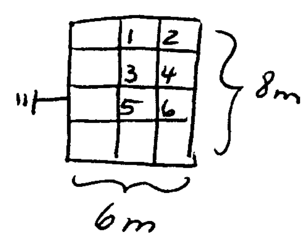
```
c*********************************
c This program solves example 2
c using finite difference method
c with SOR.
c C.Furse  4-13-95
c /grad/furse/fd_sor.f
c*********************************
      parameter(imax=4,jmax=5) ! size of rectangular duct
      real phi(imax,jmax) ! potentials
      real b(imax,jmax) ! b vector = q*h^2/eps

c--------------------------------------
c     Force Boundary Value Potentials
c     (For this problem, they are zero everywhere)
c     (Potentials on corners can be any value, because they
c        are not used in calculations.)
c--------------------------------------
      do i=1,imax
         phi(i,1) = 0.   ! top
         phi(i,jmax) = 0.   ! bottom
      enddo
      do j=1,jmax
         phi(1,j) = 0.   ! left
         phi(imax,j) = 0. ! right
      enddo

c--------------------------------------
c     Define resolution of grid: h (meters)
c--------------------------------------
      side_x = 6.  ! meters
      side_y = 8.  ! meters
      h = side_x/3.  ! resolution of grid (meters)

c--------------------------------------
c     Set up b vector: b=q*h*h/eps
c--------------------------------------
      er = 1.0 ! relative permittivity
      eo = 8.854e-12 ! electric permittivity
      eps = er*eo ! permittivity of medium inside duct
      q = 2.0*eps ! given charge density
      b_constant = -q*h*h/eps

      do i=1,imax
      do j=1,jmax
         b(i,j) = b_constant
      enddo
      enddo

c----------------------------------------
c Set up SOR constants
c----------------------------------------
      ww = 1.0 ! overrelaxation constant (1.<ww<2.)
      maxit = 25 ! maximum # of iterations
      xchange = 1e-6  ! maximum change in phi vector form k to k+1

c--------------------------------------
c     Find all internal potentials using FD and SOR
c--------------------------------------
50    do k=1,maxit ! iterate for SOR
         xch = 1e6 ! initialize
         do i=2,imax-1
         do j=2,jmax-1
            Rij = 0.25*(phi(i+1,j) + phi(i-1,j) + phi(i,j+1)
     &           + phi(i,j-1) -4.*phi(i,j) -b(i,j))
            phi_k = phi(i,j) ! kth iteration
            phi(i,j) = phi_k + ww*Rij ! (k+1) iteration /4
            xch = min(xch,abs(phi(i,j)-phi_k))
         enddo
```



```
         enddo
         if (xch.le.xchange) then ! finished
            write(6,*) 'Finished:'
            write(6,*) '  maximum change in phi=',xch
            write(6,*) '  number of iterations =',k
            goto 100
         endif
      enddo

c------------------------------
c What to do if max # of iterations exceeded:
c------------------------------
      write(6,*) 'Maximum iterations reached:'
      write(6,*) '  maximum change in phi=',xch
      write(6,*) '  number of iterations =',k
      write(6,*) 'How many more iterations ? (0 to quit)'
      read(5,*) iter
      if (iter.gt.0) then
         maxit = maxit + iter
         goto 50
      endif

c--------------------------------
c Finish
c--------------------------------
100   do i=2,imax-1
      do j=2,jmax-1
         write(6,*) 'i,j,phi=',i,j,phi(i,j)
      enddo
      enddo

      end
```

EXAMPLE 2:



$h = 2m$

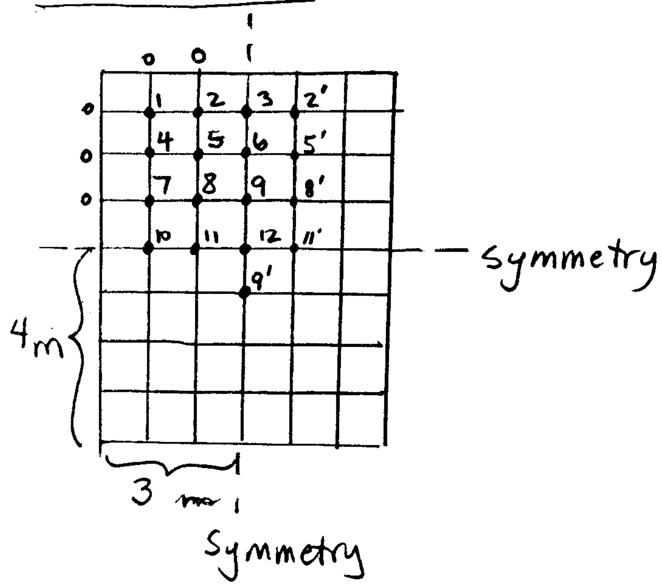
Solve $\nabla^2\phi = -\frac{\rho}{\varepsilon} = -2$

```
c********************************
c This program solves example 2
c using finite difference method
c with symmetry.
c C.Furse  4-13-95
c /grad/furse/fd_sor_sym.f
c********************************
      parameter(imax=3,jmax=4) ! size of rectangular duct
c                                  (with symmetry)
      real phi(imax,jmax) ! potentials
      real b(imax,jmax) ! b vector = q*h^2/eps

c-------------------------------------
c     Force Boundary Value Potentials
c     (For this problem, they are zero everywhere)
c     (Potentials on corners can be any value, because they
c        are not used in calculations.)
c-------------------------------------
      do i=1,imax
         phi(i,1) = 0.   ! top
      enddo
      do j=1,jmax
         phi(1,j) = 0.   ! left
      enddo

c-------------------------------------
c     Define resolution of grid: h (meters)
c-------------------------------------
      side_x = 6. ! meters
      side_y = 8. ! meters
      h = side_x/3. ! resolution of grid (meters)

c-------------------------------------
c     Set up b vector: b=q*h*h/eps
c-------------------------------------
      er = 1.0 ! relative permittivity
      eo = 8.854e-12 ! electric permittivity
      eps = er*eo ! permittivity of medium inside duct
      q = 2.0*eps ! given charge density
      b_constant = -q*h*h/eps

      do i=1,imax
      do j=1,jmax
         b(i,j) = b_constant
      enddo
      enddo

c---------------------------------------
c Set up SOR constants
c---------------------------------------
      ww = 1.0 ! overrelaxation constant (1.<ww<2.)
      maxit = 25 ! maximum # of iterations
      xcharge = 1e-6  ! maximum change in phi vector form k to k+1

c------------------------------------
c Initial guess
c------------------------------------
      do i=2,imax-1
      do j=2,jmax-1
         phi(i,j) = 0.
      enddo
      enddo

c-------------------------------------
c     Find all internal potentials using FD and SOR
c-------------------------------------
50    do k=1,maxit ! iterate for SOR
```



```
         xch = 1e6 ! initialize error measure

         do i=2,imax-1
         do j=2,jmax-1
            Rij = 0.25*(phi(i+1,j) + phi(i-1,j) + phi(i,j+1)
     &           + phi(i,j-1) -4.*phi(i,j) -b(i,j))
            phi_k = phi(i,j)
            phi(i,j) = phi_k + ww*Rij ! (k+1) iteration
            xch = min(xch,abs(phi(i,j)-phi_k))
         enddo
         enddo

c        --Force symmetry planes--
         do j=1,jmax ! right side
            phi(imax,j) = phi(imax-1,j)
         enddo
         do i=1,imax ! bottom
            phi(i,jmax) = phi(i,jmax-2)
         enddo

c        --Check to see if finished--
         if (xch.le.xchange) then ! finished
            write(6,*) 'Finished:'
            write(6,*) '  maximum change in phi=',xch
            write(6,*) '  number of iterations =',k
            goto 100
         endif

      enddo

c------------------------------
c What to do if max # of iterations exceeded:
c------------------------------
      write(6,*) 'Maximum iterations reached:'
      write(6,*) '  maximum change in phi=',xch
      write(6,*) '  number of iterations =',k
      write(6,*) 'How many more iterations ? (0 to quit)'
      read(5,*) iter
      if (iter.gt.0) then
         maxit = maxit + iter
         goto 50
      endif

c------------------------------
c Finish
c------------------------------
100   do i=2,imax-1
      do j=2,jmax-1
         write(6,*) 'i,j,phi=',i,j,phi(i,j)
      enddo
      enddo

      end
```

FINER MESH $h = 1$ m



$\phi_2 = \phi_2'$

$\phi_5 = \phi_5'$

$\phi_8 = \phi_8'$

$\phi_{11} = \phi_{11}'$

$\phi_9 = \phi_9'$

$\phi_8 = \phi_8'$

$\phi_7 = \phi_7'$

MATRIX FOR POISSON'S EQ$^{\underline{n}}$ $\nabla^2 \phi = -\frac{q_v}{\varepsilon} = -2$

$$\begin{bmatrix}
-4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 \\
1 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -2 \\
0 & 2 & -4 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -2 \\
1 & 0 & 0 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & -2 \\
0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & -2 \\
0 & 0 & 1 & 0 & 2 & -4 & 0 & 0 & 1 & 0 & 0 & 0 & -2 \\
0 & 0 & 0 & 1 & 0 & 0 & -4 & 1 & 0 & 1 & 0 & 0 & -2 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 & -2 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 2 & -4 & 0 & 0 & 1 & -2 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & -4 & 1 & 0 & -2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 1 & -4 & 1 & -2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 2 & -4 & -2
\end{bmatrix}$$

Node 3:

$\phi_2 + \phi_2' + 0 + \phi_6 - 4\phi_3 = -2$

$2\phi_2 + \phi_6 - 4\phi_3 = -2$

Node 12:

$\phi_9 + \phi_{9'} + \phi_{11} + \phi_{11}' - 4\phi_{12} = -2$

$2\phi_9 + 2\phi_{11} - 4\phi_{12} = -2$

$-\frac{q_v}{\varepsilon} h^2$

Solution using Gauss Elimination:

$$\bar{\phi} = \begin{bmatrix} 2.0428 \\ 3.04767 \\ 3.35365 \\ 3.12354 \\ 4.79423 \\ 5.31924 \\ 3.65712 \\ 5.68647 \\ 6.33486 \\ 3.81848 \\ 5.95967 \\ 6.64727 \end{bmatrix}$$

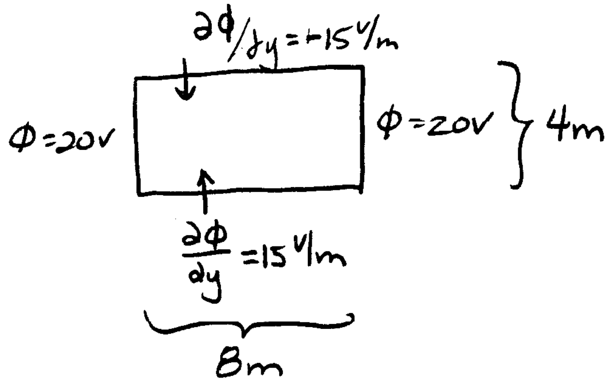
Solved using SOR

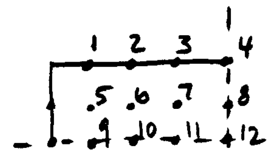
$x^{(0)} = \bar{0}$ $\omega = 1$ 33 iterations

$\sum x^{(k+1)} - x^{(k)} \le 0.01$

```
x=
2.04147
3.0456
3.35151
3.12133
4.79081
5.31571
3.65454
5.68247
6.33073
3.81598
5.9558
6.64326
```

FINITE DIFFERENCE (EXAMPLE 3)





$h = 1$m

GRID NUMBERING

Solve $\nabla^2 \phi = -\rho_v/\varepsilon$

$\rho_v = 5\times10^{-12}$ C/m²

$\varepsilon_r = 1.0$

MATRIX.IN $\bar{A} | \bar{b}$

```
12 12<-size
-4 1 0 0 2 0 0 0 0 0 0 0 9.4353
1 -4 1 0 0 2 0 0 0 0 0 0 29.4353
0 1 -4 1 -0 0 2 0 0 0 0 0 29.4353
0 0 2 -4 0 0 0 2 0 0 0 0 29.4353
1 0 0 0 -4 1 0 0 1 0 0 0 -20.5647
0 1 0 0 1 -4 1 0 0 1 0 0 -.5647
0 0 1 0 0 1 -4 1 0 0 1 0 -.5647
0 0 0 1 0 0 2 -4 0 0 0 1 -.5647
0 0 0 0 2 0 0 0 -4 1 0 0 -20.5647
0 0 0 0 0 2 0 0 1 -4 1 0 -.5647
0 0 0 0 0 0 2 0 0 1 -4 1 -.5647
0 0 0 0 0 0 0 2 0 0 2 -4 -.5647
```

$\bar{b}$

LINPACK: x:	GAUSS ELIMINATION x vector:
-11.3165	-11.3165
-30.3903	-30.3903
-41.2253	-41.2254
-44.7768	-44.7768
-2.72025	-2.72025
-19.792	-19.792
-30.1495	-30.1495
-33.6106	-33.6106
-.337145	-.337148
-16.4728	-16.4728
-26.5346	-26.5346
-29.9314	-29.9315

SOR 1

```
x=
-11.3108
-30.3801
-41.2125
-44.7634
-2.71472
-19.7822
-30.1371
-33.5978
-.331822
-16.4633
-26.5227
-29.9191
```

$\omega = 1.0$

$k = 100$

$\Delta X_{max} = 9.85e\text{-}3$

SOR 2

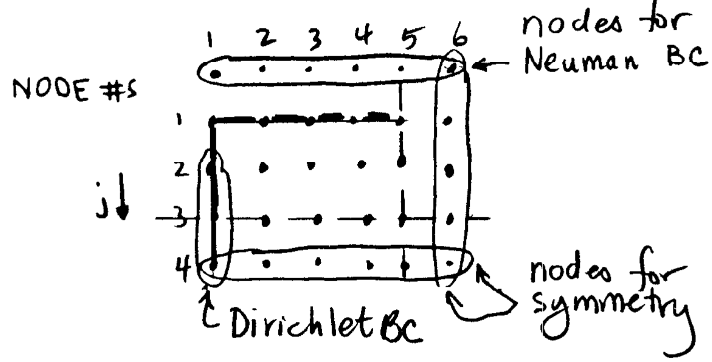
```
x=
-11.3165
-30.3903
-41.2253
-44.7768
-2.72025
-19.792
-30.1495
-33.6106
-.337143
-16.4728
-26.5346
-29.9314
```

$\omega = 1.5$

$k = 49$

$\Delta X_{max} = 5.8e\text{-}5$

FD-SOR (NO MATRIX)



node #	i,j,phi		
1	2	2	-11.3158
5	2	3	-2.71956
9	2	4	-.336462
2	3	2	-30.389
6	3	3	-19.7908
10	3	4	-16.4716
3	4	2	-41.2237
7	4	3	-30.148
11	4	4	-26.5331
4	5	2	-44.7751
8	5	3	-33.609
12	5	4	-29.9298

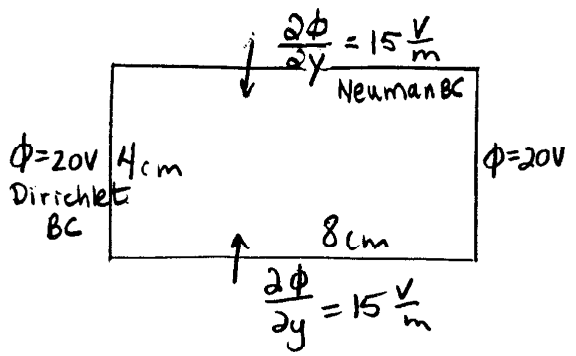
$\omega = 1.5$

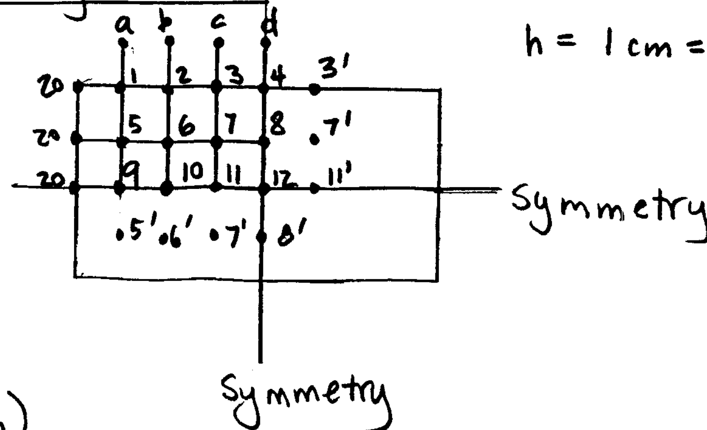
$k = 74$

$\Delta X_{max} = 9.3e\text{-}5$

Example: Neuman + Dirichlet Boundary Conditions

$h = 1\text{ cm} = 0.01\text{m}$





Poisson's Eq^{n} (Difference Form)

$$\phi_{i-1,j} + \phi_{i+1,j} + \phi_{ij-1} + \phi_{ij+1} - 4\phi_{ij} = -\frac{\rho h^2}{\varepsilon}$$

$\rho = 5\times10^{-12}\ C/m^2$

$\varepsilon = \varepsilon_0 = 8.854\times10^{-12}\ \frac{F}{m}$ $\frac{C^2}{Nm^2}$

$h = 1m$

Node 1: $\left.\frac{\partial\phi}{\partial y}\right|_{node1} = 15 = \frac{\phi_5 - \phi_a}{2h} \rightarrow \phi_a = \phi_5 - 30$

$$\phi_a + \phi_2 + \phi_5 + 20 - 4\phi_1 = -\frac{\rho h^2}{\varepsilon}$$

$$-4\phi_1 + \phi_2 + 2\phi_5 = -\frac{\rho h^2}{\varepsilon} - 20 + 30$$

Node 4: $\phi_d = \phi_8 - 0.3$

$$\phi_d + \phi_3 + \phi_8 + \underbrace{\phi_3'}_{=\phi_3} - 4\phi_4 = -\frac{\rho h^2}{\varepsilon}$$

$$-4\phi_4 + 2\phi_3 + 2\phi_8 = -\frac{\rho h^2}{\varepsilon} + 30$$

Node 9:

$$\phi_5 + 20 + \phi_{10} + \underbrace{\phi_5'}_{=\phi_5} - 4\phi_9 = -\frac{\rho h^2}{\varepsilon}$$

$$-4\phi_9 + 2\phi_5 + \phi_{10} = -\frac{\rho h^2}{\varepsilon} - 20$$

From symmetry

From Neuman BC

	1	2	3	4	5	6	7	8	9	10	11	12	13		
1	-4	1			2								R	-20	+30
2	1	-4	1			2							R		+30
3		1	-4	1			2						R		+30
4			2	-4	0			2					R		+30
5	1				-4	1			1				R	-20	
6		1			1	-4	1			1			R		
7			1			1	-4	1			1		R		
8				1			2	-4	0			1	R		
9					2			0	-4	1			R	-20	
10						2			1	-4	1		R		
11							2			1	-4	1	R		
12								2			2	-4	R		

$\left(\frac{\partial\phi}{\partial n}\right)2h$

From Dirichlet BC

From Poisson

Let $R = -\frac{\rho h^2}{\varepsilon_0} = -0.5647$

```fortran
c********************************
c This program solves example 3
c using finite difference method
c with symmetry.
c C.Furse  4-13-95
c /grad/furse/fd_sor_sym.f
c********************************
      parameter(imax=6,jmax=5) ! size of rectangular duct
c                                (with symmetry)
      real phi(imax,jmax) ! potentials
      real b(imax,jmax) ! b vector = q*h^2/eps

c-------------------------------------
c    Force Dirichlet Boundary Conditions
c     (Potentials on corners can be any value, because they
c        are not used in calculations.)
c-------------------------------------
      do j=1,jmax
         phi(1,j) = 20 ! volts
      enddo

c-------------------------------------
c     Define resolution of grid: h (meters)
c-------------------------------------
      h = 1 ! meters

c-------------------------------------
c     Set up b vector: b=q*h*h/eps
c-------------------------------------
      er = 1.0 ! relative permittivity
      eo = 8.854e-12 ! electric permittivity
      eps = er*eo ! permittivity of medium inside duct
      q = 5e-12 ! given charge density (C/m^2)
      b_constant = -q*h*h/eps

      do i=1,imax
      do j=1,jmax
         b(i,j) = b_constant
      enddo
      enddo

c---------------------------------------
c Set up SOR constants
c---------------------------------------
      ww = 1.5 ! overrelaxation constant (1.<ww<2.)
      maxit = 50 ! maximum # of iterations
      xchange = 1e-4  ! maximum change in phi vector form k to k+1

c-------------------------------------
c Initial guess
c-------------------------------------
      do i=2,imax-1
      do j=2,jmax-1
         phi(i,j) = 0.
      enddo
      enddo

c-------------------------------------
c     Find all internal potentials using FD and SOR
c-------------------------------------
50    do k=1,maxit ! iterate for SOR

         xch = 1e6 ! initialize error measure

         do i=2,imax-1
         do j=2,jmax-1
            Rij  = 0.25*(phi(i-1,j) + phi(i-1,j) + phi(i,j+1)
     &             + phi(i,j-1) -4.*phi(i,j) -b(i,j))
```



```fortran
            phi_k = phi(i,j)
            phi(i,j) = phi_k + ww*Rij ! (k+1) iteration
            xch = min(xch,abs(phi(i,j)-phi_k))
         enddo
         enddo

c     --Force Neuman boundary conditions--
      BC = -15 ! V/m (more efficient to set this outside k loop)
      do i=1,imax
         phi(i,1) = BC*2.*h + phi(i,3)
      enddo

c        --Force symmetry planes--
         do j=1,jmax ! right side
            phi(imax,j) = phi(imax-2,j)
         enddo
         do i=1,imax ! bottom
            phi(i,jmax) = phi(i,jmax-2)
         enddo

c        --Check to see if finished--
         if (xch.le.xchange) then ! finished
            write(6,*) 'Finished:'
            write(6,*) '  maximum change in phi=',xch
            write(6,*) '  number of iterations =',k
            goto 100
         endif

      enddo

c------------------------------
c What to do if max # of iterations exceeded:
c------------------------------
      write(6,*) 'Maximum iterations reached:'
      write(6,*) '  maximum change in phi=',xch
      write(6,*) '  number of iterations =',k
      write(6,*) 'How many more iterations ? (0 to quit)'
      read(5,*) iter
      if (iter.gt.0) then
         maxit = maxit + iter
         goto 50
      endif

c------------------------------
c Finish
c------------------------------
100   do i=2,imax-1
      do j=2,jmax-1
         write(6,*) 'i,j,phi=',i,j,phi(i,j)
      enddo
      enddo

      end
```