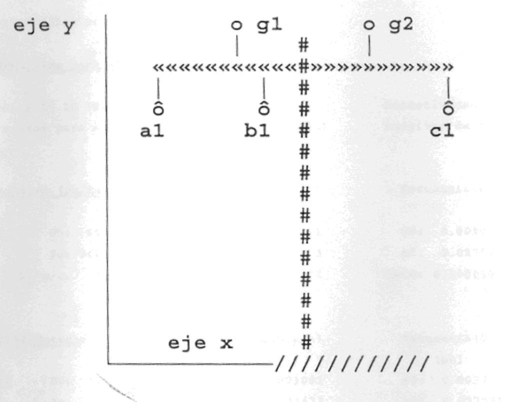
LIMPED – Bare Line Impedance calculator

by Carlos Mauricio Beltran, BS Electrical Engineering 1991, Tulsa University, OK, USA.

LIMPv3 (August 2011) calculates the positive and zero sequence impedances of bare three phase lines which have one conductor per phase and up to two guard or neutral conductors. I have verified many of the results, but I cannot give any guarantees. Use the program at your own risk.

This program is freeware; you may use it and share it freely. You may not modify or copy any portion of the program or its subroutines in whole or in part into any platform for the purpose of selling it. I reserve those rights; I am the sole intellectual proprietor of the source code. If you find the program useful please e-mail me at: **carlosmbm@msn.com** or at **carlosmbm@cre.com.bo**

The structure could be for a high voltage transmission or medium voltage distribution line.



Coordenadas de Conductores e Hilos de Guarda:

Conductor	x[cm]	y[cm]
a1	000.00	1276.00
b1	366.00	1276.00
c1	732.00	1276.00
Hilos de Guarda	**x[cm]**	**y[cm]**
g1	183.00	1466.00
g2	549.00	1466.00

Figure 1. 69 kV structure with two guard or neutral conductors

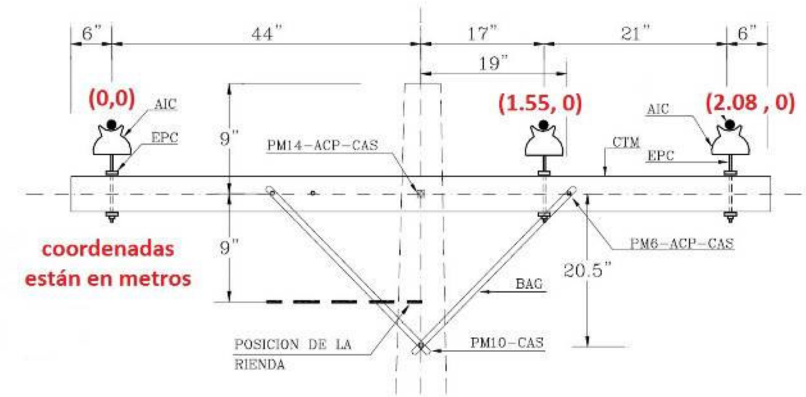


Figure 2. 10.5 kV structure with phase conductors only (has no neutral conductor)

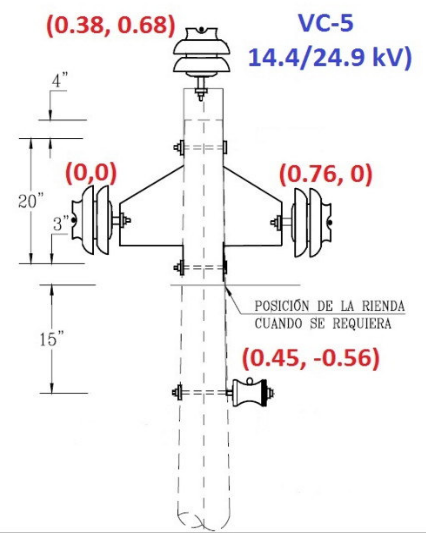


Figure 3. 24.9 kV structure with one neutral conductor

LIMPED – Bare Line Impedance calculator

by Carlos Mauricio Beltran, BS Electrical Engineering 1991, Tulsa University, OK, USA.

LIMPv3 uses the SI system, and is set for an earth resistivity of 100 ohm meters, and for 50Hz electrical systems; however you can easily change the earth resistivity and frequency by modifying the ***Dig*** and **Rg** variables (in the DATA folder of the program) to the values you need using the following formulas.

Dig=25.7*[(rho/freq)]^0.25

Rg = 986.95E-6*freq

LIMPv3 has been programmed with common ACSR conductors and support structures used by CRE(Rural Electric Cooperative, Santa Cruz – Bolivia).

You may modify the subroutines to include the conductors and structures you desire, nonetheless, the program also allows for manual entry of conductor parameters as well as (x,y) coordinates. The program does not compute shunt capacitance of the line.

The following table of ACSR conductors will be useful if you want to customize LIMPv3. You may of course use any other conductors data.

code	dc_ohm/km	gmr_cm
50	0.6535	0.41382
BITTERN	0.0443	1.3564
BLUEBIRD	0.0262	1.7922
BLUEJAY	0.0509	1.2680
BOBOLINK	0.0394	1.4387
BUNTING	0.0472	1.3137
CANARY	0.0623	1.1948
CARDINAL	0.0587	1.2314
CHICKADEE	0.1417	0.7315
CHUKAR	0.0318	1.6276
CONDOR	0.0705	1.1217
COOT	0.0712	1.0211
CORNCRAKE	0.0591	1.1521
CUCKOO	0.0705	1.1003
CURLEW	0.0541	1.2802
DIPPER	0.0417	1.3990
DOVE	0.1001	0.9540
DRAKE	0.0702	1.1430
EAGLE	0.0984	0.9997
EGRET	0.0873	1.0698
Extra High Steel	2.61	0.2873
FALCON	0.0354	1.5880
FINCH	0.0505	1.3289
FLAMINGO	0.0840	1.0211

code	dc_ohm/km	gmr_cm
FLICKER	0.1175	0.8626
GRACKLE	0.0472	1.3746
GROSBEAK	0.0876	1.0211
HAWK	0.1171	0.8839
HEN	0.1161	0.9266
IBIS	0.1404	0.8077
JOREA	0.0226	1.8928
KINGBIRD	0.0883	0.9174
KIWI	0.0262	1.7374
LAPWING	0.0354	1.5179
LARK	0.1394	0.8443
LINNET	0.1660	0.7437
MALLARD	0.0699	1.1948
MARTIN	0.0424	1.1464
MERLIN	0.1673	0.6736
MOCKINGBIRD	0.0279	1.6855
OPGW	0.2021	0.663
ORIOLE	0.1647	0.7772
ORTOLAN	0.0548	1.2222
OSPREY	0.1014	0.8656
PARAKEET	0.1007	0.9327
PARTRIDGE	0.2090	0.6614
PEACOCK	0.0925	0.9723
PELICAN	0.1181	0.8016
PENGUIN	0.2608	0.5547
PHEASANT	0.0443	1.4204
PIGEON	0.3291	0.4938
PLOVER	0.0394	1.5088
QUAIL	0.4150	0.4420
RAIL	0.0591	1.1735
RAVEN	0.5226	0.3932
REDBIRD	0.0587	1.2070
REDWING	0.0774	1.1339
ROBIN	0.6598	0.3505
ROOK	0.0879	0.9967
RUDDY	0.0627	1.1400
SKYLARK	0.0443	1.3015
SPARATE	0.8222	0.3200
SPARROW	0.8314	0.3109
STARLING	0.0781	1.0820
SWAN	1.3228	0.2469
SWANATE	1.3087	0.2530
SWIFT	0.0876	0.9144
TERN	0.0709	1.0729
THRASHER	0.0246	1.8288
TOWHEE	0.0591	1.1918
TURKEY	2.1060	0.1951
WAXWING	0.2113	0.6005

LIMPED – Bare Line Impedance calculator

by Carlos Mauricio Beltran, BS Electrical Engineering 1991, Tulsa University, OK, USA.

Just load the LIMPED text file and save it in your HP (ASCII, 9600 baud, parity = none 0, cksum 3, translate 1)

Now let´s work out an example for the following structure. The coordinates of the Phase conductors and Neutral conductor are as shown:

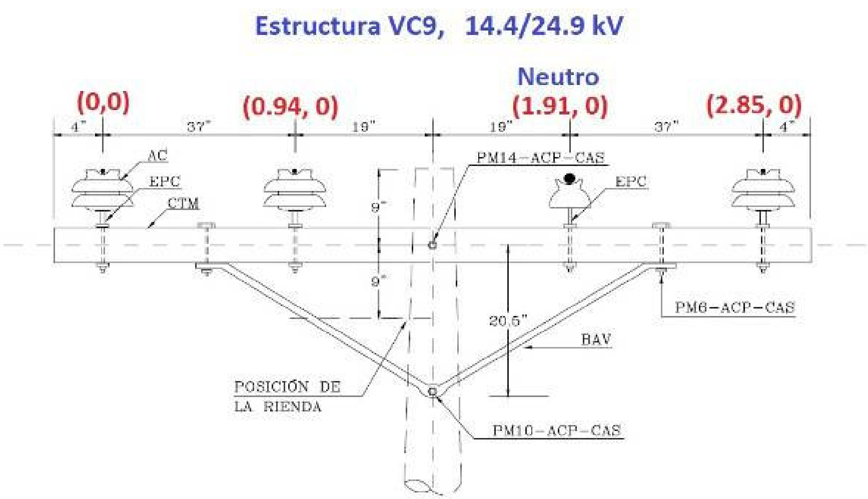
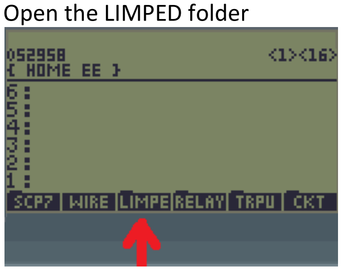


Figure 4

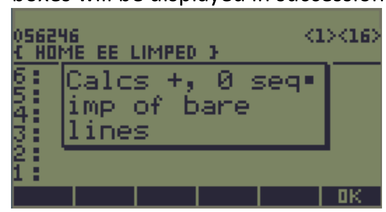
The phase conductors are ACSR 4/0 and the neutral conductor is ACSR 1/0.
Program assumes 100 ohm meter earth resistivity and 50Hz frequency.

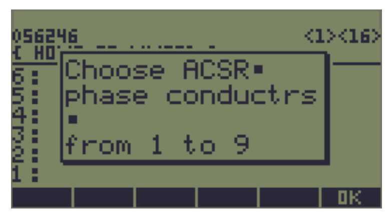
1. Open the LIMPED folder



2. Press on the RUN subroutine

3. The following three message boxes will be displayed in succession

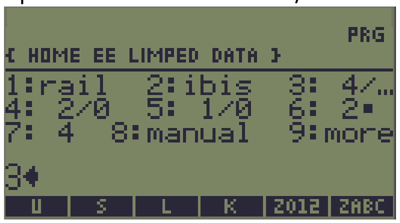




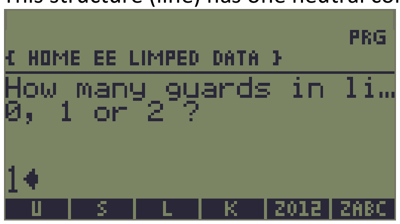
LIMPED – Bare Line Impedance calculator

by Carlos Mauricio Beltran, BS Electrical Engineering 1991, Tulsa University, OK, USA.

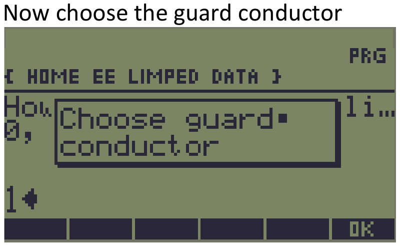
4. Option 3 selects conductor 4/0



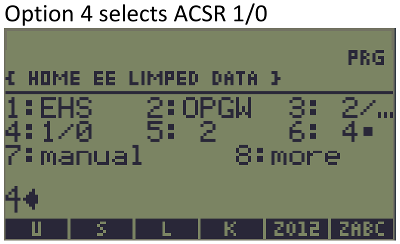
5. This structure (line) has one neutral conductor



6. Now choose the guard conductor

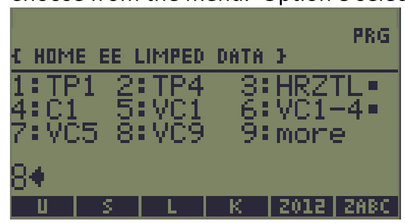


Option 4 selects ACSR 1/0



7. Afterwards, the following message box is displayed

8. Choose from the menu. Option 8 selects the structure of Figure 4.



In subroutine SUB2 you can see the coordinates of the conductors:

```
'S==8' THEN 99 98 @VC9, 25 kV structure w/neutral@
1.91 0 0 0 .94 0 2.85 0
```

Where: 99 98 are dummy (x, y) coordinates for the nonexistent second neutral conductor.

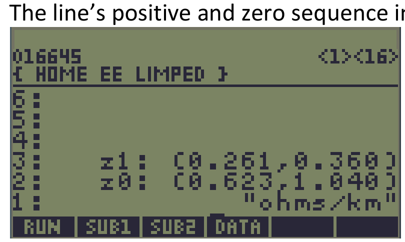
1.91 0 are the (x, y) coordinates of the neutral conductor.

0 0 are the (x, y) coordinates of phase A conductor.

.94 0 are the (x, y) coordinates of phase B conductor.

2.85 0 are the (x, y) coordinates of phase C conductor.

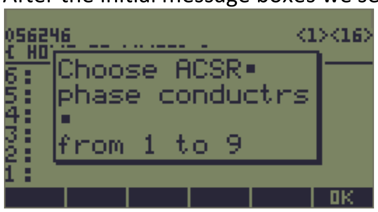
9. The line's positive and zero sequence impedances are:

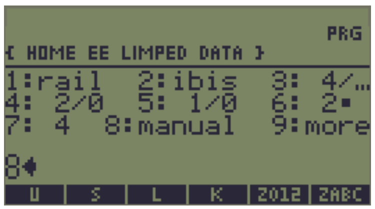


Now let's do another example in which we must enter the data manually.

We'll use the structure shown in **Figue 1**. Phase conductors are HAWK and guard conductors are EHS (Extra High Strength steel 5/16").

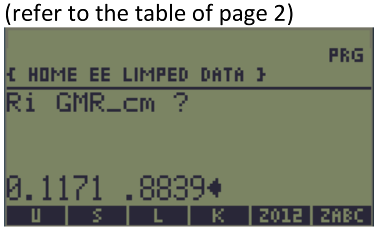
10. After the initial message boxes we select the phase conductor





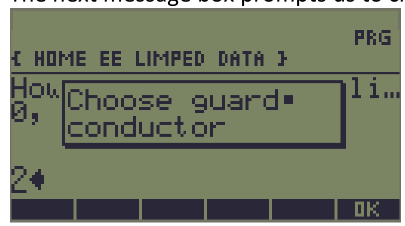
By the way, option 9 is not available (yet).

11. Now enter the DC resistance and GMR of the HAWK conductor (refer to the table of page 2)

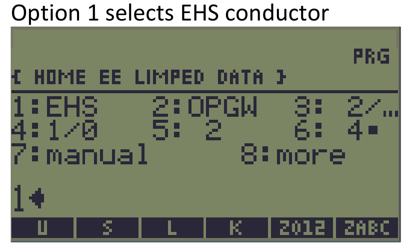


12. This line has two guard (or neutral) conductors

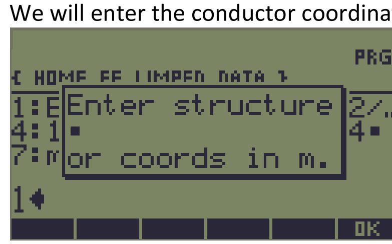
13. The next message box prompts us to choose the guard conductor



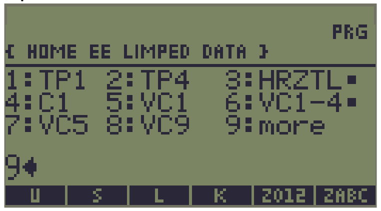
Option 1 selects EHS conductor



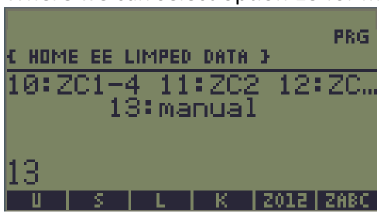
14. We will enter the conductor coordinates manually



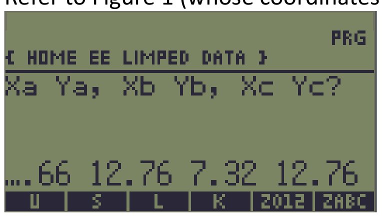
15. Option 9 will lead to an additional menu



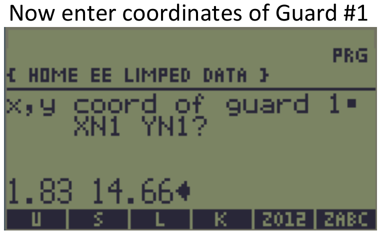
Where we can select option 13 for manual data entry



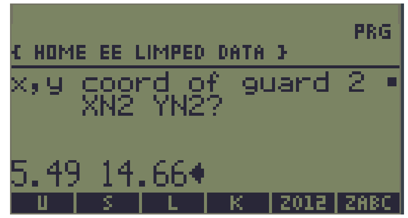
16. Make sure you enter the phase coordinates in METERS!
Refer to Figure 1 (whose coordinates are in centimeters)



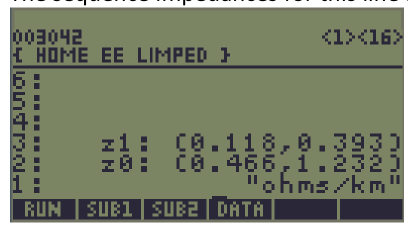
17. Now enter coordinates of Guard #1



18. And also coordinates for Guard #2



19. The sequence impedances for this line are:



I hope you find the program useful. I recommend you customize to fit your company's conductors and structures (and system frequency and earth resistivity).

Look for my Power Network Short Circuit program "SCpower" version 7 at the **hpcalc.org** software archives.

Sincerely,

Carlos Mauricio Beltran
Electrical Engineer
carlosmbm@cre.com.bo
carlosmbm@msn.com

LIMPED – Bare Line Impedance calculator

by Carlos Mauricio Beltran, BS Electrical Engineering 1991, Tulsa University, OK, USA.

```
%%HP: T(3)A(D)F(.);
DIR
  RUN                                 @main program@
    \<<
"Calcs +, 0 seq
imp of bare lines"
MSGBOX
" developed by
Carlos Beltran
carlosmbm@msn.com"      @with Rearth = 100 ohm meter, freq = 50 Hz @
MSGBOX SUB1 SUB2        @executes subroutines SUB 1 & SUB 2 @


DATA          @opens DATA directory to recall matrices for calculations@
ZFF ZFN Znn
INV * ZNF * NEG +
'zabc' STO TINV       @matrix manipulation to calculate z+ & z0 @
zabc * T * 'z012'      @ z012 matrix contains seq.imp. results @
STO z012 DUP
{ 2 2} GET 'z1' \->TAG  @ z1: displays positive seq.impedance @
SWAP { 1 1 } GET          @ z0: displays zero seq.impedance @
'z0' \->TAG UPDIR \>>               @end main program@

  SUB1                         @ begin subroutine SUB1@
    \<< DATA        @opens DATA directory to store variables in it@
"Choose ACSR
phase conductrs        @choose from menu the phase conductors @
from 1 to 9"        @ with option 8 you enter Rdc & GMR manually @
MSGBOX
"1:rail  2:ibis  3: 4/0
4: 2/0  5: 1/0  6: 2
7: 4  8:manual  9:more"    @ option 9 is not yet available @
{ "" { 3 3 } }
INPUT OBJ\-> 'K' STO
 CASE
  'K==1' THEN .0591 1.1735 END    @ R & GMR of rail @
  'K==2' THEN .1404 .8077 END     @ R & GMR of ibis @
  'K==3' THEN .2608 .5547 END      @ R & GMR of 4/0 @
  'K==4' THEN .415 .442  END       @ R & GMR of 2/0 @
  'K==5' THEN .5226 .3932 END      @ R & GMR of 1/0 @
  'K==6' THEN .8314 .3109 END    @ R & GMR of 2 ACSR @
  'K==7' THEN 1.3228 .2469 END   @ R & GMR of 4 ACSR @
  'K==8' THEN                  @ enter data manually @
    "Ri GMR_cm ?" { ""{ 3 3 } }
     INPUT OBJ\->
    END   @ end CASE 8 @
 END  @ end CASE K @
 100 / 'GMRF' STO 'Ri' STO @ converts GMR of phase conductor to meters @
"How many guards in line ?
0, 1 or 2 ?"
{ "" { 1 3 } }
INPUT OBJ\-> 'W' STO
  CASE
   'W==0' THEN END @ line has no guard conductor@
   'W==1 OR W==2' THEN @ case of line with 1 or 2 guards @
     "Choose guard conductor" MSGBOX
      "1:EHS    2:OPGW   3: 2/0
       4:1/0    5: 2     6: 4   @ menu of guard conductor @
       7:manual      8:more"
       { "" { 3 3 } }
       INPUT OBJ\-> 'L' STO
          CASE
           'L==1' THEN 2.61 .2873 END       @ R & GMR of iron steel @
           'L==2' THEN .485 .593 END        @ R & GMR of OPGW @
           'L==3' THEN .415 .442 END        @ R & GMR of 2/0 @
           'L==4' THEN .5226 .3932 END      @ R & GMR of 1/0 @
           'L==5' THEN .8314 .3109 END      @ R & GMR of 4 @
           'L==6' THEN 1.3228 .2469 END     @ R & GMR of 2 @
           'L==7' THEN
              "R_guard, GMR_cm ?"           @ enter data manually @
              { "" { 1 3 } }
               INPUT OBJ\->
            END @ case L==7@
          END @ CASE L @
 100 / 'GMRG' STO 'RGRD'      @ converts GMRG to meters @
STO znn \->NUM 'ZNN'
STO
        END
      END zii \->NUM
'zaa' STO zii \->NUM
'zbb' STO zii \->NUM
'zcc' STO UPDIR
    \>>                           @ end subroutine SUB1 @

  SUB2                            @ begin subroutine SUB2@
    \<< DATA
    0 'S' STO 0 'U' STO
    "Enter structure or coords in m."
    MSGBOX
    "1:TP1 2:TP4   3:HRZTL   @ choose from CRE structures @
    4:C1   5:VC1   6:VC1-4
    7:VC5 8:VC9   9:more"    @ 9 displays more structures @
    { "" { 3 3 } }
    INPUT OBJ\-> 'S' STO
      CASE
      'S==1' THEN 99 99 @vertical TP1 69 kV structure, 1 guard @
```

```
 .15 15.8 .9 14.6 -.9 13.1 -.9 11.6
  END  @ 99 99 means no X or Y coordinate of 2nd guard conductor@

 'S==2' THEN 99 98 @vertical TP4 69 kV structure, 1guard @
  .15 15.8 .9 14.6 .9 13.1 .9 11.6  @guard X Y coords:  0.15m, 15.8 m.@
  END  @ 99 98 means no X or Y coordinate of 2nd guard conductor@

 'S==3'THEN      @ horizontal 115 kV structure, 2 guards @
  5.49 14.66 1.83 14.66 0 12.76 3.66 12.76 7.32 12.76
  END

 'S==4' THEN 99 98 99 98 @ C1, 10 kV structure no guard @
  0 0 1.55 0 2.08 0
  END

 'S==5' THEN 99 98 @ VC1, 25 kV structure w/neutral @
  1.23 -.91 0 0 1.12 .46 2.24 0
  END

 'S==6' THEN 99 98  @ VC1-4, 25 kV structure w/neutral @
  1.5 -.91 0 0 2.06 0 2.85 0
  END

 'S==7' THEN 99 98   @ VC5, 25 kV structure w/neutral @
  .45 -.56 0 0 .38 .68 .76 0
  END

'S==8' THEN 99 98    @ VC9, 25 kV structure w/neutral @
 1.91 0 0 0 .94 0 2.85 0
  END

'S==9' THEN          @displays menu with more structures@
   "10:ZC1-4 11:ZC2 12:ZC9
    13:manual"
    { "" { 3 3 } }
   INPUT OBJ\-> 'U' STO
  END
 END

 CASE
  'U==10' THEN 99 98 @ ZC1-4, 35 kV structure w/neutral@
    1.5 -.91 0 0 2.06 0 2.72 0
   END

  'U==11' THEN 99 98 @ ZC2, 35 kV structure w/neutral@
    1.19 -.91 0 0 1.12 .31 2.24 0
   END

  'U==12' THEN 99 98 @ ZC9, 35 kV structure w/neutral@
    2.85 0 0 0 .94 0 1.91 0
   END

  'U==13' THEN         @ enter phase coords manually @
     "Xa Ya, Xb Yb, Xc Yc?"
     { "" { 1 3 } }
     INPUT OBJ\->
     'Yc' STO 'Xc' STO
     'Yb' STO 'Xb' STO
     'Ya' STO 'Xa' STO
   END
 END

 IF 'U\=/13' THEN @ if you didn't enter data manually, then ..@
  'Yc' STO 'Xc' STO 'Yb' STO @ store coordinates from @
  'Xb' STO 'Ya' STO 'Xa' STO @ menu structures automatically @
  'YN1' STO 'XN1' STO
  'YN2' STO 'XN2' STO
 END

Xb Xa Yb Ya DIST                @ subroutine DISTance @
Dab' STO                 @ distance between conductors A & B @
Xc Xa Yc Ya DIST
'Dac' STO                @ distance between conductors A & C @
Xc Xb Yc Yb DIST
'Dbc' STO                @ distance between conductors B & C @
Dab 'Dij' STO zij
\->NUM DUP
'zab' STO 'zba' STO      @calculates zab & zba which are equal@
Dac 'Dij'
STO zij \->NUM DUP
'zac' STO 'zca' STO      @calculates zac & zca which are equal@
Dbc 'Dij' STO zij
\->NUM DUP
'zbc' STO 'zcb' STO      @calculates zbc & zcb which are equal@
zaa zab
zac zba zbb zbc zca
zcb zcc { 3 3 }
\->ARRY 'ZFF' STO               @creates ZFF 3X3 matrix@

 zig
\->NUM DUP DUP2 'zag'
STO 'zbg' STO 'zcg'     @impedances of phases & guard wrt earth@
STO 'zng' STO                     @they are all equal@

IF 'W==0'                          @IF no guard onductors, THEN ..@
   THEN zag zbg zcg { 3 1 } \->ARRY
     'ZFN' STO        @ZFN matrix for no guard conductors@
```

```
     1 1 1 { 1 3 } \->ARRY
     'ZNF' STO     @ZNF matrix for no guard conductors@

     1 { 1 1 } \->ARRY
     'Znn' STO     @Znn matrix for no guard conductors@

END @ IF no guard conductors@

IF 'U==13 AND (W==1 OR W==2)' @if you entered phase coords manually @
      THEN          @ and structure has guard conductor(s) then ..@
   "x,y coord of guard 1 @ now enter guard coordinates manually @
    XN1 YN1?"
    { "" { 1 3 } }
    INPUT OBJ\-> 'YN1' STO 'XN1' STO
END

IF 'W==1 OR W==2' THEN  @ if structure has guard conductor(s) @
   XN1 Xa YN1 Ya DIST          @ calculate the distance @
  'Dan1' STO            @ from each phase conductor to guard 1 @
   XN1 Xb YN1 Yb DIST
  'Dbn1' STO
   XN1 Xc YN1 Yc DIST
  'Dcn1' STO

   Dan1 'Dij' STO
   zij \->NUM 'zan1' STO
   Dbn1 'Dij' STO
   zij \->NUM 'zbn1' STO
   Dcn1 'Dij' STO
   zij \->NUM 'zcn1' STO
END

IF 'W==1' THEN zan1 zag
    zbn1 zbg zcn1 zcg {3 2 }
    \->ARRY 'ZFN' STO     @ZFN matrix for 1 guard conductor@

    zan1 zbn1 zcn1
    1 1 1 { 2 3 } \->ARRY
    'ZNF' STO              @ZNF matrix for 1 guard conductor@

    ZNN zng 1 1 { 2 2 } \->ARRY
    'Znn' STO            @Znn matrix for 1 guard conductor@
END

IF 'U==13 AND W==2' THEN   @if you entered phase coords manually @
  "x,y coord of guard 2 @ and structure has TWO guard conductors then ..@
    XN2 YN2?"              @ enter coordinates for guard 2 @
   { "" { 1 3 } }
   INPUT OBJ\->
   'YN2'  STO 'XN2' STO
END

IF 'W==2' THEN
  XN2 Xa YN2 Ya DIST             @subroutine DISTance@
  'Dan2' STO             @distance between Guard_2 & phase A@
  XN2 Xb YN2 Yb DIST
 'Dbn2' STO              @distance between Guard_2 & phase B@
  XN2 Xc YN2 Yc DIST
 'Dcn2' STO              @distance between Guard_2 & phase C@

  Dan2 'Dij' STO zij \->NUM 'zan2' STO
  Dbn2 'Dij' STO zij \->NUM 'zbn2' STO
  Dcn2 'Dij' STO zij \->NUM 'zcn2' STO

  zan1 zan2 zag
  zbn1 zbn2 zbg
  zcn1 zcn2 zcg
  {3 3 } \->ARRY 'ZFN' STO    @ZFN matrix for 2 guard conductor@

  zan1 zbn1 zcn1
  zan2 zbn2 zcn2
  1 1 1 { 3 3 }
  \->ARRY 'ZNF' STO       @ZNF matrix for 2 guard conductor@

  XN1 XN2 YN1 YN2 DIST         @DISTance between guards@
  'DN1N2' STO
  DN1N2 'Dij' STO zij \->NUM
  DUP 'zn1n2' STO 'zn2n1' STO

  ZNN zn1n2 zng
  zn2n1 ZNN zng
  1 1 1 { 3 3 } \->ARRY
  'Znn' STO               @Znn matrix for 2 guard conductor@
END UPDIR
    \>>                            @ end subroutine SUB2@

  DATA
    DIR
      U 0
      S 1
      L 1
      K 1
      z012
[[ (.334618196749,1.38405818084) (1.31170778745E-2,-1.93989309234E-2)
(9.89985624524E-4,-1.40487181776E-2) ]
 [ (.000989985624,-1.40487181782E-2) (5.95370079595E-2,.331722910431) (-
1.41287308051E-2,2.72699637764E-2) ]
 [ (1.31170778745E-2,-1.93989309236E-2) (.014782933265,2.66903751573E-2)
(5.95370079595E-2,.33172291043) ]]
      zabc
```

```
[[ (.160853514042,.678189680813) (9.56476593098E-2,.348057293971)
(9.22240868964E-2,.324363556962) ]
 [ (9.56476593096E-2,.34805729397) (.149340983259,.683312661905)
(8.72094425835E-2,.379914419483) ]
 [ (9.22240868963E-2,.324363556961) (8.72094425835E-2,.379914419483)
(.143497715367,.686001658988) ]]
      Znn
[[ (2.61,.582590155749) (-4.93474207582E-2,-.214872253261) ]
 [ (1,0) (1,0) ]]
      zng
(-4.93474207582E-2,-.214872253261)
      zn2n1
(0,.133350048783)
      zn1n2
(0,.133350048783)
      DN1N2 3.66
      ZNF
[[ (0,.19305709596) (0,.148039788519) (0,.122798439249) ]
 [ (1,0) (1,0) (1,0) ]]
      ZFN
[[ (0,.19305709596) (-4.93474207582E-2,-.214872253261) ]
 [ (0,.148039788519) (-4.93474207582E-2,-.214872253261) ]
 [ (0,.122798439249) (-4.93474207582E-2,-.214872253261) ]]
      zcn2
(0,.15392454046)
      zbn2
(0,.15392454046)
      zan2
(0,.104319871386)
      Dcn2
2.63797270645
      Dbn2
2.63797270645
      Dan2
5.80948362593
      XN2 99
      YN2 99
      zcn1
(0,.122798439249)
      zbn1
(0,.148039788519)
      zan1
(0,.19305709596)
      Dcn1
4.3292609069
      Dbn1
2.89698118737
      Dan1
1.41509716981
      XN1 .15
      YN1 15.8
      zcg
(-4.93474207582E-2,-.214872253261)
      zbg
(-4.93474207582E-2,-.214872253261)
      zag
(-4.93474207582E-2,-.214872253261)
      ZFF
[[ (.0591,.494171768443) (0,.161373059186) (0,.136184306612) ]
 [ (0,.161373059186) (.0591,.494171768443) (0,.189396069588) ]
 [ (0,.136184306612) (0,.189396069588) (.0591,.494171768443) ]]
      zcb
(0,.189396069588)
      zbc
(0,.189396069588)
      zca
(0,.136184306612)
      zac
(0,.136184306612)
      zba
(0,.161373059186)
      zab
(0,.161373059186)
      Dij
4.3292609069
      Dbc 1.5
      Dac
3.49857113691
      Dab
2.34307490277
      Xa .9
      Ya 14.6
      Xb -.9
      Yb 13.1
      Xc -.9
      Yc 11.6
      zcc
(.0591,.494171768443)
      zbb
(.0591,.494171768443)
      zaa
(.0591,.494171768443)
      ZNN
(2.61,.582590155749)
      RGRD 2.61
      GMRG .002873
      W 1
      Ri .0591
      GMRF .011735
      TINV
```

```
[[ (.333333333333,0) (.333333333333,0) (.333333333333,0) ]
 [ (.333333333333,0) (-.166666666667,.288675134595) (-.166666666667,-
.288675134595) ]
 [ (.333333333333,0) (-.166666666667,-.288675134595) (-
.166666666667,.288675134595) ]]
      T
[[ (1,0) (1,0) (1,0) ]
 [ (1,0) (-.5,-.866025403784) (-.5,.866025403784) ]
 [ (1,0) (-.5,.866025403784) (-.5,-.866025403784) ]]

DIST   @ subroutine to calculate the distance between two points @
        \<< - SQ 3 ROLLD - SQ + \v/ \>>

Rg 4.93474207582E-2 @ Rg = 986.95E-6*freq;  freq=50 @

Dig 30.563     @ Dig=25.7*[(rho/freq)]^0.25;  rho=100 ohm*m @

zii 'Ri+i*.062832*LN(Dig/GMRF)'
@ self impedance of phase conductors@

znn 'RGRD+i*.062832*LN(Dig/GMRG)'
@ self impedance of guard conductors@

zij 'i*.062832*LN(Dig/Dij)'
@ zij = j*mu*freq*ln[Dig/r'] @

zig '-Rg+i*.062832*LN(1/Dig)'
@ zig = -Rg +j*mu*freq*ln[1/Dig] @
  END   @ DATA directory @
END  @MAIN directory@
```