CIVIL ENGINEERS
POCKET COMPUTER MONTHLY

SURVEYING - PART II

This issue contains Part Two of the three-part series on the subject of SURVEYING. It provides two programs for solving office problems in the area of GEOMETRICS. These programs are:

Open Traverse
Triangles

OPEN TRAVERSE

The Open Traverse program deals with the "latitudes and departures" type of problem. It is used to close a traverse of three or more sides. The program should be useful in solving geometric problems in property surveys, control traverses, highway interchange geometrics, and subdivision of property.

One of the features of this program which could be confusing to the user pertains to the azimuth concept. Each known leg is defined by, 1) an angle, and 2) a length. The first leg is oriented to north in the two-dimensional X-Y coordinate system by means of an angle which is simply the angle measured clockwise between north and the orientation of the first leg. The accuracy of this first angle therefore does not affect the accuracy of the traverse solution itself; it only affects the correctness of each azimuth relative to north. Indeed, if this angle is entered as zero, the first leg will be assumed to be oriented toward north, and all other azimuths calculated accordingly. Therefore, if the orientation of your traverse relative to north is not of importance, you may enter a zero for the first angle (ØØØØØ).

The program assumes that deflection angles are known at all points of intersection except those at each end of the unknown leg. The output defines the X and Y components as well as the length of the unknown leg, its azimuth, and the two unknown deflection angles. All interior angles are provided, as well.

The input parameter values are entered via the data statements, and are verified in the printout.

Entry of angles is as follows: 7Ø° is entered as 7ØØØØØ; 15° 7' 12" is entered as 15Ø712.

PROGRAM LISTING - OPEN TRAVERSE/PC 2

```
 10:"A"LPRINT "**O
    PEN TRAVERSE**
      ":LPRINT :
    CLEAR
 20:LLIST 451,
 30:"B"CLEAR :DIM
    X(100),Y(100)
 40:"J"N=N+1
 50:READ TH,L:IF T
    H=999LET PH=TK
    -TI+180:T2=PH:
    GOSUB "CHECK":
    TH=T2:PH=180-T
    H:GOTO "D"
 60:Q2=TH:GOSUB "A
    E":TH=Q2
 70:IF N=1LET T2=T
    H:GOSUB "CHECK
    ":TG=T2:TH=TG
 80:TI=TI+TH
 90:T2=TI:GOSUB "C
    HECK":TI=T2
100:X(N)=L*SIN TI
110:Y(N)=L*COS TI
120:USING
130:LPRINT :LPRINT
    "LEG# ";N:
    LPRINT "LEG LE
    NGTH=";L:
    LPRINT "LEG AZ
    IMUTH=":PH=TI:
    GOSUB "DEGMNSC
    "
140:X1=X1+X(N):SH=
    TI
150:Y1=Y1+Y(N)
160:Z=SQR(X1^2+Y1^2)
170:IF N=1THEN "J"
180:GOSUB "AZIMUTH
    "
```

PROGRAM LISTING - OPEN TRAVERSE/PC-2
(continued)

```
190:LF 1:LPRINT "A
    T";N-1;"-";N:
    LPRINT "INTERI
    OR ANGLE=":PH=
    180-TH
200:GOSUB "DEGMNSC
    "
210:LF 1:GOTO "J"
220:"D"LF 1:LPRINT
    "AT";N-1;"-";N
    :LPRINT "INTER
    IOR ANGLE=":
    GOSUB "DEGMNSC
    "
230:PH=TH:LPRINT "
    DEFL.ANGLE=":
    GOSUB "DEGMNSC
    "
240:Z=INT (Z*1000+
    .5)/1000
250:LPRINT :LPRINT
    "LEG# ";N:
    LPRINT "LEG LE
    NGTH=";Z:
    LPRINT "LEG AZ
    IMUTH=":PH=TH+
    SH:GOSUB "DEGM
    NSC"
260:T2=TG-TK+180:
    GOSUB "CHECK":
    PH=T2:TH=PH:PH
    =180-PH
270:LF 1:LPRINT "A
    T";N;"- 1":
    LPRINT "INTERI
    OR ANGLE=":
    GOSUB "DEGMNSC
    "
280:PH=TH:LPRINT "
    DEFL.ANGLE=":
    GOSUB "DEGMNSC
    "
290:LF 1:LPRINT "C
    LOSURE,X COMP.
    =":LPRINT INT
    (X1*1000+.5)/1
    000
300:LPRINT "CLOSUR
    E,Y COMP.=":
    LPRINT INT (Y1
    *1000+.5)/1000
310:LF 1:LPRINT "E
    ND":LF 3:END
320:"AZIMUTH"IF Y1
    =0LET Y1=1E-10
330:IF X1<0AND Y1<
    0LET X3=ABS X1
    :Y3=ABS Y1:TJ=
    ATN (X3/Y3):TK
    =180+TJ:RETURN
340:IF X1<0LET X3=
    ABS X1:TJ=ATN
    (X3/Y1):TK=360
    -TJ:RETURN
350:IF Y1<0LET Y3=
    ABS Y1:TJ=ATN
    (X1/Y3):TK=180
    -TJ:RETURN
360:TJ=ATN (X1/Y1)
    :TK=TJ:RETURN
370:"DEGMNSC"PL=PH
    :PJ=(PH-INT PH
    )*60:PK=INT ((
    PJ-INT PJ)*60+
    .5):PH=INT PH:
    PJ=INT PJ
380:IF PJ=59AND PK
    =60LET PH=PH+1
    :PJ=0:PK=0
390:IF PK=60LET PJ
    =PJ+1:PK=0
400:LPRINT PH;"DEG
    ";PJ;"M";PK;"S
    ";PH=PL:RETURN
410:"CHECK"IF T2<0
    LET T2=T2+360
420:IF T2>360LET T
    2=T2-360
430:RETURN
440:"AE"PH=INT (Q2
    /10000):NJ=Q2-
    10000*PH:PJ=
    INT (NJ/100):P
    K=INT (NJ-PJ*1
    00)
450:Q2=PH+(PJ+PK/6
    0)/60:RETURN
460:DATA 300000,10
470:DATA 600000,10
480:DATA 600000,10
490:DATA 600000,10
500:DATA 600000,10
510:DATA 999,999
```

TRIANGLES

This program solves triangles for the unknown sides and angles.

The sides are labelled A, B, and C; the angles are labelled ALPHA, BETA, and GAMMA. Side A is opposite angle ALPHA, Side B opposite angle BETA, and Side C opposite angle GAMMA.

The area of the triangle is provided. Four cases are solved:

A) Three sides;
B) Two sides and the included angle;
C) Two angles and the included side;
D) Two sides and the non-included angle.

The latter case presents the most complexity: The triangle might not exist, or the data might define one triangle, or the data might define two triangles.

Choose the desired case, and enter the information as it is called for. See the worked-out examples for further explanation. Press DEF A for Case A, DEF B for Case B, etc.

PROGRAM LISTING - TRIANGLES/PC-2

```
10:"="CLEAR :WAIT
   150:LPRINT "
     TRIANGLES":
   LPRINT
20:BEEP 1,200,600
   :PRINT "CASE A
   -PRESS DEF A"
30:BEEP 1,200,600
   :PRINT "CASE B
   -PRESS DEF B"
40:BEEP 1,200,600
   :PRINT "CASE C
   -PRESS DEF C"
50:BEEP 1,200,600
   :PRINT "CASE D
   -PRESS DEF D"
60:END
70:"A"CLEAR :
   LPRINT "   THR
   EE SIDES":LF 1
80:INPUT "SIDES A
   ,B,C=";A,B,C:
   GOSUB "S1":
   GOSUB "S2":
   GOSUB "S3"
90:AL=ACS ((B^2+C
   ^2-A^2)/2/B/C)
   :GOSUB "A1"
100:BE=ACS ((A^2+C
   ^2-B^2)/2/A/C)
   :GOSUB "A2"
110:GA=180-(AL+BE)
   :GOSUB "A3"
120:LF 1:GOSUB "AR
   ":LF 3:END
130:"B"CLEAR :
   LPRINT "2 SIDE
   S & INCLUDED
      ANGLE":LF
   1
140:INPUT "SIDES B
   ,C=";B,C
150:INPUT "ALPHA="
   ;AL:LPRINT "AN
   GLE ALPHA=":Q1
   =AL:GOSUB "AE"
   :AL=PH:GOSUB "
   PO"
160:A=√(B^2+C^2-2*
   B*C*COS AL):BE
   =ACS ((A^2+C^2
   -B^2)/2/A/C):
   GOSUB "A2"
170:GA=180-AL-BE:
   GOSUB "A3"
180:GOSUB "S1":
   GOSUB "S2":
   GOSUB "S3"
190:LF 1:GOSUB "AR
   ":LF 3:END
200:"C"CLEAR :
   LPRINT "  TWO
   ANGLES AND
   INCLUDED SIDE"
   :LF 1
210:INPUT "SIDE A=
   ";A
220:INPUT "BETA=";
   BE:LPRINT "ANG
   LE BETA=":Q1=B
   E:GOSUB "AE":B
   E=PH:GOSUB "PO
   "
230:INPUT "GAMMA="
   ;GA:LPRINT "AN
   GLE GAMMA=":Q1
   =GA:GOSUB "AE"
   :GA=PH:GOSUB "
   PO"
```

PROGRAM LISTING - TRIANGLES/PC-2
(continued)

```
240:AL=180-GA-BE:
    GOSUB "A1";
    GOSUB "S1"
250:B=SIN BE/SIN A
    L*A:GOSUB "S2"
260:C=SQR(B^2+A^2-2*
    A*B*COS GA):
    GOSUB "S3"
270:LF 1:GOSUB "AR
    ":LF 3:END
280:"D"CLEAR :
    LPRINT " TWO S
    IDES & NON-"
290:LPRINT "  INCL
    UDED ANGLE  ":
    LF 1
300:INPUT "SIDES A
    ,B=";A,B
310:INPUT "ALPHA="
    ;AL:LPRINT "AN
    GLE ALPHA=":Q1
    =AL:GOSUB "AE"
    :GOSUB "PO":AL
    =PH
315:BE=SIN AL*B/A:
    BE=INT (BE*100
    000+.5)/100000
320:IF BE>1LET F9=
    1:GOTO "NG"
330:IF AL>90LET QQ
    =1:GOTO "W"
340:IF INT (100*B*
    SIN AL)/100>A
    LET F9=1:GOTO
    "NG"
350:"W"BE=ASN BE:G
    A=180-AL-BE
360:C=SQR(B^2+A^2-2*
    A*B*COS GA)
370:IF QQ=0LF 1:
    LPRINT "SOLUTI
    ON#1:":LF 1
380:"Q"GOSUB "A2":
    GOSUB "A3"
390:GOSUB "S1":
    GOSUB "S2":
    GOSUB "S3"
400:GOSUB "AR":LF
    3:IF QQ=1END
410:IF INT (1000*B
    E)/1000=90OR
    INT (1000*GA)/
    1000=90LF 3:
    END
420:LF -2:LPRINT "
    SOLUTION#2:":
    LF 1
430:BE=180-BE:GA=1
    80-AL-BE:IF GA
    <=0THEN "NG"
440:C=SQR(B^2+A^2-2*
    A*B*COS GA)
450:QQ=1:GOTO "Q"
460:END
470:"AG"PJ=(PH-INT
    (PH))*60:PK=(P
    J-INT (PJ))*60
480:PJ=INT (PJ)
490:PK=INT (PK+.5)
500:RETURN
510:"PO"LPRINT INT
    PH;"DEG";PJ;"M
    ";PK;"S":
    RETURN
520:"S1"LF 1:A=INT
    (100*A+.5)/100
    :LPRINT "SIDE
    A=";A:RETURN
530:"S2"B=INT (100
    *B+.5)/100:
    LPRINT "SIDE B
    =";B:RETURN
540:"S3"C=INT (100
    *C+.5)/100:
    LPRINT "SIDE C
    =";C:LF 1:
    RETURN
550:"A1"PH=AL:
    GOSUB "AG":
    LPRINT "ANGLE
    ALPHA=":GOSUB
    "PO"
560:RETURN
570:"A2"PH=BE:
    GOSUB "AG":
    LPRINT "ANGLE
    BETA =":GOSUB
    "PO"
580:RETURN
590:"A3"PH=GA:
    GOSUB "AG":
    LPRINT "ANGLE
    GAMMA=":GOSUB
    "PO"
600:RETURN
610:"AE"PH=INT (Q1
    /10000):NJ=Q1-
    10000*PH:PJ=
    INT (NJ/100):P
    K=INT (NJ-PJ*1
    00)
620:PH=PH+(PJ+PK/6
    0)/60:RETURN
630:"NG"IF F9=1
    GOSUB "S1":
    GOSUB "S2"
640:LPRINT "SOLUTI
    ON TO THIS  TR
    IANGLE DOES NO
    T EXIST":LF 3:
    END
650:"AR"AR=C/2*B*
    SIN AL:LPRINT
    "AREA=";INT (A
    R*100+.5)/100:
    RETURN
660:END
```

WORKED-OUT EXAMPLES

All traverses are analyzed in the clockwise direction.

In this first example, a perfect hexagon is used to illustrate the program. The first leg is defined by its azimuth from north, 30° (entered in DEGMNSC as 300000) and its length, 10. (Note that all angles are measured "clockwise positive".) These are shown in DATA statement 460 as follows:

```
460 DATA 300000, 10
```

Legs 2-5 are defined in DATA statements 470-500 as DATA 600000, 10. It is obvious that the closure, Leg #6, must have an azimuth of 330°, and a length of 10.

Example #1

```
**OPEN TRAVERSE**


460:DATA 300000,10
470:DATA 600000,10
480:DATA 600000,10
490:DATA 600000,10
500:DATA 600000,10
510:DATA 999,999



LEG#  1
LEG LENGTH= 10
LEG AZIMUTH=
 30DEG 0M 0S

LEG#  2
LEG LENGTH= 10
LEG AZIMUTH=
 90DEG 0M 0S

AT 1- 2
INTERIOR ANGLE=
 120DEG 0M 0S


LEG#  3
LEG LENGTH= 10
LEG AZIMUTH=
 150DEG 0M 0S

AT 2- 3
INTERIOR ANGLE=
 120DEG 0M 0S


LEG#  4
LEG LENGTH= 10
LEG AZIMUTH=
 210DEG 0M 0S

AT 3- 4
INTERIOR ANGLE=
 120DEG 0M 0S


LEG#  5
LEG LENGTH= 10
LEG AZIMUTH=
 270DEG 0M 0S

AT 4- 5
INTERIOR ANGLE=
 120DEG 0M 0S


AT 5- 6
INTERIOR ANGLE=
 120DEG 0M 0S
DEFL.ANGLE=
 60DEG 0M 0S

LEG#  6
LEG LENGTH= 10
LEG AZIMUTH=
 330DEG 0M 0S

AT 6- 1
INTERIOR ANGLE=
 120DEG 0M 0S
DEFL.ANGLE=
 60DEG 0M 0S

CLOSURE, X COMP.=
                  5
CLOSURE, Y COMP.=
              -8.66

END
```

See Sketch, Page 8.

(Note that the last DATA statement contains 999, 999, which merely signals the computer that all data has been entered.)

Example #2

In the second example, a traverse is solved by the Open Traverse Program, and the Triangles Program.

```
  2 SIDES & INCLUDED
        ANGLE

  ANGLE ALPHA=
   90DEG 0M 0S
  ANGLE BETA =
   11DEG 18M 36S
  ANGLE GAMMA=
   78DEG 41M 24S

  SIDE A= 1529.71
  SIDE B= 300
  SIDE C= 1500


  AREA= 225000



  2 SIDES & INCLUDED
        ANGLE

  ANGLE ALPHA=
   41DEG 18M 36S
  ANGLE BETA =
   126DEG 33M 60S
  ANGLE GAMMA=
   12DEG 7M 24S

  SIDE A= 1257.29
  SIDE B= 1529.71
  SIDE C= 400


  AREA= 201962.34



  2 SIDES & INCLUDED
        ANGLE

  ANGLE ALPHA=
   23DEG 26M 0S
  ANGLE BETA =
   107DEG 4M 34S
  ANGLE GAMMA=
   49DEG 29M 26S

  SIDE A= 523.06
  SIDE B= 1257.29
  SIDE C= 1000


  AREA= 250000.64
```

```
**OPEN TRAVERSE**


460: DATA 00000, 150
     0
470: DATA 900000, 30
     0
480: DATA 600000, 40
     0
490: DATA 300000, 10
     00
500: DATA 999, 999


LEG#  1
LEG LENGTH= 1500
LEG AZIMUTH=
 0DEG 0M 0S

LEG#  2
LEG LENGTH= 300
LEG AZIMUTH=
 90DEG 0M 0S

AT 1- 2
INTERIOR ANGLE=
 90DEG 0M 0S


LEG#  3
LEG LENGTH= 400
LEG AZIMUTH=
 150DEG 0M 0S

AT 2- 3
INTERIOR ANGLE=
 120DEG 0M 0S


LEG#  4
LEG LENGTH= 1000
LEG AZIMUTH=
 180DEG 0M 0S

AT 3- 4
INTERIOR ANGLE=
 150DEG 0M 0S


AT 4- 5
INTERIOR ANGLE=
 107DEG 4M 33S
DEFL.ANGLE=
 72DEG 55M 27S
```

```
LEG#  5
LEG LENGTH= 523.05
8
LEG AZIMUTH=
 252DEG 55M 27S

AT 5- 1
INTERIOR ANGLE=
 72DEG 55M 27S
DEFL.ANGLE=
 107DEG 4M 33S

CLOSURE, X COMP.=
                500
CLOSURE, Y COMP.=
              153.59
```

See sketch, page 8.

Example #3

The solution of Cases A, B and C are obvious.

However, Case D deserves some comment. Angle Alpha and Sides A and B are entered by the user. Angle Alpha, of course, is opposite Side A. Several solutions are possible:

1) Side A is greater than Side B; only one triangle is defined.

2) Side A is less than Side B. The case is ambiguous; the program solves both triangles. However if Side A is too short, the triangle does not exist. Also, if Angle Beta is 90°, only one triangle is defined.

3) Angle Alpha is equal to or greater than 90°; Side A is equal to or less than Side B. This triangle does not exist. If A is greater than B, one solution exists.

The following Worked Out Examples illustrate the above discussion.

The following illustrates Case #2

```
 TWO SIDES & NON-
  INCLUDED ANGLE

ANGLE ALPHA=
 36DEG 52M 12S

SOLUTION#1:

ANGLE BETA =
 90DEG 0M 0S
ANGLE GAMMA=
 53DEG 7M 48S

SIDE A= 3
SIDE B= 5
SIDE C= 4

AREA= 6
```

The following illustrates Case #1

```
 TWO SIDES & NON-
  INCLUDED ANGLE

ANGLE ALPHA=
 20DEG 0M 0S

SOLUTION#1:

ANGLE BETA =
 17DEG 55M 40S
ANGLE GAMMA=
 142DEG 4M 20S

SIDE A= 5
SIDE B= 4.5
SIDE C= 8.99

AREA= 6.92

SOLUTION#2:

SOLUTION TO THIS
TRIANGLE DOES NOT
EXIST
```

The following illustrates Case #2

```
 TWO SIDES & NON-
  INCLUDED ANGLE

ANGLE ALPHA=
 37DEG 0M 0S

SOLUTION#1:

ANGLE BETA =
 59DEG 17M 15S
ANGLE GAMMA=
 83DEG 42M 45S

SIDE A= 3.5
SIDE B= 5
SIDE C= 5.78

AREA= 8.7

SOLUTION#2:

ANGLE BETA =
 120DEG 42M 45S
ANGLE GAMMA=
 22DEG 17M 15S

SIDE A= 3.5
SIDE B= 5
SIDE C= 2.21

AREA= 3.33
```

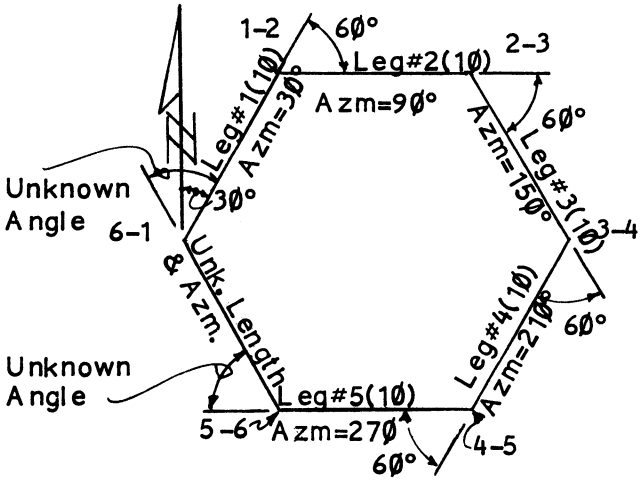
The following illustrates Case #3:

TWO SIDES & NON-
INCLUDED ANGLE

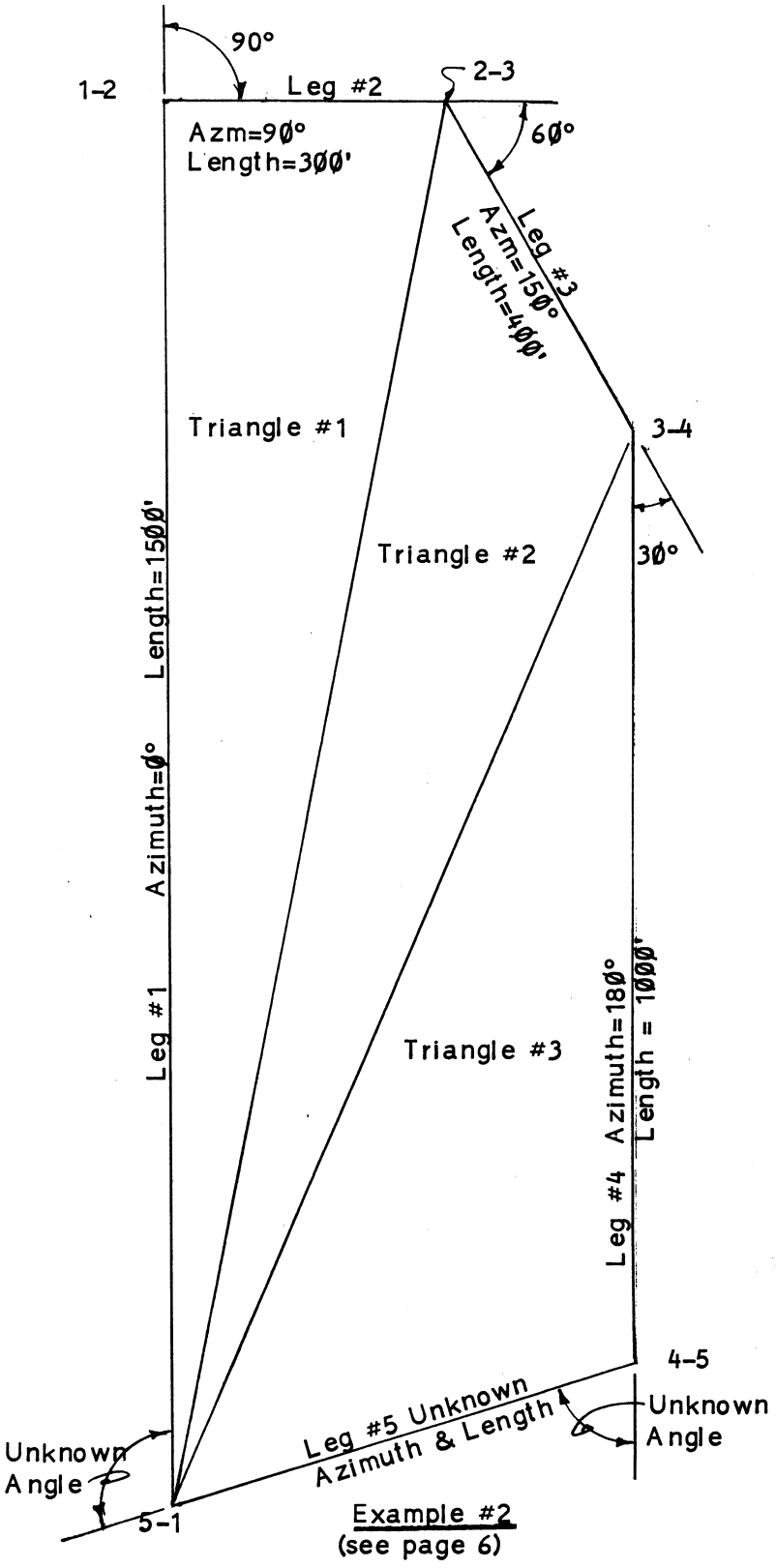
ANGLE ALPHA=
110DEG 0M 0S
ANGLE BETA =
34DEG 19M 13S
ANGLE GAMMA=
35DEG 40M 47S

SIDE A= 5
SIDE B= 3
SIDE C= 3.1

AREA= 4.37



Example #1
(see page 5)



Example #2
(see page 6)

MACHINE LANGUAGE TUTORIAL

Nat Wadsworth, a professional programmer who publishes POCKET COMPUTER NEWSLETTER (P.O. Box 232, Seymour, CT 06483) has prepared a Machine Language Tutorial described as a factual detailed reference to the machine language instruction set used in PC-2/PC-15ØØ, as well as a compendium of commonly used routines, subroutines, procedures, programming tips, and programmers aids. It is a step-by-step tutorial in four part serial form, highly illustrated for easy learning. The price is $19.95 in the US, $23.95 (US) in Canada and Mexico, and $29.95 (US) elsewhere.

TAPE RECORDERS

We have had some frustrating experiences in trying to load programs from cassette tapes. The PC-1 loads from tape "when it feels like it." We have no explanation for its erratic behavior. If the program will not load, try again at a different volume level.

Our experience with PC-2 has been much better, but not perfect either.

We use the Radio Shack Minisette-9 tape recorder. We have also used a portable Craig (Model 2625). Our advice is to use good cassettes —not the cheapest—not necessarily the most expensive either. Find a brand which works and stick with it. We have a reasonably good experience with TDK cassettes. We always SAVE at least twice. Important programs should be double-SAVEd on two tapes; this is particularly applicable to PC-1. Of course, we always make a listing so that, if all else fails, we can reload manually.

PROGRAMMERS TIP

As you know, the PC-2 can GOTO or GOSUB to a line defined by a line number. However, the PC-2 has an additional powerful programming feature which allows the program to start or to transfer to a labelled line: A, S, D, F, G, H, J, K, L, Z, X, C, V, B, N, M, as well as =, and the space bar. The label is placed in quotes after the line number. The user may start the computer program running by sending the pointer to a line designated by one of those letters/symbols; this is done by pressing, for example DEF A, which will send the pointer to the line beginning with "A".

For GOTO, GOSUB, IF THEN, and ON GOTO the label can be a word (i.e., a string of letters, numerals, symbols) up to 73 characters (the quotation mark is not permitted for obvious reasons.) You may wish to use a label describing the calculation. For example, if your subroutine calculates flexural stress, you could use FLEX, as in GOSUB "FLEX".

The following are illustrations of the use of labels:

```
1Ø     "A" INPUT I, J, K, X
2ØØ    IF X=Ø THEN "A"
3Ø     GOTO "=" (sends the pointer
       to line beginning with "=")
61Ø    ON X GOTO "A", "B", "C"
1Ø1Ø   GOSUB "SHEAR"
```

If labels are used in GOTO and GOSUB references, instead of line numbers, the program lines can be readily renumbered. The use of labels is especially important when programs are MERGEd.

This labelling feature adds power to the PC-2/PC-15ØØ version of BASIC, compared with other versions. It tends to overcome one of the fundamental faults in the BASIC language.
