.37475	-1.28233
-0.87187	-1.04778	-0.87772
-0.90132	-1.07525	-1.16767

qn =

1.855346	1.392775	1.346039
0.987063	1.081103	0.890407
0.901695	1.09702	1.330873

theta =

-56.586	-80.7728	-107.699
-62.0428	-75.7371	-80.3177
-91.6567	-78.5657	-61.3269

29.14 When the program is run, the result of the last iteration is:

iteration = 6

37.80666	32.41632	34.29438
59.01953	57.51727	54.73884
80.71922	83.84434	77.14262

ea =

0.688182	0.130204	0.019913
0.010934	0.020874	0.049146
0.033768	0.037176	0.024653

qx =

0.6758	0.086051	-0.4308
0.060827	0.104877	0.184173
-0.58419	0.087626	0.829186

qy =

-1.44598	-1.40917	-1.3411
-1.05136	-1.25999	-1.04978
-1.49402	-1.53083	-1.5989

qn =

1.596108	1.411798	1.408596
1.053116	1.264344	1.065815
1.604174	1.533333	1.801118

theta =

-64.9502	-86.5056	-107.809
-86.6888	-85.2419	-80.0493
-111.356	-86.7239	-62.5888

29.15 When the program is run, the result of the last iteration is:

iteration = 15

38.86852	25.31308	19.66293	17.3681	17.16645	19.48972	27.17735
54.995	42.71618	35.95756	32.62971	31.80514	33.62176	39.22063
63.33804	54.57569	48.80562	45.37425	43.79945	43.97646	46.08048
68.75568	63.42793	59.30269	56.25934	54.04625	52.40787	51.1222
73.23512	71.06672	68.71675	66.32057	63.72554	60.48986	55.99875
78.10995	78.88691	78.1834	76.58771	74.05069	69.82967	62.38146
85.32617	88.19118	88.54443	87.79909	86.06219	82.39736	73.69545

ea =

0.711857	0.292484	0.219373	0.125895	0.030211	0.001306	0.013573
0.234333	0.119163	0.06299	0.032735	0.01195	0.009868	0.014938
0.100506	0.040025	0.007652	0.007407	0.003258	0.007935	0.010582

0.054168	0.016381	0.006687	0.010552	0.002001	0.006646	0.007611
0.024103	0.009696	0.000684	0.000661	0.003571	0.006687	0.006189
0.010882	0.012035	0.007912	0.00547	0.005693	0.006169	0.005055
0.011993	0.011445	0.007054	0.004884	0.004262	0.003984	0.003025

qx =

2.434659	0.941074	0.389304	0.122327	-0.10396	-0.49053	-1.495
1.581907	0.932835	0.494237	0.203469	-0.04861	-0.36336	-0.80253
1.000791	0.712088	0.450871	0.245303	0.068492	-0.11177	-0.29515
0.567032	0.463196	0.351261	0.257566	0.188722	0.143278	0.117986
0.192731	0.2214	0.232561	0.24457	0.285705	0.378613	0.514003
-0.19046	-0.0036	0.112661	0.202503	0.331144	0.571792	0.971654
-0.64637	-0.15769	0.019212	0.12163	0.264685	0.60597	1.587471

qy =

-2.69476	-2.09309	-1.76192	-1.59886	-1.55845	-1.64747	-1.92181
-1.19901	-1.43387	-1.42799	-1.3723	-1.30502	-1.19985	-0.92625
-0.67427	-1.01488	-1.14391	-1.15785	-1.08981	-0.92052	-0.58318
-0.48496	-0.80806	-0.97565	-1.02637	-0.97638	-0.80916	-0.486
-0.45836	-0.75749	-0.92515	-0.99609	-0.98022	-0.85367	-0.5517
-0.59246	-0.8391	-0.97156	-1.05245	-1.0945	-1.07347	-0.86714
-1.07261	-1.03454	-1.06901	-1.1472	-1.27152	-1.47835	-1.84331

qn =

3.631704	2.29492	1.804417	1.603529	1.561915	1.718944	2.434829
1.984955	1.710602	1.511103	1.387304	1.305922	1.253663	1.225563
1.206742	1.239775	1.22956	1.183552	1.091965	0.92728	0.653614
0.746129	0.931403	1.036951	1.058194	0.99445	0.821744	0.500112
0.497231	0.789183	0.953937	1.025675	1.021006	0.933861	0.75404
0.622322	0.839106	0.978066	1.071752	1.143494	1.216256	1.302321
1.252313	1.046491	1.069186	1.153632	1.298773	1.597719	2.432663

theta =

-47.9028	-65.7909	-77.5404	-85.6249	-93.8164	-106.581	-127.88
-37.1603	-56.9531	-70.9089	-81.5663	-92.1332	-106.848	-130.907
-33.9697	-54.9446	-68.4882	-78.0382	-86.4039	-96.923	-116.845
-40.5389	-60.1778	-70.1997	-75.9126	-79.0603	-79.9587	-76.3542
-67.1942	-73.7074	-75.8896	-76.2051	-73.7501	-66.0821	-47.026
-107.821	-90.2458	-83.3856	-79.1087	-73.1666	-61.9576	-41.7469
-121.074	-98.6669	-88.9704	-83.9479	-78.241	-67.7114	-49.2647

29.16

$$\Sigma q = 0$$

$$q_{left} - q_{right} + q_{lower-A} + q_{lower-B} - q_{upper-A} - q_{upper-B} + q_{source} = 0$$

$$k_A h\Delta z \frac{T_{21}-T_{22}}{h} - k_B h\Delta z \frac{T_{22}-T_{23}}{h/2} + k_A \frac{h}{2}\Delta z \frac{T_{12}-T_{22}}{h} + k_B \frac{h}{4}\Delta z \frac{T_{12}-T_{22}}{h}$$

$$- k_A \frac{h}{2}\Delta z \frac{T_{22}-T_{32}}{h} - k_B \frac{h}{4}\Delta z \frac{T_{22}-T_{32}}{h} + S(h)\left(\frac{3}{4}h \cdot h\right)\Delta z = 0$$

$$0.25(10)(0.25)\frac{T_{21}-T_{22}}{10} - 0.45(10)(0.25)\frac{T_{22}-T_{23}}{5} + 0.25(5)(0.25)\frac{T_{12}-T_{22}}{10}$$

$$+ 0.45(2.5)(0.25)\frac{T_{12}-T_{22}}{10} - 0.25(5)(0.25)\frac{T_{22}-T_{32}}{10}$$

$$- 0.45(2.5)(0.25)\frac{T_{22}-T_{32}}{10} + 6(10)(7.5)(0.25) = 0$$

$$0.0625(T_{21}-T_{22})-0.225(T_{22}-T_{23})+0.03125(T_{12}-T_{22})+0.028125(T_{12}-T_{22})$$
$$-0.03125(T_{22}-T_{32})-0.028125(T_{22}-T_{32})+112.5=0$$
$$0.0625T_{21}-(0.0625+0.225+0.03125+0.028125+0.03125+0.028125)T_{22}+0.225T_{23}$$
$$+(0.03125+0.028125)T_{12}+(0.03125+0.028125)T_{32}=-112.5$$
$$-0.40625T_{22}+0.0625T_{21}+0.225T_{23}+0.059375T_{12}+0.059375T_{32}=-112.5$$

This equation can be multiplied by –4/0.40625 in order that the coefficient of the T_{22} term is 4. The result is

$$4T_{22}-0.61538T_{21}-2.21538T_{23}-0.58462T_{12}-0.58462T_{32}=1107.692$$

29.17

Horizontal flux:

$$q_{xA}=-0.25\frac{51.6-74.2}{10}=0.565 \qquad \text{from } A \text{ into } B$$
$$q_{xB}=-0.45\frac{45.3-51.6}{5}=0.567 \qquad \text{from } A \text{ into } B$$

The horizontal flux at the boundary should be equal.

Vertical flux:

$$q_{yA}=-0.25\frac{38.6-87.4}{2(10)}=0.610 \qquad \text{upward}$$
$$q_{yB}=-0.45\frac{38.6-87.4}{2(10)}=1.098 \qquad \text{upward}$$

The 2 vertical fluxes are unequal.

29.18

120	**90.05528**	**74.61385**	**66.58249**	**63.96664**				
100	**82.80365**	**70.9088**	**63.87474**	**61.35079**				
80	**70.2505**	**62.34297**	**56.65689**	**53.68704**				
60	**55.85537**	**51.5557**	**46.72278**	**40.08361**	**26.69476**	**16.97098**	**8.328404**	**0**
40	**41.61528**	**41.30169**	**38.59493**	**33.22986**	**24.86222**	**16.43038**	**8.171316**	**0**
20	**29.30405**	**33.44084**	**33.12539**	**29.37867**	**23.09389**	**15.71702**	**7.926474**	**0**
0	**22.16008**	**30.03222**	**31.08714**	**28.06553**	**22.41766**	**15.41731**	**7.817566**	**0**

29.19 Substituting centered finite-divided differences for the second derivatives gives

$$\frac{T_{i+1,j,k}-2T_{i,j,k}+T_{i-1,j,k}}{\Delta x^2}+\frac{T_{i,j+1,k}-2T_{i,j,k}+T_{i,j-1,k}}{\Delta y^2}+\frac{T_{i,j,k+1}-2T_{i,j,k}+T_{i,j,k-1}}{\Delta z^2}=f(x,y,z)$$

Consolidating terms gives

$$T_{i+1,j,k}+T_{i-1,j,k}+T_{i,j+1,k}+T_{i,j-1,k}+T_{i,j,k+1}+T_{i,j,k-1}-6T_{i,j,k}=f(x,y,z)\Delta x^2$$

This can be solved for

$$T_{i,j,k} = \frac{T_{i+1,j,k} + T_{i-1,j,k} + T_{i,j+1,k} + T_{i,j-1,k} + T_{i,j,k+1} + T_{i,j,k-1} - f(x,y,z)\Delta x^2}{6}$$

or substituting $f = -10$ and $\Delta x = \Delta y = \Delta z = 1/6$ gives,

$$T_{i,j,k} = \frac{T_{i+1,j,k} + T_{i-1,j,k} + T_{i,j+1,k} + T_{i,j-1,k} + T_{i,j,k+1} + T_{i,j,k-1} + 0.277778}{6}$$

The following table of values summarizes the resulting solution:

z = 0	x = 0	x = 0.1667	x = 0.3333	x = 0.5	x = 0.6667	x = 0.8333	x = 1	z = 0
y = 1	0	0	0	0	0	0	0	y = 1
y = 0.8333	0	0	0	0	0	0	0	y = 0.8333
y = 0.6667	0	0	0	0	0	0	0	y = 0.6667
y = 0.5	0	0	0	0	0	0	0	y = 0.5
y = 0.3333	0	0	0	0	0	0	0	y = 0.3333
y = 0.1667	0	0	0	0	0	0	0	y = 0.1667
y = 0	0	0	0	0	0	0	0	y = 0
z = 0.1667	x = 0	x = 0.1667	x = 0.3333	x = 0.5	x = 0.6667	x = 0.8333	x = 1	z = 0.1667
y = 1	0	0	0	0	0	0	0	y = 1
y = 0.8333	0	0.1474	0.2023	0.2166	0.2023	0.1474	0	y = 0.8333
y = 0.6667	0	0.2023	0.2858	0.3088	0.2858	0.2023	0	y = 0.6667
y = 0.5	0	0.2166	0.3088	0.3344	0.3088	0.2166	0	y = 0.5
y = 0.3333	0	0.2023	0.2858	0.3088	0.2858	0.2023	0	y = 0.3333
y = 0.1667	0	0.1474	0.2023	0.2166	0.2023	0.1474	0	y = 0.1667
y = 0	0	0	0	0	0	0	0	y = 0
z = 0.3333	x = 0	x = 0.1667	x = 0.3333	x = 0.5	x = 0.6667	x = 0.8333	x = 1	z = 0.3333
y = 1	0	0	0	0	0	0	0	y = 1
y = 0.8333	0	0.2023	0.2858	0.3088	0.2858	0.2023	0	y = 0.8333
y = 0.6667	0	0.2858	0.4153	0.4521	0.4153	0.2858	0	y = 0.6667
y = 0.5	0	0.3088	0.4521	0.4934	0.4521	0.3088	0	y = 0.5
y = 0.3333	0	0.2858	0.4153	0.4521	0.4153	0.2858	0	y = 0.3333
y = 0.1667	0	0.2023	0.2858	0.3088	0.2858	0.2023	0	y = 0.1667
y = 0	0	0	0	0	0	0	0	y = 0
z = 0.5	x = 0	x = 0.1667	x = 0.3333	x = 0.5	x = 0.6667	x = 0.8333	x = 1	z = 0.5
y = 1	0	0	0	0	0	0	0	y = 1
y = 0.8333	0	0.2166	0.3088	0.3344	0.3088	0.2166	0	y = 0.8333
y = 0.6667	0	0.3088	0.4521	0.4934	0.4521	0.3088	0	y = 0.6667
y = 0.5	0	0.3344	0.4934	0.5397	0.4934	0.3344	0	y = 0.5
y = 0.3333	0	0.3088	0.4521	0.4934	0.4521	0.3088	0	y = 0.3333
y = 0.1667	0	0.2166	0.3088	0.3344	0.3088	0.2166	0	y = 0.1667
y = 0	0	0	0	0	0	0	0	y = 0
z = 0.6667	x = 0	x = 0.1667	x = 0.3333	x = 0.5	x = 0.6667	x = 0.8333	x = 1	z = 0.6667
y = 1	0	0	0	0	0	0	0	y = 1
y = 0.8333	0	0.2023	0.2858	0.3088	0.2858	0.2023	0	y = 0.8333
y = 0.6667	0	0.2858	0.4153	0.4521	0.4153	0.2858	0	y = 0.6667
y = 0.5	0	0.3088	0.4521	0.4934	0.4521	0.3088	0	y = 0.5
y = 0.3333	0	0.2858	0.4153	0.4521	0.4153	0.2858	0	y = 0.3333
y = 0.1667	0	0.2023	0.2858	0.3088	0.2858	0.2023	0	y = 0.1667
y = 0	0	0	0	0	0	0	0	y = 0
z = 0.8333	x = 0	x = 0.1667	x = 0.3333	x = 0.5	x = 0.6667	x = 0.8333	x = 1	z = 0.8333
y = 1	0	0	0	0	0	0	0	y = 1

y = 0.8333	0	0.1474	0.2023	0.2166	0.2023	0.1474	0	y = 0.8333
y = 0.6667	0	0.2023	0.2858	0.3088	0.2858	0.2023	0	y = 0.6667
y = 0.5	0	0.2166	0.3088	0.3344	0.3088	0.2166	0	y = 0.5
y = 0.3333	0	0.2023	0.2858	0.3088	0.2858	0.2023	0	y = 0.3333
y = 0.1667	0	0.1474	0.2023	0.2166	0.2023	0.1474	0	y = 0.1667
y = 0	0	0	0	0	0	0	0	y = 0
z = 1	x = 0	x = 0.1667	x = 0.3333	x = 0.5	x = 0.6667	x = 0.8333	x = 1	z = 1
y = 1	0	0	0	0	0	0	0	y = 1
y = 0.8333	0	0	0	0	0	0	0	y = 0.8333
y = 0.6667	0	0	0	0	0	0	0	y = 0.6667
y = 0.5	0	0	0	0	0	0	0	y = 0.5
y = 0.3333	0	0	0	0	0	0	0	y = 0.3333
y = 0.1667	0	0	0	0	0	0	0	y = 0.1667
y = 0	0	0	0	0	0	0	0	y = 0
	x = 0	x = 0.1667	x = 0.3333	x = 0.5	x = 0.6667	x = 0.8333	x = 1	