```
  1   !Stick2It.F90
  2   Program Stick2It
  3   !2006.03.10.1625cst JMS -A700:GeForce3/NT4.0:Sp6/APF9.0/f90gl1.2.1:glut3.7
  4   !Solution aid for 9x9 puzzles.
  5   !This program is not warranted for any particular purpose.
  6   !Use this program at your own risk.
  7   !Compiled using: Absoft Pro Fortran 9.0
  8   !------------------
  9    implicit none
  10    integer*4::nSteps,nStepSto(81)
  11    integer*1::i,i1,i2,iu,j,j1,j2,ju,k,k1,M,N,Nu
  12    integer*1::L(9,9,0:9)
  13    integer*1::Nin(9,9),Nout(9,9)
  14    character::cType*22
  15    integer*4::nUnResolved(0:81)
  16    integer*4::nTry2(81),nT,Lsto(9,9,0:9,81)
  17  !------------------
  18  ! data((Nin(i,j),j=1,9),i=1,9) /  &
  19  !            6,0,3,0,0,0,2,0,4    &
  20  !           ,0,0,0,9,0,2,0,0,0    &
  21  !           ,0,1,0,0,0,0,0,6,0    &
  22  !           ,0,0,1,6,8,9,7,0,0    &
  23  !           ,0,0,0,0,0,0,0,0,0    &
  24  !           ,0,0,6,2,4,7,5,0,0    &
  25  !           ,0,9,0,0,0,0,0,5,0    &
  26  !           ,0,0,0,5,0,8,0,0,0    &
  27  !           ,5,0,4,0,0,0,6,0,1  /
  28  !     Puzzle #240 in: "Su Doku for Dummies"
  29  !           rated: "Diabolical"
  30
  31     write(6,"(' Stick2It.exe vsn. 0.7 - Jeff Setterholm  2006.03.10')")
  32     write(6,"(' This program is not warranted for any particular purpose.')")
  33     write(6,"('               Use this program at your own risk.')")
  34     write(6,"('  Input data: Stick2It.dat 9 rows of 9 integers [1,9]')")
  35
  36     open(unit=14,file='Stick2It.dat',action='read')
  37       do i=1,9 ;read(14,*) (Nin(i,j),j=1,9) ;end do
  38     close(14)
  39
  40     do i=1,9 ;write(6,"(1x,9i3)") Nin(i,1:9) ;enddo
  41
  42     open(unit=15,file='Stick2It.out',action='write')
  43     write(15,"('Stick2It.exe vsn. 0.7 - Jeff Setterholm  2006.03.10')")
  44     write(15,"('This program is not warranted for any particular purpose.')")
  45     write(15,"('               Use this program at your own risk.')")
  46     write(15,"(' Input data: Stick2It.dat 9 rows of 9 integers [1,9]')")
  47     write(15,*)
  48     do i=1,9 ;write(15,"(9i3)") Nin(i,1:9) ;enddo
  49     write(15,*)
  50     write(15,"(' Solution sequence: Stick2It.out')")
  51
  52     nTry2=0 ;nT=0 ;Lsto=0
  53     nSteps=0
  54     Nout=Nin
  55                   L       =0
  56     do i=1,9 ;do j=1,9 ;N=Nin(i,j)
  57       if(N.eq.0) then
  58         Do k=1,9 ;L(i,j,k)=k ;enddo
  59       endif
  60       if(N.gt.0) then
  61         L(i,j,N)=0  !=N
  62         L(i,j,0)=N
  63       endif
  64     enddo ;enddo
  65     do i=1,9 ;do j=1,9 ;N=Nin(i,j)
  66       if(N.ne.0) then
  67         do i1=1,9 ;if(i1.ne.i) L(i1,j,N)=0 ;enddo
  68         do j1=1,9 ;if(j1.ne.j) L(i,j1,N)=0 ;enddo
  69         i2=i-mod(i-1,3) ;j2=j-mod(j-1,3)
  70         do i1=i2,i2+2 ;do j1=j2,j2+2
```

```
 71           if((i1.ne.i).or.(j1.ne.j)) L(i1,j1,N)=0
 72         enddo ;enddo
 73       endif
 74     enddo ;enddo
 75     cType='Initial Puzzle Values '
 76     goto 15
 77
 78  10 continue
 79        L(iu,ju,1:9) =  0
 80        L(iu,ju,  0) = Nu
 81        L(iu,ju, Nu) =  0  != Nu
 82     Nout(iu,ju    ) = Nu
 83     do i1=1,9 ;if(i1.ne.iu) L(i1,ju,Nu)=0 ;enddo
 84     do j1=1,9 ;if(j1.ne.ju) L(iu,j1,Nu)=0 ;enddo
 85     i2=iu-mod(iu-1,3) ;j2=ju-mod(ju-1,3)
 86     do i1=i2,i2+2 ;do j1=j2,j2+2
 87       if((i1.ne.iu).or.(j1.ne.ju)) L(i1,j1,Nu)=0
 88     enddo ;enddo
 89  15 nUnResolved(nT)=0
 90     do i=1,9 ;do j=1,9 ;M=L(i,j,0)
 91       if(M.le.0) nUnResolved(nT)=nUnResolved(nT)+1
 92     enddo ;enddo
 93     do i=1,9 ;do j=1,9 ;M=L(i,j,0)    !Tidy up
 94       if(M.le.0) then; M=0
 95         do k=1,9 ;
 96           if(L(i,j,k).gt.0) then
 97             M=M-1
 98           endif
 99         enddo
100         L(i,j,0)=M
101         if(M.eq. 0) then   !Invalid result
102           cType='Intended operation    '
103           write( 6,"( 1x,'i=',i1,' j=',i1,' N=',i1,' nStep=',i2,2x,a22)")  &
104                    iu,ju,Nu,nSteps,cType
105           write(15,"(/1x,'i=',i1,' j=',i1,' N=',i1,' nStep=',i2,2x,a22)")  &
106                    iu,ju,Nu,nSteps,cType
107           cType='Resulting error       '
108           iu=i ;ju=j; Nu=0
109           write( 6,"( 1x,'i=',i1,' j=',i1,' N=',i1,' nStep=',i2,2x,a22)")  &
110                    iu,ju,Nu,nSteps,cType
111           write(15,"(/1x,'i=',i1,' j=',i1,' N=',i1,' nStep=',i2,2x,a22)")  &
112                    iu,ju,Nu,nSteps,cType
113           if(nT.eq.0) goto 45
114           goto 35
115         endif
116       endif
117     enddo ;enddo
118     nSteps=nSteps+1
119     call ShowL(iu,ju,L,Nin,Nu,nSteps,cType)
120     if(nUnResolved(nT).eq.0) then
121       goto 45
122     endif
123
124  20 continue
125     iu=0 ;ju=0 ;Nu=0
126     cType='Single Choice         '
127     do i=1,9 ;do j=1,9 ;M=L(i,j,0)
128       if(M.le.0) then; M=0
129         do k=1,9 ;
130           if(L(i,j,k).gt.0) then
131             M=M-1 ;iu=i ;ju=j ;Nu=k
132           endif
133         enddo
134         L(i,j,0)=M
135         if(M.eq.-1) goto 10
136       endif
137     enddo ;enddo
138
139     cType='Single Choice - Row   '
140     do i=1,9 ;do k=1,9 ;M=0
```

```
141       do j=1,9
142         if((L(i,j,0).lt.0).and.(L(i,j,k).gt.0)) then
143           M=M-1 ;iu=i ;ju=j ;Nu=k
144         endif
145       end do
146       if(M.eq.-1) goto 10
147     enddo ;enddo
148
149     cType='Single Choice - Column'
150     do j=1,9 ;do k=1,9 ;M=0
151       do i=1,9
152         if((L(i,j,0).lt.0).and.(L(i,j,k).gt.0)) then
153           M=M-1 ;iu=i ;ju=j ;Nu=k
154         endif
155       end do
156       if(M.eq.-1) goto 10
157     enddo ;enddo
158
159     cType='Single Choice -  Group'
160     do i=1,9,3 ;do j=1,9,3
161       do k=1,9 ;M=0
162         do i1=i,i+2 ;do j1=j,j+2 ;
163           if((L(i1,j1,0).lt.0).and.(L(i1,j1,k).gt.0)) then
164             M=M-1 ;iu=i1 ;ju=j1 ;Nu=k
165           endif
166         end do ;end do
167         if(M.eq.-1) goto 10
168       end do
169     enddo ;enddo
170
171     nT=nT+1
172 30  cType='DualChoice-Trial&Error'
173     if(nTry2(nT).eq.0) then
174       Lsto(1:9,1:9,0:9,nT)=L
175       nStepSto(nT)=nSteps
176     endif
177 35  nTry2(nT)=nTry2(nT)+1 ;k=0
178     L=Lsto(1:9,1:9,0:9,nT)
179     nSteps=nStepSto(nT)
180     do i=1,9 ;do j=1,9 ;M=L(i,j,0)
181       if(M.eq.-2) then
182 !       (M.lt.0)          ...could be used & would catch > dual-choice.
183         do k1=1,9
184           if(L(i,j,k1).gt.0)then
185             k=k+1
186             if(k.eq.nTry2(nT)) then
187               iu=i ;ju=j ;Nu=k1
188               write(cType,"(i2,':T&E: i=',i1,' j=',i1,' N=',i1)") nT,i,j,k1
189               goto 10      ! ^ this is the nesting of the trial & error test.
190             endif
191           endif
192         enddo
193       endif
194     enddo ;end do
195     nTry2(nT)=0
196     Lsto(1:9,1:9,0:9,nT)=0
197     nT=nT-1
198     if(nT.le.0) then
199 !     cType='Trial&Error-insuffic. '
200       nSteps=nStepSto(nT)
201       goto 45
202     endif
203     L=Lsto(1:9,1:9,0:9,nT)
204     nSteps=nStepSto(nT)
205     goto 30
206
207 45  iu=0 ;ju=0; Nu=0
208     cType='All values resolved   '
209     if(nUnResolved(nT).gt.0)  &
210     write(cType,"('UnResolved: ',i3,7x)") nUnResolved(nT)
```

```
 211
 212 50 call ShowL(iU,jU,L,Nin,Nu,nSteps,cType)
 213    write(15,*)
 214    do i=1,9 ;write(15,"(9i3)") L(i,1:9,0) ;enddo
 215    close(15)
 216    write(6,"('  Solution sequence: Stick2It.out')")
 217    do i=1,9 ;write(6,"(1x,9i3)") L(i,1:9,0) ;enddo
 218    write(6,*)
 219    pause 'Press "enter" to exit.'
 220 End Program Stick2It
 221 !-------------------------------------------------------------------------------7 9
 222 Subroutine ShowL(iu,ju,L,Nin,Nu,nSteps,cType)
 223 !2006.03.07.0945cst JMS -A700:GeForce3/NT4.0:Sp6/APF9.0/f90gl1.2.1:glut3.7
 224 !Printout for the solution to the 9x9 puzzle
 225 !------------------
 226  implicit none
 227   integer*1::i,iu,j,ju,k,Nu
 228   integer*4::nSteps
 229   character::cType*22
 230   integer*1::Nin(9,9)
 231   integer*1::L(9,9,0:9)
 232 !------------------
 233    write( 6,"( 1x,'i=',i1,' j=',i1,' N=',i1,' nStep=',i2,2x,a22)")  &
 234                 iu,ju,Nu,nSteps,cType
 235    write(15,"(/1x,'i=',i1,' j=',i1,' N=',i1,' nStep=',i2,2x,a22)")  &
 236                 iu,ju,Nu,nSteps,cType
 237    do i=1,9
 238    write(15,"('|',3(i2,1x,9i1.0,'/',i2,1x,9i1.0,'/',i2,1x,9i1.0,'|'))") &
 239                      ((L(i,j,k),k=0,9),j=1,9)
 240    if((i.eq.3).or.(i.eq.6)) &
 241    write(15,"(118('-'))")
 242    end do
 243   return
 244 End Subroutine ShowL
 245 !-------------------------------------------------------------------------------7 9
 246
```