REFERENCES

Assawaphomthada, C. (2006) The Natural Gas Reservoir Simulation. M.S. Thesis, The Petroleum and Petrochemical Collage, Chulalongkorn University, Bangkok, Thailand.

Baoyan, L., Zhangxin, C., and Guanren, H. (2003) The sequential method for the black-oil reservoir simulation on unstructured grids. Journal of Computational Physics, 192, 36-72.

Baoyan, L., Zhangxin, C., and Guanren, H. (2004) Comparison of solution schemes for black oil reservoir simulations with unstructured grids. Computer Methods in Applied Mechanics and Engineering, 193, 319-355.

Beggs, H.D. (2004) Production Optimization Using NODAL™ Analysis. Tulsa: Oil & Gas Consultants International, Inc.

Carnahan, B., Luther, H.A., and Wilkes J.O. (2002) Applied Numerical Methods. Malabar, Florida: Robert E. Krieger Publishing Company, Inc.

Chamnakyut, K. (2004) A simulation of the underground storage of natural gas. M.S. Thesis, The Petroleum and Petrochemical Collage, Chulalongkorn University, Bangkok, Thailand.

Chapra, S.C. and Canale, R.P. (2002) Numerical Methods for Engineers. Boston: McGraw-Hill Companies, Inc.

Craft, B.C., Hawkins, M.F., and Terry R.D. (1991) Applied Petroleum Reservoir Engineering. Boston: Prentice-Hall, Inc.

Dawe, R.A. (2000) Modern Petroleum Technology Volume I Upstream. New York: John Wiley & Sons, LTD.

Edgelow, C. (1992) Natural Gas Processing. Edmonton, Alberta: Canadian Institute for Petroleum Industry Development.

Henderson, N., Flores, E., Sampaio, M., Freitas, L., and Platt, G.M. (2005) Supercritical fluid flow in porous media: modeling and simulation. Progress In Electromagnetics Research, PIER 52, 23-46.

Ibrahiem, A., Dale, C., Tabbara, W., and Wiart, J. (2005) Analysis of the temperature increase linked to the power induced by RF source. Journal of Chemical Engineering Science, 60, 1797-1808.

Pozrikidis, C. (1998) Numerical Computation in Science ad Engineering. New York: Oxford University Press, Inc.

Ridha, B.C. (2004) Use of reservoir simulation for optimizing recovery performance. Journal of Petroleum Science and Engineering, 42, 183-194.

Rojey, A. (1997) Natural Gas: Petroleum Processing Transport. Paris: Editionstechnip.

Ruben, J. (2004) A variational multiscale finite element method for multiphase flow in porous media. Finite Elements in Analysis and Design, 41, 763-777.

William, K.S. and Roland, W.L. (2002) Three-dimensional finite element simulation of three-phase flow in a deforming fissured reservoir. Comput. Methods Appl. Mech. Enrgy, 191, 2631-2659.

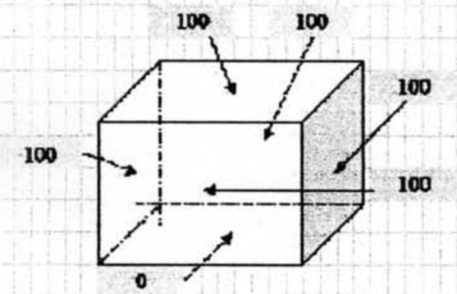
Wilkes, J.O. (1999) Fluid Mechanics for Chemical Engineers. New Jersey: Prentice-Hall, Inc.

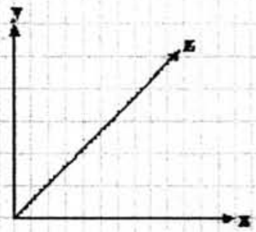
Wu, Y.S. and Pruess, K. (1998) A Numerical Method for Simulation non-Newtonian Fluid Flow and Displacement in Porous Media. Advances in Water Resources, 21, 351-362.

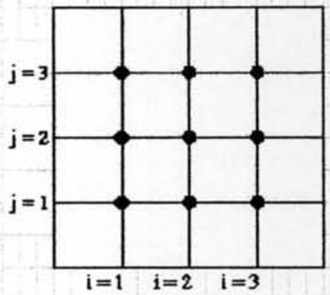
APPENDICES

APPENDIX A ADI Method by Calculation of Excel Spreadsheet.

Example ขนาด 4 x 4 x 4







k=1

	i=1	i=2	i=3
j=3	25	25	25
j=2	25	25	25
j=1	25	25	25

k=2

	i=1	i=2	i=3
j=3	25	25	25
j=2	25	25	25
j=1	25	25	25

k=3

	i=1	i=2	i=3
j=3	25	25	25
j=2	25	25	25
j=1	25	25	25

$\Delta x = \Delta y = \Delta z$

k = 0.835

Δt = 10

Δx = 10

λ = 0.0835

Forward

$P_N = \gamma_N$

$P_i = \gamma_i - \dfrac{c_i P_{i+1}}{\beta_i}$, i = N-1, N-2,…,1.

Backward

$\beta_1 = b_1$, $\gamma_1 = d_1/\beta_1$.

$\beta_i = b_i - \dfrac{a_i c_{i-1}}{\beta_{i-1}}$, i = 2,3,…,N.

$\gamma_i = \dfrac{d_i - a_i \gamma_{i-1}}{\beta_i}$, i = 2,3,…,N.

Step 1. ----- $t = t_0 + \Delta t/3$

k=1

	i=1	i=2	i=3
j=3	28.353	28.353	28.353
j=2	26.320	26.320	26.320
j=1	25.643	25.643	25.643

k=2

	i=1	i=2	i=3
j=3	27.154	27.154	27.154
j=2	25.070	25.070	25.070
j=1	24.375	24.375	24.375

k=3

	i=1	i=2	i=3
j=3	28.353	28.353	28.353
j=2	26.320	26.320	26.320
j=1	25.643	25.643	25.643

Step 2. ----- $t = t_0 + (\Delta t \times 2/3)$

k=1	i=1	i=2	i=3
j=3	31.378	29.436	31.378
j=2	29.505	27.508	29.505
j=1	28.881	26.865	28.881

k=2	i=1	i=2	i=3
j=3	29.361	27.338	29.361
j=2	27.345	25.264	27.345
j=1	26.673	24.573	26.673

k=3	i=1	i=2	i=3
j=3	31.378	29.436	31.378
j=2	29.505	27.508	29.505
j=1	28.881	26.865	28.881

Step 3. ----- $t = t_0 + \Delta t$

k=1	i=1	i=2	i=3
j=3	34.071	30.525	34.071
j=2	31.458	27.676	31.458
j=1	29.265	25.593	29.265

k=2	i=1	i=2	i=3
j=3	32.205	28.535	32.205
j=2	29.366	25.448	29.366
j=1	27.233	23.429	27.233

k=3	i=1	i=2	i=3
j=3	34.071	30.525	34.071
j=2	31.458	27.676	31.458
j=1	29.265	25.593	29.265

Input

```
Amount of node:           3
Input Dt:
10.0
Input Dx:
10.0
Input k:
0.835
```

Output

```
ratio =     .08350
```

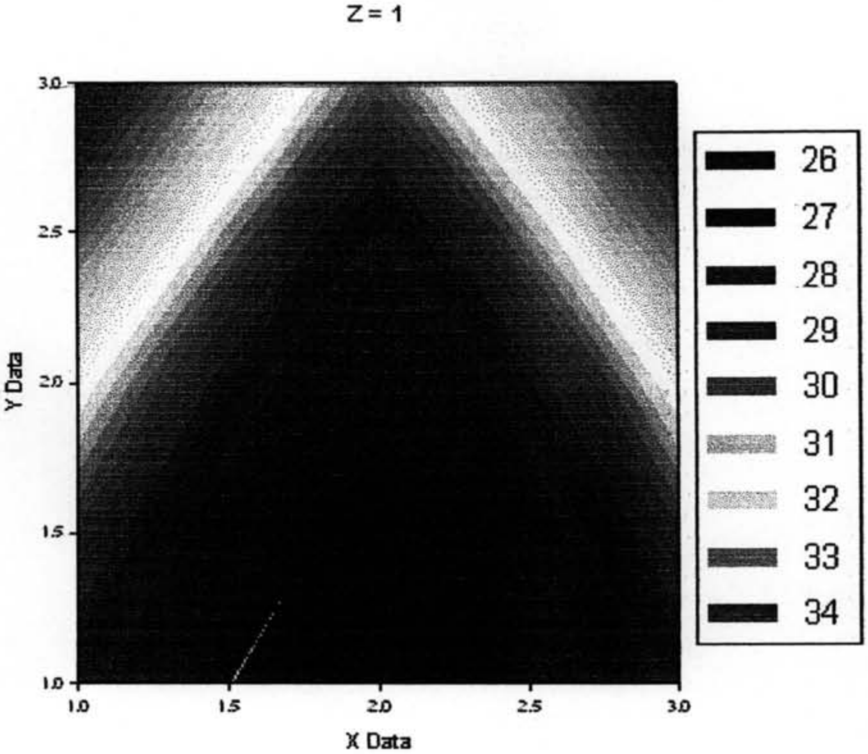
```
Initial Temperature at plane k:          1
          25.00000        25.00000       25.00000
          25.00000        25.00000       25.00000
          25.00000        25.00000       25.00000

Initial Temperature at plane k:          2
          25.00000        25.00000       25.00000
          25.00000        25.00000       25.00000
          25.00000        25.00000       25.00000

Initial Temperature at plane k:          3
          25.00000        25.00000       25.00000
          25.00000        25.00000       25.00000
          25.00000        25.00000       25.00000
```

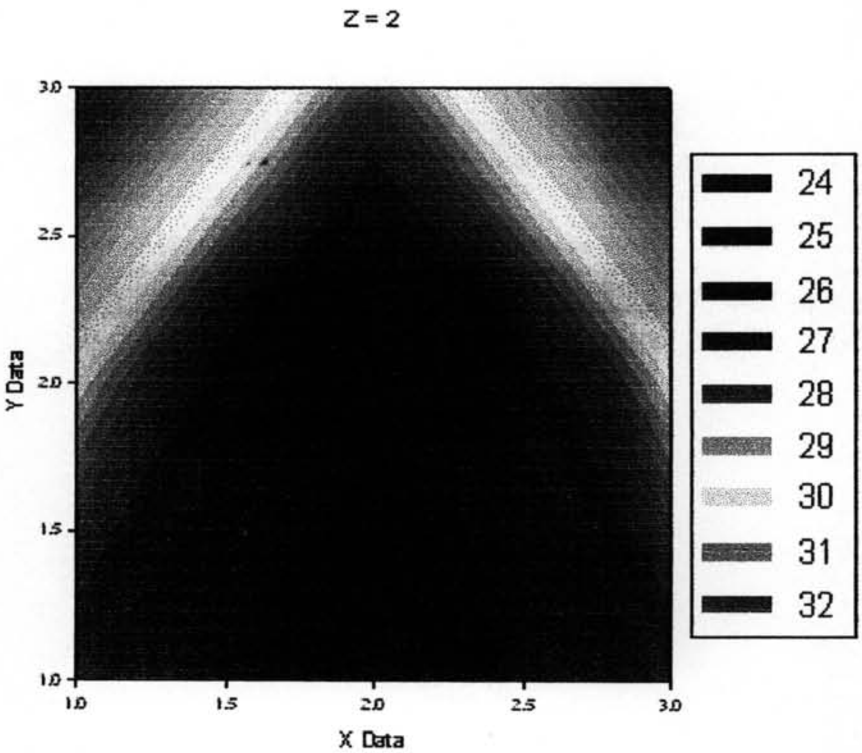
```
a =   -.08350   b =   3.16700   c =   -.08350
```

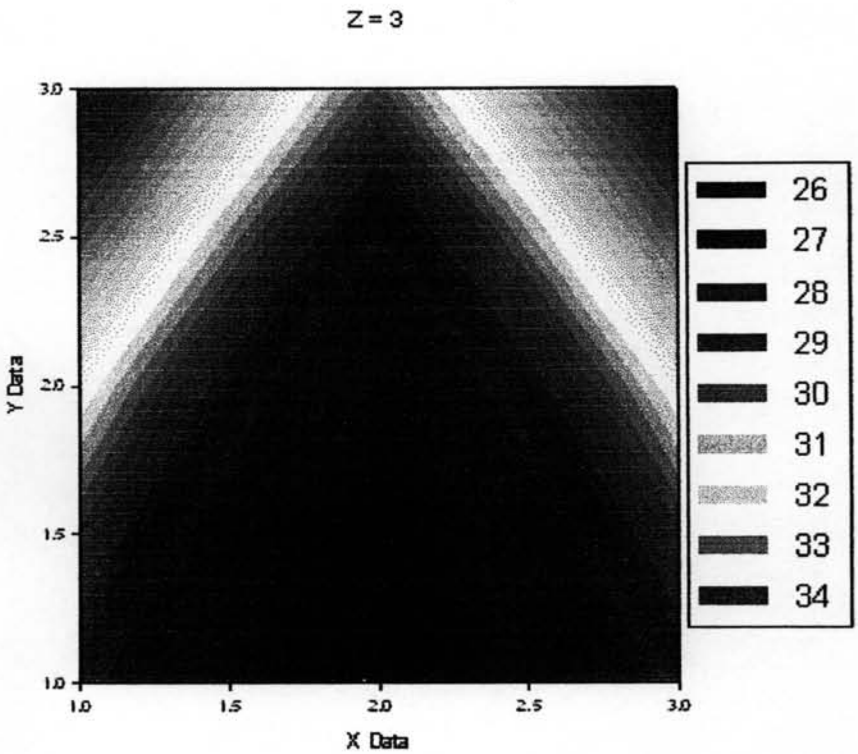
```
 First Step
 At t =   3.33333

Temperature of first Step at plane k:           1
          28.35250        28.35250        28.35250
          26.32012        26.32012        26.32012
          25.64265        25.64265        25.64265

Temperature of first Step at plane k:           2
          27.15421        27.15421        27.15421
          25.06961        25.06961        25.06961
          24.37475        24.37475        24.37475

Temperature of first Step at plane k:           3
          28.35250        28.35250        28.35250
          26.32012        26.32012        26.32012
          25.64265        25.64265        25.64265
```

```
Second Step
 At t =   6.66667

Temperature of Second Step at plane k:          1
          31.37802        29.43565        31.37802
          29.50546        27.50797        29.50546
          28.88127        26.86541        28.88127

Temperature of Second Step at plane k:          2
          29.36075        27.33770        29.36075
          27.34505        25.26413        27.34505
          26.67315        24.57294        26.67315

Temperature of Second Step at plane k:          3
          31.37802        29.43565        31.37802
          29.50546        27.50797        29.50546
          28.88127        26.86541        28.88127
```

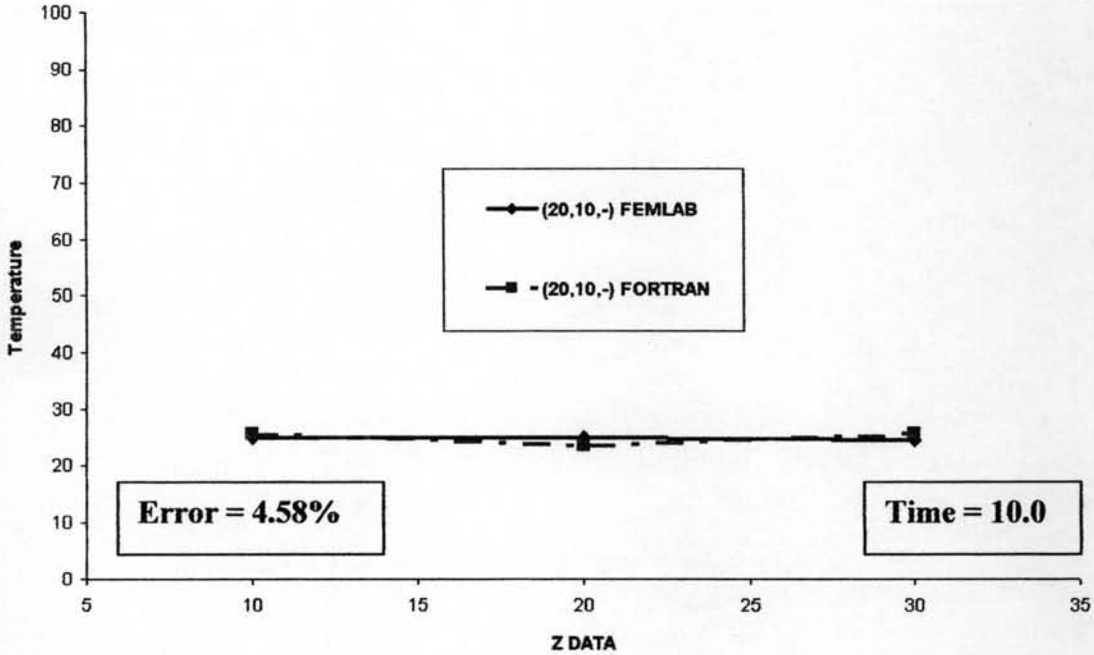
```
Third Step
 At t =  10.00000

Final temperature at plane k:           1
          34.07078        30.52525        34.07078
          31.45836        27.67626        31.45836
          29.26466        25.59338        29.26466

Final temperature at plane k:           2
          32.20519        28.53489        32.20519
          29.36640        25.44816        29.36640
          27.23330        23.42922        27.23330

Final temperature at plane k:           3
          34.07077        30.52524        34.07077
          31.45835        27.67626        31.45835
          29.26466        25.59338        29.26466
```

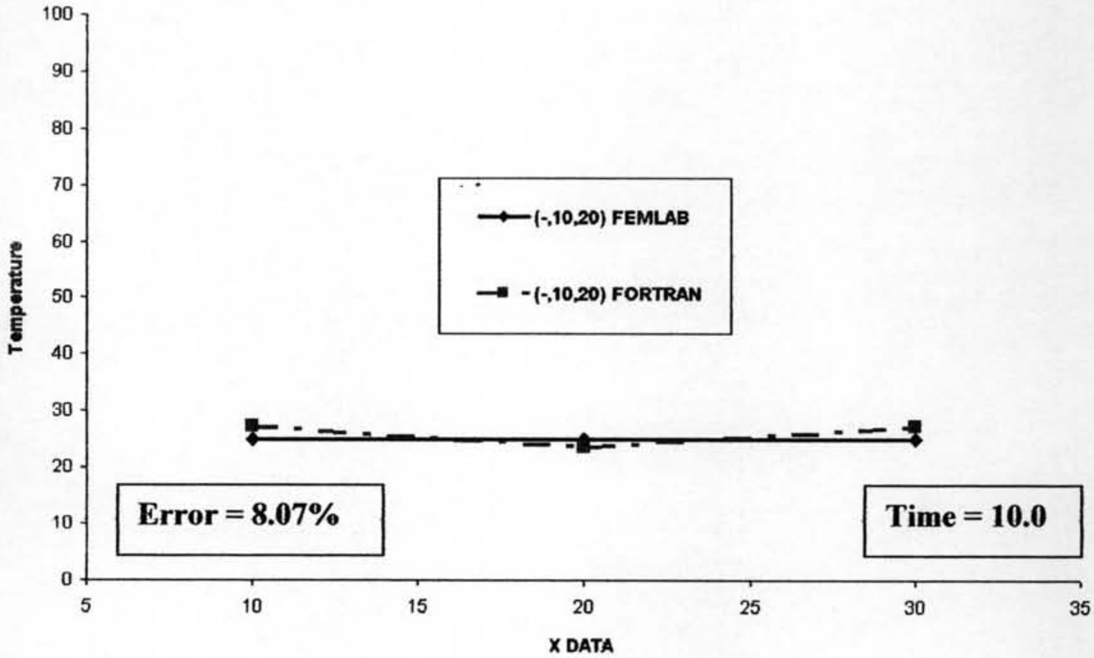

Z = 1
3.0
2.5
2.0
1.5
1.0
Y Data
1.0
1.5
2.0
2.5
3.0
X Data
26
27
28
29
30
31
32
33
34


Z = 2
3.0
2.5
2.0
1.5
1.0
Y Data
1.0
1.5
2.0
2.5
3.0
X Data
24
25
26
27
28
29
30
31
32


Z = 3
Y Data
X Data
1.0
1.5
2.0
2.5
3.0
26
27
28
29
30
31
32
33
34

APPENDIX B Comparison of Results between FEMLAB and ADI Method.





APPENDIX C Fortran Code of ADI Method to solve Heat Transfer Problem

```
c       Applied Numerical Methods, Regular Shape
c       Three-dimensional Heat Transfer Problem
c       Solved By ADI Technique

c       T = 0 at the bottom surface,
c       T = 100 at the other surface,
c       T = 25 everywhere in this cubic.

        IMPLICIT NONE
        Integer N,NN,NT
        Integer i,j,k,ICOUNT
        Real a,b,c,d,d1,d2,f
        Real Dt,Dx,constant,t,t0
        Real ratio,Ti,T1,T2,Temp,V
        Dimension Ti(5,5,5),T1(5,5,5),T2(5,5,5),Temp(5,5,5)
        Dimension V(3),d(3),d1(3),d2(3)

        OPEN(6,FILE='Temperature 2 OUTPUT.DOC')
        OPEN(7,FILE='Temperature 2 OUTPUT.DAT')
        OPEN(8,FILE='Third Step 2 OUTPUT.DAT')

c       .....Set Basic Values.....
        ICOUNT = 0.0
        N = 3
        Write(6,*) 'Amounts of node: ',N
        t0 = 0
        t = 0
        NN = N+1
        NT = N+2
```

```
c       .....Read and Check Input Parameter.....
        Dt = 10.0
        Write(6,*) 'Dt: ',Dt
        Dx = 10.0
        Write(6,*) 'Dx: ',Dx
        constant = 0.835
        Write(6,*) 'k: ',constant
        Write(6,*)
        ratio = constant*Dt/(Dx**2)
        Write(6,800) ratio
        Write(6,*)

c       .....Set Initial and Boundary Values.....
        Do 500 j = 2,NN
         Do 501 k = 2,NN
         Ti(1,j,k)  = 100
         Ti(NT,j,k) = 100
         T1(1,j,k)  = Ti(1,j,k)
         T1(NT,j,k) = Ti(NT,j,k)
         T2(1,j,k)  = Ti(1,j,k)
         T2(NT,j,k) = Ti(NT,j,k)
501      Continue
500     Continue

        Do 502 i = 2,NN
         Do 503 k = 2,NN
         Ti(i,1,k)  = 0
         Ti(i,NT,k) = 100
         T1(i,1,k)  = Ti(i,1,k)
         T1(i,NT,k) = Ti(i,NT,k)
         T2(i,1,k)  = Ti(i,1,k)
         T2(i,NT,k) = Ti(i,NT,k)
```

```
503     Continue
502     Continue

        Do 504 i = 2,NN
         Do 505 j = 2,NN
         Ti(i,j,1)  = 100
         Ti(i,j,NT) = 100
         T1(i,j,1)  = Ti(i,j,1)
         T1(i,j,NT) = Ti(i,j,NT)
         T2(i,j,1)  = Ti(i,j,1)
         T2(i,j,NT) = Ti(i,j,NT)
505      Continue
504     Continue

        Do 506 i = 2,NN
         Do 507 j = 2,NN
          Do 508 k = 2,NN
          Ti(i,j,k) = 25
508       Continue
507      Continue
506     Continue

        Do 510 j = NN,2,-1
        Write(7,803) (Ti(i,j,2),i=2,NN),(Ti(i,j,3),i=2,NN),
       *             (Ti(i,j,4),i=2,NN)

        Write(8,803) (Ti(i,j,2),i=2,NN),(Ti(i,j,3),i=2,NN),
       *             (Ti(i,j,4),i=2,NN)
510     Continue
        Write(7,*)
        Write(8,*)
```

```
c          .....Set Coefficient Array a, b, and c
           a = -1*ratio
           b = 3+(2*ratio)
           c = -1*ratio
           f = 3-(4*ratio)
           Write(6,801) a,b,c
           Write(6,*)

300        ICOUNT = ICOUNT + 1
           Write(6,*) ' ICOUNT: ',ICOUNT
           Write(7,*) ' ICOUNT: ',ICOUNT
           Write(8,*) ' ICOUNT: ',ICOUNT

c          .....Compute Temperatures at First Step.....
c          .....Implicit in x direction.....
           t = t0 + Dt/3
           Write(6,*) ' First Step'
           Write(6,802) t
           Write(6,*)

           Do 550 i = 2,NN
            Do 551 j = 2,NN
             Do 552 k = 2,NN
             d(k) = ratio*Ti(i,j-1,k)+ratio*Ti(i,j+1,k)+ratio*Ti(i,j,k-1)
          *        +ratio*Ti(i,j,k+1)+f*Ti(i,j,k)
552          Continue
            Call TRIDAG (2,NN,a,b,c,d,V)
             Do 553 k = 2,NN
             T1(i,j,k) = V(k)
553          Continue
551         Continue
550        Continue
```

```
      Do 556 j = NN,2,-1
      Write(7,803) (T1(i,j,2),i=2,NN),(T1(i,j,3),i=2,NN),
     *             (T1(i,j,4),i=2,NN)
556   Continue
      Write(7,*)

c     .....Compute Temperatures at Second Step.....
c     .....Implicit in y direction.....
      t = t0 + 2*Dt/3
      Write(6,*) 'Second Step'
      Write(6,802) t
      Write(6,*)

      Do 600 j = 2,NN
       Do 601 i = 2,NN
        Do 602 k = 2,NN
        d1(k) = ratio*T1(i,j,k-1)+ratio*T1(i,j,k+1)
     *         +ratio*T1(i-1,j,k)+ratio*T1(i+1,j,k)+f*T1(i,j,k)
602     Continue
       Call TRIDAG (2,NN,a,b,c,d1,V)
        Do 603 k = 2,NN
        T2(i,j,k) = V(k)
603     Continue
601    Continue
600   Continue

      Do 606 j = NN,2,-1
      Write(7,803) (T2(i,j,2),i=2,NN),(T2(i,j,3),i=2,NN),
     *             (T2(i,j,4),i=2,NN)
606   Continue
      Write(7,*)
```

```
c       .....Compute Temperatures at Third Step......
c       .....Implicit in z direction.....
        t = t0 + Dt
        Write(6,*) 'Third Step'
        Write(6,802) t
        Write(6,*)

        Do 650 k = 2,NN
         Do 651 i = 2,NN
          Do 652 j = 2,NN
          d2(j) = ratio*T2(i-1,j,k)+ratio*T2(i+1,j,k)
     *          +ratio*T2(i,j-1,k)+ratio*T2(i,j+1,k)+f*T2(i,j,k)
652       Continue
         Call TRIDAG (2,NN,a,b,c,d2,V)
          Do 653 j = 2,NN
          Temp(i,j,k) = V(j)
          Ti(i,j,k) = Temp(i,j,k)
653       Continue
651      Continue
650     Continue

        Do 656 j = NN,2,-1
        Write(7,803) (Temp(i,j,2),i=2,NN),(Temp(i,j,3),i=2,NN),
     *               (Temp(i,j,4),i=2,NN)

        Write(8,803) (Temp(i,j,2),i=2,NN),(Temp(i,j,3),i=2,NN),
     *               (Temp(i,j,4),i=2,NN)
656     Continue
        Write(7,*)
        Write(8,*)
```

```
c       .....Checking Conditions.....
        t0 = t
        If (ICOUNT .NE. 200) Then
        GOTO 300
        End If

c       .....Format of Output Statement.....
800     Format(2x,'ratio =',F10.5)
801     Format(2x,'a =',F10.5,3x,'b =',F10.5,3x,'c =',F10.5)
802     Format(2x,'At t =',F10.5)
803     Format(2x,3F10.5,5x,'###',3F10.5,5x,'###',3F10.5)

        CLOSE(6)
        CLOSE(7)
        CLOSE(8)

        Stop
        End
```

```
        Subroutine TRIDAG (First,Last,a,b,c,d,V)
c       .....Procedure for solving a system of simultaneous.....
c       .....Linear equation with a tri-diagonal coefficient matrix.....

        IMPLICIT NONE
        Integer First,Last,e
        Real a,b,c,d,v,gamma,beta
        Dimension d(3),V(3)
        Dimension gamma(3),beta(3)

c       .....Compute Intermediate Arrays Beta and Gamma.....
        beta(First) = b
        gamma(First) = d(First)/beta(First)

        Do 900 e = First+1,Last
        beta(e) = b-(a*c)/beta(e-1)
        gamma(e) = (d(e)-(a*gamma(e-1)))/beta(e)
900     Continue

c       .....Compute Temperature.....
        V(Last) = gamma(Last)
        Do 901 e = Last-1,First,-1
        V(e) = gamma(e)-(c*V(e+1))/beta(e)
901     Continue

        Return
        End
```

APPENDIX D Fluid Flow from Reservoir into Well

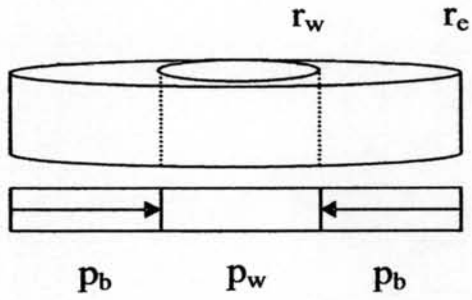


Figure D1. Radial flow system.

<u>Assume</u>: Natural gas flow in reservoir behaves like flow in radial system
From Darcy's law

$$q = vA \quad \text{(D-1)}$$

$$q = -\frac{k}{\mu}\frac{dp}{dr} * 2\pi rh \quad \text{(D-2)}$$

Compare volumetric flow at standard condition,

$$Qi = \frac{\rho q}{\rho_{sc}} \quad \text{(D-3)}$$

When,

$$\frac{\rho}{\rho_{sc}} = \frac{pT_{sc}}{Zp_{sc}T} \quad \text{(D-4)}$$

Replace Eq. (D-4) into Eq. (D-3)

$$Q_i = -\frac{pT_{sc}}{Zp_{sc}T} * 2\pi rh * \frac{k}{\mu}\frac{dp}{dr} \tag{D-5}$$

Integration from surface (r_w) to any location (r)

BC. $r = r_w, \quad p = p_{wi}$

$r = r, \quad p = p$

$$\frac{-Q_i Zp_{sc}T\mu}{2\pi hkT_{sc}} \int_{r_w}^{r} dr / r = \int_{p_{wi}}^{p} pdp \tag{D-6}$$

$$-\frac{Q_i TZp_{sc}\mu}{T_{sc}\pi hk} \ln\frac{r}{r_w} = (p^2 - p_{si}^2) \tag{D-7}$$

$$p^2 = p_{si}^2 - \frac{Q_i TZp_{sc}\mu}{T_{sc}\pi hk} \ln\frac{r}{r_w} \tag{D-8}$$

Get p_b^2 from the average of p^2 over radius r_w and r_e

$$p_b^2 = \frac{\int_{r_w}^{r_e} p^2 rdr}{\int_{r_w}^{r_e} rdr} \tag{D-9}$$

$$p_b^2 = \frac{\int_{r_w}^{r_e} (p_{wi}^2 - \frac{Q_i TZp_{sc}\mu}{T_{sc}\pi hk} \ln\frac{r}{rw}) rdr}{\int_{r_w}^{r_e} rdr} \tag{D-10}$$

$$p_b^2 = \frac{\left[\frac{r^2}{2} p_{wi}^2 - \left(\frac{Q_i TZp_{sc}\mu}{T_{sc}\pi hk} \times \frac{r^2}{2}(\ln\frac{r}{rw} - 0.5)\right)\right]_{r_w}^{r_e}}{\frac{r_e^{\,2}}{2} - \frac{r_w^{\,2}}{2}} \tag{D-11}$$

Gives, $r_e^2 - r_w^2 = r_e^2 (1 - (r_w^2 / r_e^2)) \approx r_e^2$

$$p_b^2 = \frac{2}{re^2}\left[\frac{r^2}{2} p_{wi}^2 - \left(\frac{Q_i TZp_{sc}\mu}{T_{sc}\pi hk} \times \frac{r^2}{2}(\ln\frac{r}{r_w} - 0.5)\right)\right]_{r_w}^{r_e} \tag{D-12}$$

$$p_b^2 = p_{wi}^2 - \frac{Q_i TZp_{sc}\mu}{T_{sc}\pi hk} * \frac{2}{r_e^{\,2}}\left[\frac{r^2}{2}(\ln\frac{r}{r_w} - 0.5)\right]_{r_w}^{r_e} \tag{D-13}$$

$$p_b^2 = p_{wi}^2 - \frac{Q_i TZp_{sc}\mu}{T_{sc}\pi hk} * \frac{2}{r_e^{\,2}}\left[\frac{r_e^{\,2}}{2}\ln\frac{r_e}{r_w} - 0.5\frac{r_e^{\,2}}{2} - \frac{r_w^{\,2}}{2}\ln\frac{r_w}{r_w} + 0.5\frac{r_w^{\,2}}{2}\right] \tag{D-14}$$

$$p_b^2 = p_{wi}^2 - \frac{Q_i TZp_{sc}\mu}{T_{sc}\pi hk} * \frac{2}{r_e^{\,2}}\left[\frac{re^2}{2}\ln\frac{r_e}{r_w} - \frac{0.5}{2}(r_e^{\,2} - r_w^{\,2})\right] \tag{D-15}$$

$$p_b^2 = p_{wi}^2 - \frac{Q_i TZp_{sc}\mu}{T_{sc}\pi hk}(\ln\frac{r_e}{r_w} - 0.5) \tag{D-16}$$

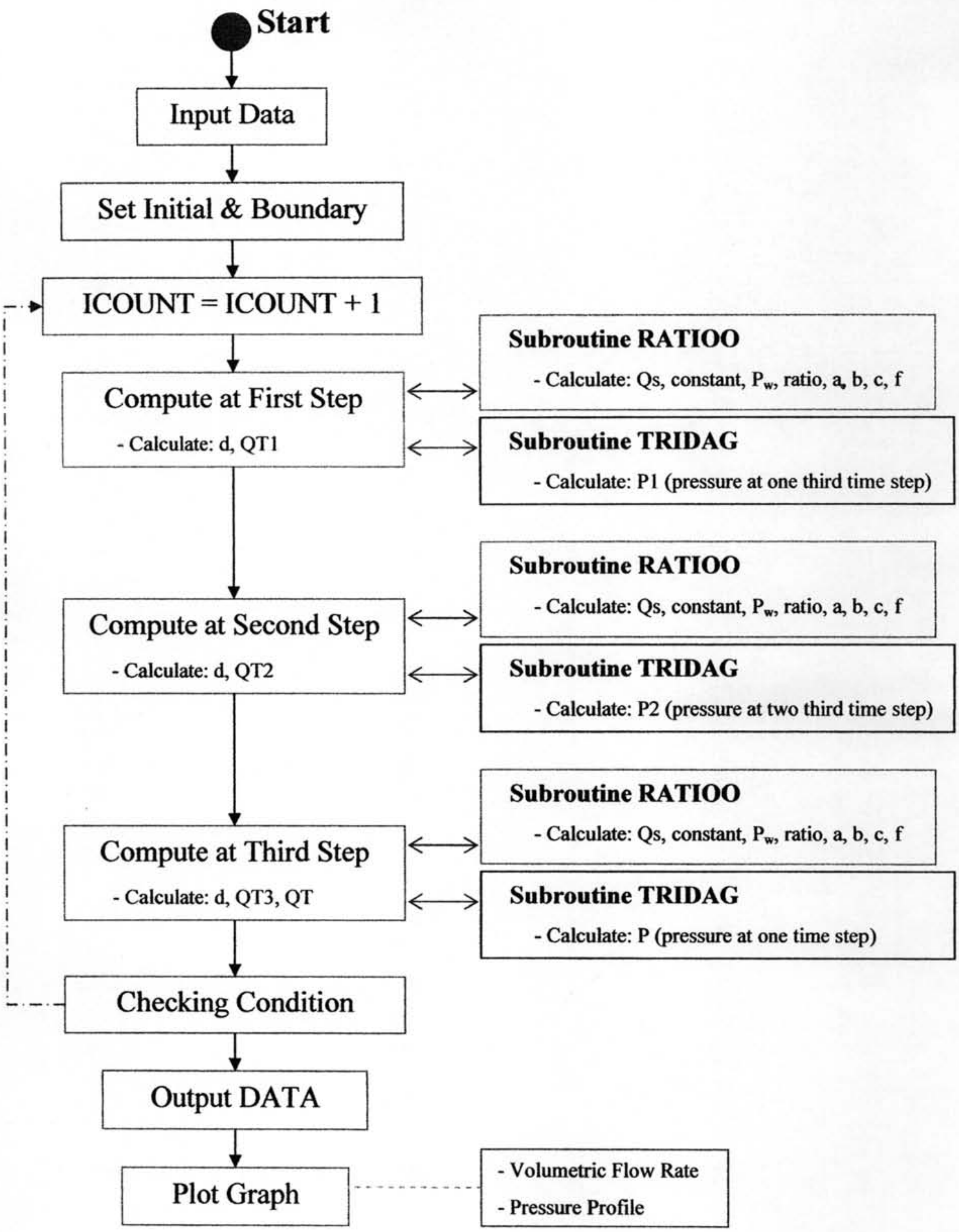
$$p_{wi}^2 - p_b^2 = \frac{Q_i TZp_{sc}\mu}{T_{sc}\pi hk}(\ln\frac{r_e}{r_w} - 0.5) \tag{D-17}$$

Rearrange Eq. (D-17) into volume flow rate, Q_i ;

$$Q_i = \frac{T_{sc}\pi hk}{TZP_{sc}\mu} \frac{(p_{wi}^2 - p_b^2)}{\ln\frac{r_e}{r_w} - 0.5} \tag{D-18}$$

where, p_{wi}, is well pressure; p_b, the block average pressure; Qi, the volume flow rate; r_e, the equivalent radius of external boundary; r_w, the well radius; Z, the mean compressibility factor; μ, the gas viscosity; k, the rock permeability; and h, the reservoir thickness.

APPENDIX E Flow Chart of the Reservoir Simulation Programming



APPENDIX F Fortran Code for Computing Pressure Profile for Reservoir Simulation with One Well

```
c       ADI-Reservoir Simulation-Pressure (1 Well)

        IMPLICIT NONE
        Integer i,j,k,o,o1,o2,N,NN,COUNT,ICOUNT,Year,Day
        Real L,W,H,Dx,Dy,Dz,WellX,WellY,WellZ
        Real t,t0,Dt,constanta,constant,Por,GV,Z,Qmax
        Real Ps,Ts,Tr,Pw,rw,re,Alpha,Beta,center
        Real Pi,P1,P2,P,PHI,V,Qs,QT1,QT2,QT3,QT,PP,DQT,QTOld
        Real a,b,c,d,f,ratio,constant1,Pb
        Dimension P1(103,103,103),P2(103,103,103),P(103,103,103)
        Dimension Pi(103,103,103),PHI(103,103,103),PP(103,103,103)
        Dimension a(103),b(103),c(103),d(103),f(103)
        Dimension constant(103,103,103)
        Dimension Qs(103),V(103),ratio(103),constant1(103)

        OPEN(4,FILE='Input Data 1 Well.DAT')
        OPEN(6,FILE='OUTPUT 1 Well.DOC')
        OPEN(7,FILE='OUTPUT 1 Well.DAT')
        OPEN(8,FILE='Third Step 1 OUTPUTl.DAT')
        OPEN(9,FILE='Graph 1 OUTPUT.DAT')
        OPEN(10,FILE='QT Graph OUTPUT 1 Well.DAT')

c       .....Set Basic Values.....
        ICOUNT = 0.0
        t0 = 0.0
        t = 0.0
        constanta = 0.0
        Qmax = 0.0
        QT = 0.0
```

```
QTOld = 0.0
Write(10,*) t,QT

L = 5000.0
Write(6,*) 'Reservoir Length: ',L,' ','ft'
W = 5000.0
Write(6,*) 'Reservoir Width: ',W,' ','ft'
H = 5000.0
Write(6,*) 'Reservoir Thickness: ',H,' ','ft'
N = 51.0
Write(6,*) 'Amounts of grid: ',N
NN = N + 1

Dx = L/(N-1)
Dy = W/(N-1)
Dz = H/(N-1)
If (Dx .LE. Dy) Then
 If (Dy .LE. Dz) Then
 Dx = Dz
 Else
 Dx = Dy
 End If
Else If (Dx .LE. Dz) Then
Dx = Dz
End If
Write(6,*) 'Grid spacing: ',Dx,'  ','ft'

Dt = 1.0
Write(6,*) 'Time step: ',Dt,'  ','days'
Por = 0.148
Write(6,*) 'Porosity: ',Por
GV = 0.05
```

```
Write(6,*) 'Gas viscosity: ',GV,' ','cP'
Ps = 14.7
Write(6,*) 'Standard pressure: ',Ps,' ','psia'
Ts = 518.67
Write(6,*) 'Standard temperature: ',Ts,' ','R'
Tr = 609.67
Write(6,*) 'Reservoir temperature: ',Tr,' ','R'
Pw = 3000.0
Write(6,*) 'Well pressure: ',Pw,' ','psia'
rw = 0.5
Write(6,*) 'Well radius: ',rw,' ','ft'

re = SQRT((4.0*Dx**2.0)/(22.0/7.0))
Write(6,*) 'Equivalent radius: ',re,' ','ft'

Z = 1.0
Write(6,*) 'Compressibility factor: ',Z

Alpha = Tr*Ps/Ts
Beta = Por*SQRT(GV/2.0)

center = INT(NN/2)
WellX = center
Write(6,*) 'Well location in x direction: ', WellX
WellY = center
Write(6,*) 'Well location in y direction: ', WellY
WellZ = center
Write(6,*) 'Well location in z direction: ', WellZ

Write(6,*)
```

```
c          .....Set Initial and Boundary Values.....
          Do 400 k = NN,2,-1
           Do 401 i = 2,NN
           Read(4,*) (Pi(i,j,k),j=2,NN)
            Do 402 j = 2,NN
             If (Pi(i,j,k) .EQ. 0.0) Then
             constant(i,j,k) = 0.0
             Else If (i .EQ. WellX .AND. j .EQ. WellY .AND.
     *           k .EQ. WellZ) Then
             Pi(i,j,k) = Pw
             constant(i,j,k) = 0.001
             Else
             constant(i,j,k) = 0.001
             End If
            PHI(i,j,k) = (Pi(i,j,k)**2.0)/(2.0*GV)
            PP(i,j,k) = PHI(i,j,k)
402         Continue
401        Continue
400       Continue

          Write(7,*) 'Initial Pressure at plane k:'
           Do 457 i = 2,NN
            Do 458 j = 2,NN
            Write(7,*) i-1,j-1,Pi(i,j,2),Pi(i,j,WellZ),Pi(i,j,NN)
            Write(8,*) i-1,j-1,Pi(i,j,2),Pi(i,j,WellZ),Pi(i,j,NN)
458         Continue
           Write(9,*) i-1,Pi(i,WellY,WellZ)
457        Continue
          Write(7,*)

          Do 350 i = WellX-1,WellX+1
           Do 351 j = WellY-1,WellY+1
```

```
      Pb = Pb + Pi(i,j,WellZ)
       constanta = constanta + constant(i,j,WellZ)
351    Continue
350   Continue
      Pb = (Pb - Pi(WellX,WellY,WellZ))/8.0
      constanta = (constanta - constant(WellX,WellY,WellZ))/8.0

      Qmax = -1.0*((22.0/7.0)*constanta*(Pw**2 - Pb**2))
     *        /(Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)

      write(6,*) ' Qmax: ',Qmax,'  day-1'

300   ICOUNT = ICOUNT + 1
      Write(*,*) ' ICOUNT: ',ICOUNT
      Write(6,*) ' ICOUNT: ',ICOUNT
      Write(7,*) ' ICOUNT: ',ICOUNT
      Write(8,*) ' ICOUNT: ',ICOUNT
      Write(9,*) ' ICOUNT: ',ICOUNT
      COUNT = 0.0
      QT1 = 0.0
      QT2 = 0.0
      QT3 = 0.0
      QT = 0.0

c     .....Compute Pressure at First Step.....
c     .....Implicit in x direction.....
      t = t0 + Dt/3
      Pb = 0.0
      Write(6,*) ' First Step: ',t,'  ','days'

      Do 500 i = WellX-1,WellX+1
       Do 501 j = WellY-1,WellY+1
```

```
      Pb = Pb + Pi(i,j,WellZ)
501   Continue
500   Continue
      Pb = (Pb - Pi(WellX,WellY,WellZ))/8.0

      Write(6,*)

      Do 550 i = 2,NN
       Do 551 k = 2,NN
        Do 552 j = 2,NN
311     o = j
        Call RATIOO (i,j,k,o,constant,Dt,Beta,Dx,Alpha,PHI,Pb,Pw,Z,
     *              Qmax,GV,re,rw,Qs,ratio,constant1,a,b,c,f)

        If (i .EQ. WellX .AND. j .EQ. WellY .AND. k .EQ. WellZ) Then
        Qs(j) = 0.0
        End If

        COUNT = COUNT + 1.0
        QT1 = QT1 + Qs(j)

        If (constant(i,j-1,k) .EQ. 0) Then
        PHI(i,j-1,k) = PP(i,j+1,k)
        PHI(i,j+1,k) = PP(i,j+1,k)
        PHI(i,j,k) = PP(i,j,k)
        Else If (constant(i,j+1,k) .EQ. 0) Then
        PHI(i,j+1,k) = PP(i,j-1,k)
        PHI(i,j-1,k) = PP(i,j-1,k)
        PHI(i,j,k) = PP(i,j,k)
        Else
        PHI(i,j-1,k) = PP(i,j-1,k)
        PHI(i,j+1,k) = PP(i,j+1,k)
```

```
PHI(i,j,k) = PP(i,j,k)
End If

If (constant(i,j,k-1) .EQ. 0) Then
PHI(i,j,k-1) = PP(i,j,k+1)
PHI(i,j,k+1) = PP(i,j,k+1)
PHI(i,j,k) = PP(i,j,k)
Else If (constant(i,j,k+1) .EQ. 0) Then
PHI(i,j,k+1) = PP(i,j,k-1)
PHI(i,j,k-1) = PP(i,j,k-1)
PHI(i,j,k) = PP(i,j,k)
ELse
PHI(i,j,k-1) = PP(i,j,k-1)
PHI(i,j,k+1) = PP(i,j,k+1)
PHI(i,j,k) = PP(i,j,k)
End If

 d(j) = ratio(j)*PHI(i,j-1,k) + ratio(j)*PHI(i,j+1,k)
*       + ratio(j)*PHI(i,j,k-1) + ratio(j)*PHI(i,j,k+1)
*       + f(j)*PHI(i,j,k) + f(j)*PHI(i,j,k) - constant1(j)

If (k .EQ. WellZ) Then
 If (j .EQ. WellY-1 .AND. i .EQ. WellX) Then
 o1 = 2.0
 o2 = WellY - 1.0
 GOTO 310
 Else If (j .EQ. NN .AND. i .EQ. WellX) Then
 o1 = WellY + 1.0
 o2 = NN
 GOTO 310
 Else
 o1 = 2.0
```

```
        o2 = NN
        End If
       Else
       o1 = 2.0
       o2 = NN
       End If

552    Continue
310    Call TRIDAG (o1,o2,a,b,c,d,Pw,GV,V)
       If (j .LT. NN) Then
       j = j + 1.0
       GOTO 311
       End If

       Do 553 j = 2,NN
       If (i .EQ. WellX .AND. j .EQ. WellY .AND. k .EQ. WellZ) Then
       P1(i,j,k) = Pw
       PHI(i,j,k) = (Pw**2.0)/(2.0*GV)
       Else
       PHI(i,j,k) = V(j)
       PP(i,j,k) = PHI(i,j,k)
       P1(i,j,k) = SQRT(2.0*GV*PHI(i,j,k))
       End If
553    Continue
551    Continue
550    Continue

       QT1 = QT1/(COUNT - 1.0)

       Write(7,*) 'Pressure of First Step at plane k:'
       Do 557 i = 2,NN
        Do 558 j = 2,NN
```

```
          Write(7,*) i-1,j-1,P1(i,j,2),P1(i,j,WellZ),P1(i,j,NN)
558       Continue
          Write(7,*)
557       Continue
          Write(7,*)

c         .....Compute Pressure at Second Step.....
c         .....Implicit in y direction.....
          t = t0 + (2*Dt/3)
          Pb = 0.0
          Write(6,*) ' Second Step: ',t,'  ','days'

          Do 600 i = WellX-1,WellX+1
           Do 601 j = WellY-1,WellY+1
           Pb = Pb + P1(i,j,WellZ)
601        Continue
600       Continue
          Pb = (Pb - P1(WellX,WellY,WellZ))/8.0

          Write(6,*)

          Do 650 j = 2,NN
           Do 651 k = 2,NN
            Do 652 i = 2,NN
321         o = i
            Call RATIOO (i,j,k,o,constant,Dt,Beta,Dx,Alpha,PHI,Pb,Pw,Z,
     *                   Qmax,GV,re,rw,Qs,ratio,constant1,a,b,c,f)

            If (i .EQ. WellX .AND. j .EQ. WellY .AND. k .EQ. WellZ) Then
            Qs(i) = 0.0
            End If
```

```
COUNT = COUNT + 1.0
QT2 = QT2 + Qs(i)

If (constant(i-1,j,k) .EQ. 0) Then
PHI(i-1,j,k) = PP(i+1,j,k)
PHI(i+1,j,k) = PP(i+1,j,k)
PHI(i,j,k) = PP(i,j,k)
Else If (constant(i+1,j,k) .EQ. 0) Then
PHI(i+1,j,k) = PP(i-1,j,k)
PHI(i-1,j,k) = PP(i-1,j,k)
PHI(i,j,k) = PP(i,j,k)
Else
PHI(i-1,j,k) = PP(i-1,j,k)
PHI(i+1,j,k) = PP(i+1,j,k)
PHI(i,j,k) = PP(i,j,k)
End If

If (constant(i,j,k-1) .EQ. 0) Then
PHI(i,j,k-1) = PP(i,j,k+1)
PHI(i,j,k+1) = PP(i,j,k+1)
PHI(i,j,k) = PP(i,j,k)
Else If (constant(i,j,k+1) .EQ. 0) Then
PHI(i,j,k+1) = PP(i,j,k-1)
PHI(i,j,k-1) = PP(i,j,k-1)
PHI(i,j,k) = PP(i,j,k)
ELse
PHI(i,j,k-1) = PP(i,j,k-1)
PHI(i,j,k+1) = PP(i,j,k+1)
PHI(i,j,k) = PP(i,j,k)
End If
```

```
      d(i) =  ratio(i)*PHI(i-1,j,k) + ratio(i)*PHI(i+1,j,k)
     *         + ratio(i)*PHI(i,j,k-1) + ratio(i)*PHI(i,j,k+1)
     *         + f(i)*PHI(i,j,k) + f(i)*PHI(i,j,k) - constant1(i)

      If (k .EQ. WellZ) Then
       If (i .EQ. WellX-1 .AND. j .EQ. WellY) Then
       o1 = 2.0
       o2 = WellX - 1.0
       GOTO 320
       Else If (i .EQ. NN .AND. j .EQ. WellY) Then
       o1 = WellX + 1.0
       o2 = NN
       GOTO 320
       Else
       o1 = 2.0
       o2 = NN
       End If
      Else
      o1 = 2.0
      o2 = NN
      End If

652   Continue
320   Call TRIDAG (o1,o2,a,b,c,d,Pw,GV,V)
      If (i .LT. NN) Then
      i = i + 1.0
      GOTO 321
      End If

      Do 653 i = 2,NN
      If (i .EQ. WellX .AND. j .EQ. WellY .AND. k .EQ. WellZ) Then
      P2(i,j,k) = Pw
```

```
      PHI(i,j,k) = (Pw**2.0)/(2.0*GV)
      Else
      PHI(i,j,k) = V(i)
      PP(i,j,k) = PHI(i,j,k)
      P2(i,j,k) = SQRT(2.0*GV*PHI(i,j,k))
      End If
653   Continue
651   Continue
650   Continue

      QT2 = QT2/(COUNT - 1.0)

      Write(7,*) 'Pressure of Second Step at plane k:'
      Do 657 i = 2,NN
      Do 658 j = 2,NN
      Write(7,*) i-1,j-1,P2(i,j,2),P2(i,j,WellZ),P2(i,j,NN)
658   Continue
      Write(7,*)
657   Continue
      Write(7,*)

c     .....Compute Pressure at Third Step.....
c     .....Implicit in z direction.....
      t = t0 + Dt
      Pb = 0.0
      Write(6,*) ' Third Step: ',t,'  ','days'

      Do 700 i = WellX-1,WellX+1
      Do 701 j = WellY-1,WellY+1
      Pb = Pb + P2(i,j,WellZ)
701   Continue
700   Continue
```

```
      Pb = (Pb - P2(WellX,WellY,WellZ))/8.0

      Write(6,*)
      Write(6,*) 'Qs: ',Qs

      Do 750 k = 2,NN
       Do 751 i = 2,NN
        Do 752 j = 2,NN
331     o = j
        Call RATIOO (i,j,k,o,constant,Dt,Beta,Dx,Alpha,PHI,Pb,Pw,Z,
     *               Qmax,GV,re,rw,Qs,ratio,constant1,a,b,c,f)

        If (i .EQ. WellX .AND. j .EQ. WellY .AND. k .EQ. WellZ) Then
        Qs(j) = 0.0
        End If

        COUNT = COUNT + 1.0
        QT3 = QT3 + Qs(j)

        If (constant(i-1,j,k) .EQ. 0) Then
        PHI(i-1,j,k) = PP(i+1,j,k)
        PHI(i+1,j,k) = PP(i+1,j,k)
        PHI(i,j,k) = PP(i,j,k)
        Else If (constant(i+1,j,k) .EQ. 0) Then
        PHI(i+1,j,k) = PP(i-1,j,k)
        PHI(i-1,j,k) = PP(i-1,j,k)
        PHI(i,j,k) = PP(i,j,k)
        Else
        PHI(i-1,j,k) = PP(i-1,j,k)
        PHI(i+1,j,k) = PP(i+1,j,k)
        PHI(i,j,k) = PP(i,j,k)
        End If
```

```
If (constant(i,j-1,k) .EQ. 0) Then
PHI(i,j-1,k) = PP(i,j+1,k)
PHI(i,j+1,k) = PP(i,j+1,k)
PHI(i,j,k) = PP(i,j,k)
Else If (constant(i,j+1,k) .EQ. 0) Then
PHI(i,j-1,k) = PP(i,j-1,k)
PHI(i,j+1,k) = PP(i,j-1,k)
PHI(i,j,k) = PP(i,j,k)
Else
PHI(i,j-1,k) = PP(i,j-1,k)
PHI(i,j+1,k) = PP(i,j+1,k)
PHI(i,j,k) = PP(i,j,k)
End If

d(j) =  ratio(j)*PHI(i-1,j,k) + ratio(j)*PHI(i+1,j,k)
*        + ratio(j)*PHI(i,j-1,k) + ratio(j)*PHI(i,j+1,k)
*        + f(j)*PHI(i,j,k) + f(j)*PHI(i,j,k) - constant1(j)

If (k .EQ. WellZ) Then
If (j .EQ. WellY-1 .AND. i .EQ. WellX) Then
o1 = 2.0
o2 = WellY - 1.0
GOTO 330
Else If (j .EQ. NN .AND. i .EQ. WellX) Then
o1 = WellY + 1.0
o2 = NN
GOTO 330
Else
o1 = 2.0
o2 = NN
End If
Else
```

```
        o1 = 2.0
        o2 = NN
        End If

752     Continue
330     Call TRIDAG (o1,o2,a,b,c,d,Pw,GV,V)
        If (j .LT. NN) Then
        j = j + 1.0
        GOTO 331
        End If

        Do 753 j = 2,NN
        If (i .EQ. WellX .AND. j .EQ. WellY .AND. k .EQ. WEllZ) Then
        P(i,j,k) = Pw
        PHI(i,j,k) = (Pw**2.0)/(2.0*GV)
        Else
        PHI(i,j,k) = V(j)
        P(i,j,k) = SQRT(2.0*GV*PHI(i,j,k))
        End If
        PP(i,j,k) = PHI(i,j,k)
        Pi(i,j,k) = P(i,j,k)
753     Continue
751     Continue
750     Continue

        QT3 = QT3/(COUNT - 1.0)

        Write(7,*) 'Pressure of Third Step at plane k:'
        Do 757 i = 2,NN
        Do 758 j = 2,NN
        Write(7,*) i-1,j-1,P(i,j,2),P(i,j,WellZ),P(i,j,NN)
```

```
      If (ICOUNT .EQ. 1.0 .OR. ICOUNT .EQ. 5.0 .OR. ICOUNT .EQ. 10.0
     *   .OR. ICOUNT .EQ. 20.0 .OR. ICOUNT .EQ. 40.0
     *   .OR. ICOUNT .EQ. 70.0 .OR.  ICOUNT .EQ. 95.0) Then

      Write(8,*) i-1,j-1,P(i,j,2),P(i,j,WellZ),P(i,j,NN)

      End If

758   Continue
      Write(7,*)
      Write(9,*) i-1,P(i,WellY,WellZ)
757   Continue
      Write(7,*)
      Write(9,*)

      QT = QT1 + QT2 + QT3
      Write(10,*) t,QT

c     .....Checking Conditions.....
      DQT = ABS(QTOld - QT)

      t0 = t
      If (Pb .LE. 1000.0 .OR. DQT .LE. 0.00001 .AND. DQT .NE. 0.0) Then

      Year = INT(t/365.0)
      Day = t - (Year*365.0)
      Write(6,*) ' Production Time: ',Year,'  ','years',Day,'  ','days'
      Write(*,*) ' Production Time: ',Year,'  ','years',Day,'  ','days'
      Stop
      Else
      GOTO 300
      End If
```

```
      CLOSE(6)
      CLOSE(7)
      CLOSE(8)
      CLOSE(9)
      CLOSE(10)

      Stop
      End

      Subroutine RATIOO (i,j,k,o,constant,Dt,Beta,Dx,Alpha,PHI,Pb,Pw,Z,
     *                  Qmax,GV,re,rw,Qs,ratio,constant1,a,b,c,f)
c     .....Set Coefficient Array a, b, and c.....

      IMPLICIT NONE
      Integer i,j,k,o
      Real constant,Dt,Beta,Dx,Alpha,PHI,Pb,Pw,Z,Check
      Real GV,re,rw,Qs,ratio,constant1,a,b,c,f,Qmax
      Dimension PHI(103,103,103),constant(103,103,103)
      Dimension ratio(103),constant1(103)
      Dimension a(103),b(103),c(103),f(103),Qs(103)

      Qs(o) = Qmax
      If (constant(i,j,k) .EQ. 0) Then
      Qs(o) = -1.0*((22.0/7.0)*constant(i,j,k)*(Pw**2 - Pb**2.0))
     *        /(Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)

      GOTO 901
      Else
      Check = -1.0*(Qs(o)*Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)
     *        /((22.0/7.0)*constant(i,j,k)) + Pb**2

      End If
```

```
      If (Check .LE. 0) Then
      Qs(o) = -1.0*((22.0/7.0)*constant(i,j,k)*(Pw**2 - Pb**2.0))
     *         /(Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)

      Else
      Pw = SQRT(-1.0*(Qs(o)*Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)
     *              /((22.0/7.0)*constant(i,j,k)) + Pb**2)

      End If

901   ratio(o) = constant(i,j,k)*Dt*SQRT(PHI(i,j,k))
     *         /(3.0*Beta*Dx**2.0)

      constant1(o) = Alpha*SQRT(PHI(i,j,k))*Qs(o)*Dt/(3.0*Beta)

      a(o) = -1.0*ratio(o)
      b(o) = 1.0+(2.0*ratio(o))
      c(o) = -1.0*ratio(o)
      f(o) = (1.0-(4.0*ratio(o)))/2.0

      Return
      End

      Subroutine TRIDAG (First,Last,a,b,c,d,Pw,GV,V)
c     .....Procedure for solving a system of simultaneous.....
c     .....Linear equation with a tri-diagonal coefficient matrix.....

      IMPLICIT NONE
      Integer e,First,Last
      Real a,b,c,d,Pw,PHIw,GV,V,beta,gamma
      Dimension a(103),b(103),c(103),d(103)
      Dimension beta(103),gamma(103),V(103)
```

```
      PHIw = (Pw**2.0)/(2.0*GV)

c     .....Compute Intermediate Arrays Beta and Gamma.....
      beta(First) = b(First)
      gamma(First) = d(First)/beta(First)

      Do 950 e = First+1,Last
      beta(e) = b(e)-((a(e)*c(e-1))/beta(e-1))
      gamma(e) = (d(e)-(a(e)*gamma(e-1)))/beta(e)
950   Continue

c     .....Compute Pressure.....
      V(Last) = gamma(Last)
      If (V(Last) .LT. PHIw .AND. V(Last) .NE. 0) Then
      V(Last) = PHIw
      End If

      Do 951 e = Last-1,First,-1
       V(e) = gamma(e)-(c(e)*V(e+1))/beta(e)
       If (V(e) .LT. PHIw .AND. V(e) .NE. 0) Then
       V(e) = PHIw
       End If
951   Continue

      Return
      End
```

APPENDIX G Fortran Code for Computing Pressure Profile for Reservoir Simulation with Two Wells

```
c          ADI-Reservoir Simulation-Pressure (2 Wells)

           IMPLICIT NONE
           Integer i,j,k,ww,o,o1,o2,N,NN,COUNT,ICOUNT,Year,Day
           Integer WellX,WellY,WellZ,nw
           Real L,W,H,Dx,Dy,Dz
           Real t,t0,Dt,constanta,constant,Por,GV,Z,Qmax
           Real Ps,Ts,Tr,Pw,rw,re,Alpha,Beta
           Real P0,Pi,P1,P2,P,PHI,V,PP
           Real Qs1,Qs2,QT1,QT2,QT3,QT,QTOld,DQT
           Real a,b,c,d,f,ratio,constant1,constant2,Pb
           Dimension P1(103,103,103),P2(103,103,103),P(103,103,103)
           Dimension Pi(103,103,103),PHI(103,103,103),PP(103,103,103)
           Dimension a(103),b(103),c(103),d(103),f(103)
           Dimension constant(103,103,103),constant1(103),constant2(103)
           Dimension V(103),ratio(103),constanta(2)
           Dimension WellX(2),WellY(2),WellZ(2),Pb(2),Pw(2)
           Dimension Qmax(2),Qs1(103),Qs2(103)
           Dimension QT1(2),QT2(2),QT3(2),QT(2),QTOld(2),DQT(2)

           OPEN(4,FILE='Input Data 2 Wells.DAT')
           OPEN(6,FILE='OUTPUT 2 Wells.DOC')
           OPEN(7,FILE='Third Step 2 OUTPUT Well1.DAT')
           OPEN(8,FILE='Third Step 2 OUTPUT Well2.DAT')
           OPEN(9,FILE='Graph 2 OUTPUT Well1.DAT')
           OPEN(10,FILE='Graph 2 OUTPUT Well2.DAT')
           OPEN(11,FILE='QT Graph OUTPUT 2 Wells.DAT')
```

```
c       .....Set Basic Values.....
        ICOUNT = 0.0
        t0 = 0.0
        t = 0.0
        constanta = 0.0
        Qmax = 0.0
        QT(1) = 0.0
        QT(2) = 0.0
        QTOld(1) = 0.0
        QTOld(2) = 0.0
        P0 = 4000.0
        Write(11,*) t,QT

        L = 5000.0
        Write(6,*) 'Reservoir Length: ',L,' ','ft'
        W = 5000.0
        Write(6,*) 'Reservoir Width: ',W,' ','ft'
        H = 5000.0
        Write(6,*) 'Reservoir Thickness: ',H,' ','ft'
        N = 51.0
        Write(6,*) 'Amounts of grid: ',N
        NN = N + 1

        Dx = L/(N-1)
        Dy = W/(N-1)
        Dz = H/(N-1)
        If (Dx .LE. Dy) Then
         If (Dy .LE. Dz) Then
         Dx = Dz
         Else
         Dx = Dy
         End If
```

```
Else If (Dx .LE. Dz) Then
Dx = Dz
End If
Write(6,*) 'Grid spacing: ',Dx,' ','ft'

Dt = 1.0
Write(6,*) 'Time step: ',Dt,' ','days'
Por = 0.148
Write(6,*) 'Porosity: ',Por
GV = 0.05
Write(6,*) 'Gas viscosity: ',GV,' ','cP'
Ps = 14.7
Write(6,*) 'Standard pressure: ',Ps,' ','psia'
Ts = 518.67
Write(6,*) 'Standard temperature: ',Ts,' ','R'
Tr = 609.67
Write(6,*) 'Reservoir temperature: ',Tr,' ','R'
Pw(1) = 3000.0
Write(6,*) 'The First Well pressure: ',Pw(1),' ','psia'
Pw(2) = 2800.0
Write(6,*) 'The Second Well pressure: ',Pw(2),' ','psia'
rw = 0.5
Write(6,*) 'Well radius: ',rw,' ','ft'
re = SQRT((4.0*Dx**2.0)/(22.0/7.0))
Write(6,*) 'Equivalent radius: ',re,' ','ft'

Z = 1.0
Write(6,*) 'Compressibility factor: ',Z

Alpha = Tr*Ps/Ts
Beta = Por*SQRT(GV/2.0)
```

```
      nw = 2.0
      Write(6,*) 'Number of Wells: ',nw

      WellX(1) = 35.0+1.0
      WellY(1) = 15.0+1.0
      WellZ(1) = 26.0
      Write(6,*) 'First Well Location: ',WellX(1)-1,WellY(1)-1,WellZ(1)

      WellX(2) = 20.0+1.0
      WellY(2) = 35.0+1.0
      WellZ(2) = 26.0
      Write(6,*) 'Second Well Location: ',WellX(2)-1,WellY(2)-1,WellZ(2)

      Write(6,*)

c     .....Set Initial and Boundary Values.....
      Do 400 k = NN,2,-1
       Do 401 i = 2,NN
       Read(4,*) (Pi(i,j,k),j=2,NN)
        Do 402 j = 2,NN
         If (Pi(i,j,k) .EQ. 0.0) Then
         constant(i,j,k) = 0.0
         Else If (i .EQ. WellX(1) .AND. j .EQ. Welly(1) .AND.
     *           k .EQ. WellZ(1)) Then
         Pi(i,j,k) = Pw(1)
         constant(i,j,k) = 0.001
         Else If (i .EQ. WellX(2) .AND. j .EQ. Welly(2) .AND.
     *           k .EQ. WellZ(2)) Then
         Pi(i,j,k) = Pw(2)
         constant(i,j,k) = 0.001
         Else
         constant(i,j,k) = 0.001
```

```
          End If
          PHI(i,j,k) = (Pi(i,j,k)**2.0)/(2.0*GV)
          PP(i,j,k) = PHI(i,j,k)
402       Continue
401      Continue
400     Continue

         Do 457 i = 2,NN
          Do 458 j = 2,NN
          Write(7,*) i-1,j-1,Pi(i,j,2),Pi(i,j,WellZ(1)),Pi(i,j,NN)
          Write(8,*) i-1,j-1,Pi(i,j,2),Pi(i,j,WellZ(2)),Pi(i,j,NN)
458       Continue
         Write(9,*) i-1,Pi(i,WellY(1),WellZ(1))
         Write(10,*) i-1,Pi(i,WellY(2),WellZ(2))
457      Continue

        Do 352 ww = 1,nw
         Do 350 i = WellX(ww)-1,WellX(ww)+1
          Do 351 j = WellY(ww)-1,WellY(ww)+1
          Pb(ww) = Pb(ww) + Pi(i,j,WellZ(ww))
          constanta(ww) = constanta(ww) + constant(i,j,WellZ(ww))
351       Continue
350      Continue
         Pb(ww) = (Pb(ww) - Pi(WellX(ww),WellY(ww),WellZ(ww)))/8.0

         constanta(ww) = (constanta(ww) -
     *                 constant(WellX(ww),WellY(ww),WellZ(ww)))/8.0

         Qmax(ww) = -1.0*((22.0/7.0)*constanta(ww)*(Pw(ww)**2 -
     *              Pb(ww)**2))/(Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)

352     Continue
```

```
      Write(6,*) ' Qmax: ',Qmax,' day-1'

300   ICOUNT = ICOUNT + 1
      Write(*,*) ' ICOUNT: ',ICOUNT
      Write(6,*) ' ICOUNT: ',ICOUNT
      Write(7,*) ' ICOUNT: ',ICOUNT
      Write(8,*) ' ICOUNT: ',ICOUNT
      Write(9,*) ' ICOUNT: ',ICOUNT
      Write(10,*) ' ICOUNT: ',ICOUNT
      COUNT = 0.0
      QT1 = 0.0
      QT2 = 0.0
      QT3 = 0.0
      QT = 0.0

c     .....Compute Pressure at First Step.....
c     .....Implicit in x direction.....
      t = t0 + Dt/3
      Pb = 0.0
      Write(6,*) ' First Step: ',t,' ','days'

      Do 532 ww = 1,nw
       Do 500 i = WellX(ww)-1,WellX(ww)+1
        Do 501 j = WellY(ww)-1,WellY(ww)+1
        Pb(ww) = Pb(ww) + Pi(i,j,WellZ(ww))
501     Continue
500    Continue
       Pb(ww) = (Pb(ww) - Pi(WellX(ww),WellY(ww),WellZ(ww)))/8.0
532   Continue

      Write(6,*)
```

```
      Do 550 i = 2,NN
       Do 551 k = 2,NN
        Do 552 j = 2,NN
311     o = j
        Call RATIOO (i,j,k,o,constant,Dt,Beta,Dx,Alpha,PHI,Pb,Pw,Z,
     *               Qmax,GV,re,rw,Qs1,Qs2,ratio,constant1,constant2,
     *               a,b,c,f)

        If (i .EQ. WellX(1) .AND. j .EQ. WellY(1) .AND.
     *     k .EQ. WellZ(1)) Then
        Qs1(j) = 0.0
        End If

        If (i .EQ. WellX(2) .AND. j .EQ. WellY(2) .AND.
     *     k .EQ. WellZ(2)) Then
        Qs2(j) = 0.0
        End If

        COUNT = COUNT + 1.0
        QT1(1) = QT1(1) + Qs1(j)
        QT1(2) = QT1(2) + Qs2(j)

        If (constant(i,j-1,k) .EQ. 0) Then
        PHI(i,j-1,k) = PP(i,j+1,k)
        PHI(i,j+1,k) = PP(i,j+1,k)
        PHI(i,j,k) = PP(i,j,k)
        Else If (constant(i,j+1,k) .EQ. 0) Then
        PHI(i,j+1,k) = PP(i,j-1,k)
        PHI(i,j-1,k) = PP(i,j-1,k)
        PHI(i,j,k) = PP(i,j,k)
        Else
        PHI(i,j-1,k) = PP(i,j-1,k)
```

```
PHI(i,j+1,k) = PP(i,j+1,k)
PHI(i,j,k) = PP(i,j,k)
End If

If (constant(i,j,k-1) .EQ. 0) Then
PHI(i,j,k-1) = PP(i,j,k+1)
PHI(i,j,k+1) = PP(i,j,k+1)
PHI(i,j,k) = PP(i,j,k)
Else If (constant(i,j,k+1) .EQ. 0) Then
PHI(i,j,k+1) = PP(i,j,k-1)
PHI(i,j,k-1) = PP(i,j,k-1)
PHI(i,j,k) = PP(i,j,k)
ELse
PHI(i,j,k-1) = PP(i,j,k-1)
PHI(i,j,k+1) = PP(i,j,k+1)
PHI(i,j,k) = PP(i,j,k)
End If

d(j) =  ratio(j)*PHI(i,j-1,k) + ratio(j)*PHI(i,j+1,k)
*     + ratio(j)*PHI(i,j,k-1) + ratio(j)*PHI(i,j,k+1)
*     + f(j)*PHI(i,j,k) + f(j)*PHI(i,j,k)
*     - constant1(j) - constant2(j)

If (WellZ(1) .EQ. WellZ(2) .AND. k .EQ. WellZ(1)) Then
 If (WellX(1) .EQ. WellX(2) .AND. i .EQ. WellX(1)) Then
  If (WellY(1) .GT. WellY(2) .AND. j .EQ. WellY(2)-1) Then
  o1 = 2.0
  o2 = WellY(2) - 1.0
  GOTO 310
  Else If (WellY(1) .GT. WellY(2) .AND. j .EQ. WellY(1)-1) Then
  o1 = WellY(2) + 1.0
  o2 = WellY(1) - 1.0
```

```
GOTO 310
Else If (WellY(1) .GT. WellY(2) .AND. j .EQ. NN) Then
o1 = WellY(1) + 1.0
o2 = NN
GOTO 310

Else If (WellY(1) .LT. WellY(2) .AND. j .EQ. WellY(2)-1) Then
o1 = WellY(1) + 1.0
o2 = WellY(2) - 1.0
GOTO 310
Else If (WellY(1) .LT. WellY(2) .AND. j .EQ. WellY(1)-1) Then
o1 = 2.0
o2 = WellY(1) - 1.0
GOTO 310
Else If (WellY(1) .LT. WellY(2) .AND. j .EQ. NN) Then
o1 = WellY(2) + 1.0
o2 = NN
GOTO 310

Else If (WellY(1) .EQ. WellY(2) .AND. j .EQ. WellY(1)-1) Then
o1 = 2.0
o2 = WellY(1) - 1.0
GOTO 310
Else If (WellY(1) .EQ. WellY(2) .AND. j .EQ. NN) Then
o1 = WellY(1) + 1.0
o2 = NN
GOTO 310
End If

Else If (WellX(1) .NE. WellX(2) .AND. i .EQ. WellX(1)) Then
If (j .EQ. WellY(1)-1) Then
o1 = 2.0
```

```
o2 = WellY(1) - 1.0
GOTO 310
Else If (j .EQ. NN) Then
o1 = WellY(1) + 1.0
o2 = NN
GOTO 310
End If

Else If (WellX(1) .NE. WellX(2) .AND. i .EQ. WellX(2)) Then
If (j .EQ. WellY(2)-1) Then
o1 = 2.0
o2 = WellY(2) - 1.0
GOTO 310
Else If (j .EQ. NN) Then
o1 = WellY(2) + 1.0
o2 = NN
GOTO 310
End If

Else
o1 = 2.0
o2 = NN
End If

Else If (WellZ(1) .NE. WellZ(2) .AND. k .EQ. WellZ(1)) Then
If (j .EQ. WellY(1)-1 .AND. i .EQ. WellX(1)) Then
o1 = 2.0
o2 = WellY(1) - 1.0
GOTO 310
Else If (j .EQ. NN) Then
o1 = WellY(1) + 1.0
o2 = NN
```

```
      GOTO 310
      End If

      Else If (WellZ(1) .NE. WellZ(2) .AND. k .EQ. WellZ(2)) Then
      If (j .EQ. WellY(2)-1 .AND. i .EQ. WellX(2)) Then
      o1 = 2.0
      o2 = WellY(2) - 1.0
      GOTO 310
      Else If (j .EQ. NN) Then
      o1 = WellY(2) + 1.0
      o2 = NN
      GOTO 310
      End If

      Else
      o1 = 2.0
      o2 = NN
      End If

552   Continue
310   Call TRIDAG (o1,o2,a,b,c,d,V)
      If (j .LT. NN) Then
      j = j + 1.0
      GOTO 311
      End If

      Do 553 j = 2,NN
      If (i .EQ. WellX(1) .AND. j .EQ. WellY(1) .AND.
     *   k .EQ. WellZ(1)) Then
      P1(i,j,k) = Pw(1)
      PHI(i,j,k) = (Pw(1)**2.0)/(2.0*GV)
```

```
      Else If (i .EQ. WellX(2) .AND. j .EQ. WellY(2) .AND.
     *         k .EQ. WellZ(2)) Then
      P1(i,j,k) = Pw(2)
      PHI(i,j,k) = (Pw(2)**2.0)/(2.0*GV)
      Else
      PHI(i,j,k) = V(j)
      PP(i,j,k) = PHI(i,j,k)
      P1(i,j,k) = SQRT(2.0*GV*PHI(i,j,k))
      End If
553   Continue
551   Continue
550   Continue

      QT1(1) = QT1(1)/(COUNT - 1.0)
      QT1(2) = QT1(2)/(COUNT - 1.0)

c     .....Compute Pressure at Second Step.....
c     .....Implicit in y direction.....
      t = t0 + (2*Dt/3)
      Pb = 0.0
      Write(6,*) ' Second Step: ',t,'  ','days'

      Do 632 ww = 1,nw
       Do 600 i = WellX(ww)-1,WellX(ww)+1
        Do 601 j = WellY(ww)-1,WellY(ww)+1
        Pb(ww) = Pb(ww) + P1(i,j,WellZ(ww))
601     Continue
600    Continue
       Pb(ww) = (Pb(ww) - P1(WellX(ww),WellY(ww),WellZ(ww)))/8.0
632   Continue

      Write(6,*)
```

```
      Do 650 j = 2,NN
       Do 651 k = 2,NN
        Do 652 i = 2,NN
321     o = i
        Call RATIOO (i,j,k,o,constant,Dt,Beta,Dx,Alpha,PHI,Pb,Pw,Z,
     *                 Qmax,GV,re,rw,Qs1,Qs2,ratio,constant1,constant2,
     *                 a,b,c,f)

        If (i .EQ. WellX(1) .AND. j .EQ. WellY(1) .AND.
     *     k .EQ. WellZ(1)) Then
        Qs1(i) = 0.0
        End If

        If (i .EQ. WellX(2) .AND. j .EQ. WellY(2) .AND.
     *     k .EQ. WellZ(2)) Then
        Qs2(i) = 0.0
        End If

        COUNT = COUNT + 1.0
        QT2(1) = QT2(1) + Qs1(i)
        QT2(2) = QT2(2) + Qs2(i)

        If (constant(i-1,j,k) .EQ. 0) Then
        PHI(i-1,j,k) = PP(i+1,j,k)
        PHI(i+1,j,k) = PP(i+1,j,k)
        PHI(i,j,k) = PP(i,j,k)
        Else If (constant(i+1,j,k) .EQ. 0) Then
        PHI(i+1,j,k) = PP(i-1,j,k)
        PHI(i-1,j,k) = PP(i-1,j,k)
        PHI(i,j,k) = PP(i,j,k)
```

```
Else
PHI(i-1,j,k) = PP(i-1,j,k)
PHI(i+1,j,k) = PP(i+1,j,k)
PHI(i,j,k) = PP(i,j,k)
End If

If (constant(i,j,k-1) .EQ. 0) Then
PHI(i,j,k-1) = PP(i,j,k+1)
PHI(i,j,k+1) = PP(i,j,k+1)
PHI(i,j,k) = PP(i,j,k)
Else If (constant(i,j,k+1) .EQ. 0) Then
PHI(i,j,k+1) = PP(i,j,k-1)
PHI(i,j,k-1) = PP(i,j,k-1)
PHI(i,j,k) = PP(i,j,k)
ELse
PHI(i,j,k-1) = PP(i,j,k-1)
PHI(i,j,k+1) = PP(i,j,k+1)
PHI(i,j,k) = PP(i,j,k)
End If

d(i) = ratio(i)*PHI(i-1,j,k) + ratio(i)*PHI(i+1,j,k)
*     + ratio(i)*PHI(i,j,k-1) + ratio(i)*PHI(i,j,k+1)
*     + f(i)*PHI(i,j,k) + f(i)*PHI(i,j,k)
*     - constant1(i) - constant2(i)

If (WellZ(1) .EQ. WellZ(2) .AND. k .EQ. WellZ(1)) Then
 If (WellY(1) .EQ. WellY(2) .AND. j .EQ. WellY(1)) Then
  If (WellX(1) .GT. WellX(2) .AND. i .EQ. WellX(2)-1) Then
  o1 = 2.0
  o2 = WellX(2) - 1.0
  GOTO 320
```

```
Else If (WellX(1) .GT. WellX(2) .AND. i .EQ. WellX(1)-1) Then
o1 = WellX(2) + 1.0
o2 = WellX(1) - 1.0
GOTO 320
Else If (WellX(1) .GT. WellX(2) .AND. i .EQ. NN) Then
o1 = WellX(1) + 1.0
o2 = NN
GOTO 320

Else If (WellX(1) .LT. WellX(2) .AND. i .EQ. WellX(2)-1) Then
o1 = WellX(1) + 1.0
o2 = WellX(2) - 1.0
GOTO 320
Else If (WellX(1) .LT. WellX(2) .AND. i .EQ. WellX(1)-1) Then
o1 = 2.0
o2 = WellX(1) - 1.0
GOTO 320
Else If (WellX(1) .LT. WellX(2) .AND. i .EQ. NN) Then
o1 = WellX(2) + 1.0
o2 = NN
GOTO 320

Else If (WellX(1) .EQ. WellX(2) .AND. i .EQ. WellX(1)-1) Then
o1 = 2.0
o2 = WellX(1) - 1.0
GOTO 320
Else If (WellX(1) .EQ. WellX(2) .AND. i .EQ. NN) Then
o1 = WellX(1) + 1.0
o2 = NN
GOTO 320
End If
```

```
Else If (WellY(1) .NE. WellY(2) .AND. j .EQ. WellY(1)) Then
 If (i .EQ. WellX(1)-1) Then
 o1 = 2.0
 o2 = WellX(1) - 1.0
 GOTO 320
 Else If (i .EQ. NN) Then
 o1 = WellX(1) + 1.0
 o2 = NN
 GOTO 320
 End If

Else If (WellY(1) .NE. WellY(2) .AND. j .EQ. WellY(2)) Then
 If (i .EQ. WellX(2)-1) Then
 o1 = 2.0
 o2 = WellX(2) - 1.0
 GOTO 320
 Else If (i .EQ. NN) Then
 o1 = WellX(2) + 1.0
 o2 = NN
 GOTO 320
 End If

Else
o1 = 2.0
o2 = NN
End If

Else If (WellZ(1) .NE. WellZ(2) .AND. k .EQ. WellZ(1)) Then
If (i .EQ. WellX(1)-1 .AND. j .EQ. WellY(1)) Then
o1 = 2.0
o2 = WellX(1) - 1.0
GOTO 320
```

```
       Else If (i .EQ. NN) Then
       o1 = WellX(1) + 1.0
       o2 = NN
       GOTO 320
       End If

      Else If (WellZ(1) .NE. WellZ(2) .AND. k .EQ. WellZ(2)) Then
       If (i .EQ. WellX(2)-1 .AND. j .EQ. WellY(2)) Then
       o1 = 2.0
       o2 = WellX(2) - 1.0
       GOTO 320
       Else If (i .EQ. NN) Then
       o1 = WellX(2) + 1.0
       o2 = NN
       GOTO 320
       End If

      Else
      o1 = 2.0
      o2 = NN
      End If

652   Continue
320   Call TRIDAG (o1,o2,a,b,c,d,V)
      f (i .LT. NN) Then
      i = i + 1.0
      GOTO 321
      End If

      Do 653 i = 2,NN
      If (i .EQ. WellX(1) .AND. j .EQ. WellY(1) .AND.
     *   k .EQ. WellZ(1)) Then
```

```
              P2(i,j,k) = Pw(1)
              PHI(i,j,k) = (Pw(1)**2.0)/(2.0*GV)
              Else If (i .EQ. WellX(2) .AND. j .EQ. WellY(2) .AND.
     *            k .EQ. WellZ(2)) Then
              P2(i,j,k) = Pw(2)
              PHI(i,j,k) = (Pw(2)**2.0)/(2.0*GV)
              Else
              PHI(i,j,k) = V(i)
              PP(i,j,k) = PHI(i,j,k)
              P2(i,j,k) = SQRT(2.0*GV*PHI(i,j,k))
              End If
653           Continue
651          Continue
650         Continue

            QT2(1) = QT2(1)/(COUNT - 1.0)
            QT2(2) = QT2(2)/(COUNT - 1.0)

c           .....Compute Pressure at Third Step.....
c           .....Implicit in z direction.....
            t = t0 + Dt
            Pb = 0.0
            Write(6,*) ' Third Step: ',t,' ','days'

            Do 732 ww = 1,nw
             Do 700 i = WellX(ww)-1,WellX(ww)+1
              Do 701 j = WellY(ww)-1,WellY(ww)+1
            Pb(ww) = Pb(ww) + P2(i,j,WellZ(ww))
701           Continue
700          Continue
             Pb(ww) = (Pb(ww) - P2(WellX(ww),WellY(ww),WellZ(ww)))/8.0
732         Continue
```

```
      Write(6,*)

      Do 750 k = 2,NN
       Do 751 i = 2,NN
        Do 752 j = 2,NN
331     o = j
        Call RATIOO (i,j,k,o,constant,Dt,Beta,Dx,Alpha,PHI,Pb,Pw,Z,
     *               Qmax,GV,re,rw,Qs1,Qs2,ratio,constant1,constant2,
     *               a,b,c,f)

        If (i .EQ. WellX(1) .AND. j .EQ. WellY(1) .AND.
     *      k .EQ. WellZ(1)) Then
        Qs1(j) = 0.0
        End If

        If (i .EQ. WellX(2) .AND. j .EQ. WellY(2) .AND.
     *      k .EQ. WellZ(2)) Then
        Qs2(j) = 0.0
        End If

        COUNT = COUNT + 1.0
        QT3(1) = QT3(1) + Qs1(j)
        QT3(2) = QT3(2) + Qs2(j)

        If (constant(i-1,j,k) .EQ. 0) Then
        PHI(i-1,j,k) = PP(i+1,j,k)
        PHI(i+1,j,k) = PP(i+1,j,k)
        PHI(i,j,k) = PP(i,j,k)
        Else If (constant(i+1,j,k) .EQ. 0) Then
        PHI(i+1,j,k) = PP(i-1,j,k)
        PHI(i-1,j,k) = PP(i-1,j,k)
        PHI(i,j,k) = PP(i,j,k)
```

```
Else
PHI(i-1,j,k) = PP(i-1,j,k)
PHI(i+1,j,k) = PP(i+1,j,k)
PHI(i,j,k) = PP(i,j,k)
End If

If (constant(i,j-1,k) .EQ. 0) Then
PHI(i,j-1,k) = PP(i,j+1,k)
PHI(i,j+1,k) = PP(i,j+1,k)
PHI(i,j,k) = PP(i,j,k)
Else If (constant(i,j+1,k) .EQ. 0) Then
PHI(i,j+1,k) = PP(i,j-1,k)
PHI(i,j-1,k) = PP(i,j-1,k)
PHI(i,j,k) = PP(i,j,k)
Else
PHI(i,j-1,k) = PP(i,j-1,k)
PHI(i,j+1,k) = PP(i,j+1,k)
PHI(i,j,k) = PP(i,j,k)
End If

d(j) = ratio(j)*PHI(i-1,j,k) + ratio(j)*PHI(i+1,j,k)
*      + ratio(j)*PHI(i,j-1,k) + ratio(j)*PHI(i,j+1,k)
*      + f(j)*PHI(i,j,k) + f(j)*PHI(i,j,k)
*      - constant1(j) - constant2(j)

If (WellZ(1) .EQ. WellZ(2) .AND. k .EQ. WellZ(1)) Then
 If (WellX(1) .EQ. WellX(2) .AND. i .EQ. WellX(1)) Then
  If (WellY(1) .GT. WellY(2) .AND. j .EQ. WellY(2)-1) Then
  o1 = 2.0
  o2 = WellY(2) - 1.0
  GOTO 330
  Else If (WellY(1) .GT. WellY(2) .AND. j .EQ. WellY(1)-1) Then
```

```
o1 = WellY(2) + 1.0
o2 = WellY(1) - 1.0
GOTO 330
Else If (WellY(1) .GT. WellY(2) .AND. j .EQ. NN) Then
o1 = WellY(1) + 1.0
o2 = NN
GOTO 330

Else If (WellY(1) .LT. WellY(2) .AND. j .EQ. WellY(2)-1) Then
o1 = WellY(1) + 1.0
o2 = WellY(2) - 1.0
GOTO 330
Else If (WellY(1) .LT. WellY(2) .AND. j .EQ. WellY(1)-1) Then
o1 = 2.0
o2 = WellY(1) - 1.0
GOTO 330
Else If (WellY(1) .LT. WellY(2) .AND. j .EQ. NN) Then
o1 = WellY(2) + 1.0
o2 = NN
GOTO 330

Else If (WellY(1) .EQ. WellY(2) .AND. j .EQ. WellY(1)-1) Then
o1 = 2.0
o2 = WellY(1) - 1.0
GOTO 330
Else If (WellY(1) .EQ. WellY(2) .AND. j .EQ. NN) Then
o1 = WellY(1) + 1.0
o2 = NN
GOTO 330
End If
```

```
Else If (WellX(1) .NE. WellX(2) .AND. i .EQ. WellX(1)) Then
 If (j .EQ. WellY(1)-1) Then
 o1 = 2.0
 o2 = WellY(1) - 1.0
 GOTO 330
 Else If (j .EQ. NN) Then
 o1 = WellY(1) + 1.0
 o2 = NN
 GOTO 330
 End If

Else If (WellX(1) .NE. WellX(2) .AND. i .EQ. WellX(2)) Then
 If (j .EQ. WellY(2)-1) Then
 o1 = 2.0
 o2 = WellY(2) - 1.0
 GOTO 330
 Else If (j .EQ. NN) Then
 o1 = WellY(2) + 1.0
 o2 = NN
 GOTO 330
 End If

Else
o1 = 2.0
o2 = NN
End If

Else If (WellZ(1) .NE. WellZ(2) .AND. k .EQ. WellZ(1)) Then
If (j .EQ. WellY(1)-1 .AND. i .EQ. WellX(1)) Then
o1 = 2.0
o2 = WellY(1) - 1.0
GOTO 330
```

```
        Else If (j .EQ. NN) Then
        o1 = WellY(1) + 1.0
        o2 = NN
        GOTO 330
        End If

       Else If (WellZ(1) .NE. WellZ(2) .AND. k .EQ. WellZ(2)) Then
        If (j .EQ. WellY(2)-1 .AND. i .EQ. WellX(1)) Then
        o1 = 2.0
        o2 = WellY(2) - 1.0
        GOTO 330
        Else If (j .EQ. NN) Then
        o1 = WellY(2) + 1.0
        o2 = NN
        GOTO 330
        End If

       Else
       o1 = 2.0
       o2 = NN
       End If

752    Continue
330    Call TRIDAG (o1,o2,a,b,c,d,V)
       If (j .LT. NN) Then
       j = j + 1.0
       GOTO 331
       End If

       Do 753 j = 2,NN
       If (i .EQ. WellX(1) .AND. j .EQ. WellY(1) .AND.
     *   k .EQ. WellZ(1)) Then
```

```
      P(i,j,k) = Pw(1)
      PHI(i,j,k) = (Pw(1)**2.0)/(2.0*GV)
      Else If (i .EQ. WellX(2) .AND. j .EQ. WellY(2) .AND.
*           k .EQ. WellZ(2)) Then
      P(i,j,k) = Pw(2)
      PHI(i,j,k) = (Pw(2)**2.0)/(2.0*GV)
      Else
      PHI(i,j,k) = V(j)
      P(i,j,k) = SQRT(2.0*GV*PHI(i,j,k))
      End If
      PP(i,j,k) = PHI(i,j,k)
      Pi(i,j,k) = P(i,j,k)
753   Continue
751  Continue
750 Continue

    QT3(1) = QT3(1)/(COUNT - 1.0)
    QT3(2) = QT3(2)/(COUNT - 1.0)

    Do 757 i = 2,NN
     Do 758 j = 2,NN
     If (ICOUNT .EQ. 1.0 .OR. ICOUNT .EQ. 5.0 .OR. ICOUNT .EQ. 10.0
*      .OR. ICOUNT .EQ. 15.0 .OR. ICOUNT .EQ. 20.0
*      .OR. ICOUNT .EQ. 35.0 .OR.  ICOUNT .EQ. 51.0) Then

     Write(7,*) i-1,j-1,P(i,j,2),P(i,j,WellZ(1)),P(i,j,NN)
     Write(8,*) i-1,j-1,P(i,j,2),P(i,j,WellZ(2)),P(i,j,NN)

     End If

758  Continue
    Write(9,*) i-1,P(i,WellY(1),WellZ(1))
```

```
      Write(10,*) i-1,P(i,WellY(2),WellZ(2))
757   Continue

      Do 775 ww = 1,nw
      QT(ww) = QT1(ww) + QT2(ww) + QT3(ww)
775   Continue

      Write(11,*) t,QT

c     .....Checking Conditions.....
      Do 950 ww = 1,nw
      DQT(ww) = ABS(QTOld(ww) - QT(ww))
      QTOld(ww) = QT(ww)
950   Continue

      t0 = t
      If (Pb(1) .LE. 1000.0 .AND. Pb(2) .LE. 1000.0 .OR.
     *    DQT(1) .LE. 0.00001 .AND. DQT(2) .LE. 0.00001 .AND.
     *    DQT(1) .NE. 0.0 .AND. DQT(2) .NE. 0.0) Then
      Year = INT(t/365.0)
      Day = t - (Year*365.0)
      Write(6,*) ' Production Time: ',Year,' ','years',Day,' ','days'
      Write(*,*) ' Production Time: ',Year,' ','years',Day,' ','days'
      Stop
      Else
      GOTO 300
      End If

      CLOSE(6)
      CLOSE(7)
      CLOSE(8)
      CLOSE(9)
```

```
      CLOSE(10)

      Stop
      End

      Subroutine RATIOO (i,j,k,o,constant,Dt,Beta,Dx,Alpha,PHI,Pb,Pw,Z,
     *                  Qmax,GV,re,rw,Qs1,Qs2,ratio,constant1,constant2,
     *                  a,b,c,f)
c     .....Set Coefficient Array a, b, and c.....

      IMPLICIT NONE
      Integer i,j,k,o
      Real constant,Dt,Beta,Dx,Alpha,PHI,Pb,Pw,Z,Check
      Real GV,re,rw,Qs1,Qs2,ratio,constant1,constant2,a,b,c,f,Qmax
      Dimension PHI(103,103,103),constant(103,103,103)
      Dimension ratio(103),constant1(103),constant2(103)
      Dimension a(103),b(103),c(103),f(103),Qs1(103),Qs2(103)
      Dimension Qmax(2),Check(2),Pb(2),Pw(2)

      Qs1(o) = Qmax(1)
      Qs2(o) = Qmax(2)

      If (constant(i,j,k) .EQ. 0) Then
      Qs1(o) = -1.0*((22.0/7.0)*constant(i,j,k)*(Pw(1)**2 - Pb(1)**2.0))
     *        /(Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)

      Qs2(o) = -1.0*((22.0/7.0)*constant(i,j,k)*(Pw(2)**2 - Pb(2)**2.0))
     *        /(Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)
      GOTO 901
```

```
      Else
      Check(1) = -1.0*(Qs1(o)*Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)
     *           /((22.0/7.0)*constant(i,j,k)) + Pb(1)**2

      Check(2) = -1.0*(Qs2(o)*Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)
     *           /((22.0/7.0)*constant(i,j,k)) + Pb(2)**2

      End If

      If (Check(1) .LE. 0) Then
      Qs1(o) = -1.0*((22.0/7.0)*constant(i,j,k)*(Pw(1)**2 - Pb(1)**2.0))
     *          /(Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)

      Else
      Pw(1) = SQRT(-1.0*(Qs1(o)*Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)
     *                 /((22.0/7.0)*constant(i,j,k)) + Pb(1)**2)

      End If

      If (Check(2) .LE. 0) Then
      Qs2(o) = -1.0*((22.0/7.0)*constant(i,j,k)*(Pw(2)**2 - Pb(2)**2.0))
     *          /(Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)

      Else
      Pw(2) = SQRT(-1.0*(Qs2(o)*Z*Alpha*GV*(ALOG(re/rw)-0.5)*Dx**2.0)
     *                 /((22.0/7.0)*constant(i,j,k)) + Pb(2)**2)

      End If

901   ratio(o) = constant(i,j,k)*Dt*SQRT(PHI(i,j,k))
     *          /(3.0*Beta*Dx**2.0)
```

```
      constant1(o) = Alpha*SQRT(PHI(i,j,k))*Qs1(o)*Dt/(3.0*Beta)
      constant2(o) = Alpha*SQRT(PHI(i,j,k))*Qs2(o)*Dt/(3.0*Beta)

      a(o) = -1.0*ratio(o)
      b(o) = 1.0+(2.0*ratio(o))
      c(o) = -1.0*ratio(o)
      f(o) = (1.0-(4.0*ratio(o)))/2.0

      Return
      End

      Subroutine TRIDAG (First,Last,a,b,c,d,V)
c     .....Procedure for solving a system of simultaneous.....
c     .....Linear equation with a tri-diagonal coefficient matrix.....

      IMPLICIT NONE
      Integer e,First,Last
      Real a,b,c,d,V,beta,gamma
      Dimension a(103),b(103),c(103),d(103)
      Dimension beta(103),gamma(103),V(103)

c     .....Compute Intermediate Arrays Beta and Gamma.....
      beta(First) = b(First)
      gamma(First) = d(First)/beta(First)

      Do 950 e = First+1,Last
      beta(e) = b(e)-((a(e)*c(e-1))/beta(e-1))
      gamma(e) = (d(e)-(a(e)*gamma(e-1)))/beta(e)
950   Continue
```

```
c       .....Compute Pressure.....
        V(Last) = gamma(Last)
        Do 951 e = Last-1,First,-1
         V(e) = gamma(e)-(c(e)*V(e+1))/beta(e)
951     Continue

        Return
        End
```

CURRICULUM VITAE

Name: Mr. Harith Janthontapta-one

Date of Birth: January 23, 1981

Nationality: Thai

University Education:

1999-2003 Bachelor Degree of Chemical Engineering, Faculty of Engineering, King Mongkut's University of Technology Thonburi, Bangkok, Thailand.