## Contents

<table>
<thead>
<tr>
<th>No.s</th>
<th>Items</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Radial Bearing &amp; H-Distance by Coordinates</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Radial Coordinates by Bearing Distance</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>Point to Point XYZ</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>Reference Line</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Chainage Offset</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>Intersections by Coordinates</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td>Intersections by Bearings</td>
<td>8-9</td>
</tr>
<tr>
<td>8</td>
<td>Intersections by Angles</td>
<td>10</td>
</tr>
<tr>
<td>9</td>
<td>Cross Point</td>
<td>11</td>
</tr>
<tr>
<td>10</td>
<td>Angles Clock wise Anti-Clock wise</td>
<td>12</td>
</tr>
<tr>
<td>11</td>
<td>Point to Point Easting Northing</td>
<td>13</td>
</tr>
<tr>
<td>12</td>
<td>Point to Point Bearing&amp;Distance</td>
<td>14</td>
</tr>
<tr>
<td>13</td>
<td>2 Point Resection</td>
<td>15</td>
</tr>
<tr>
<td>14</td>
<td>3 Point Resection</td>
<td>16-17</td>
</tr>
<tr>
<td>15</td>
<td>3 Point Centre of Circle Coordinates</td>
<td>18</td>
</tr>
<tr>
<td>16</td>
<td>Simple Horizontal Curve Left &amp; Right</td>
<td>19-20</td>
</tr>
<tr>
<td>17</td>
<td>Level</td>
<td>21</td>
</tr>
<tr>
<td>18</td>
<td>Interpolation</td>
<td>22</td>
</tr>
<tr>
<td>19</td>
<td>Vertical Curve</td>
<td>23</td>
</tr>
<tr>
<td>20</td>
<td>Triangles</td>
<td>24-25</td>
</tr>
<tr>
<td>21</td>
<td>Calculator Information</td>
<td>26</td>
</tr>
</tbody>
</table>

**Instruction**

Before using the programs make some checks and make sure its working properly.
Radial Bearing & H-Distance by Coordinates

Introduction
Compute the "Bearing&Distance" from any point to other points. This program removes the need to keep re-entering the first set of co-ordinates.

Program
Deg:"STN E"?A:"STN N"?B:"SCALE FACTOR"?Z
Lbl 0:"POINT E"?C:"POINT N"?D
"BEARING=":Arg((B-D)+(A-C)i)+180►DMS
"H-DISTANCE=":Abs ((B-D)+(A-C)i)÷Z
Goto 0

Example with Diagram

Input
"From "co-ordinates: (STN E , STN N)
"To "co-ordinates: (POINT E , POINT N),?
"Depend on area" IF NO SCALE FACTOR THEN =1.00000

Output
Bearing & Distance (STN E , STN N) to (POINT E , POINT N)
(STN E , STN N) to ?
Radial Coordinates by Bearing Distance

Introduction
Compute the "Coordinates" from any point to other points. This program removes the need to keep re-entering the first set of co-ordinates.

Program
Deg:"STN E"?A:"STN N"?B:"SCALE FACTOR"?Z Lbl 0:"BEARING"?C:"H-DISTANCE"?D "EASTING=":(sin(C)×(D×Z))+A Goto 0 "NORTHING=":(cos(C)×(D×Z))+B

Example with Diagram

Input
"From "co-ordinates: (STN E , STN N) 
"To "bearing distance: (BEARING , H-DISTANCE),? 
"Curvature correction" IF NO SCALE FACTOR THEN =1.00000

Output
Easting & Northing (STN E , STN N) to (POINT E , POINT N) 
(STN E , STN N) to ?
Introduction

Compute the "Easting Northing Elevation" a point to next point

Program

Deg: Lbl 0

"STN E"?A:"STN N"?B:"STN ELEV"?C:"AZIMUTH"?D:"ZENITH ANGLE"?E:"H-DISTANCE"?F:
"H.I"?G:"T.H"?H:"SCALE FACTOR"?Z

"OBS EAST":=A+(sin(D)×(F×Z))
"OBS NORTH":=B+(cos(D)×(F×Z))
"OBS ELEV":=C+G+(sin(90-E)×F)-H

Goto 0

Example with Diagram

Input

"At Station" (STN E, STN N, STN ELEV, I.H)
"To Station" (AZIMUTH, ZENITH ANGLE, H-DISTANCE, T.H)
"Curvature correction" IF NO SCALE FACTOR THEN = 1.00000

Output

"To Station"

OBS EAST
OBS NORTH
OBS ELEV
Reference Line

Introduction
Compute the "Easting&Northing" from base line to any other point with reference of distances.

Program
Deg: "POINT-1E"?A:"POINT-1N"?B:"POINT-2E"?C:"POINT-2N"?D
Lbl 0: "DISTANCE"?E:"OFFSET+/-"?F
"REQ -E": A + (sin(Arg((B-D)+(A-C)i)+180)x E) + (sin(Arg((B-D)+(A-C)i)+180+90)x F)
"REQ -N": B + (cos(Arg((B-D)+(A-C)i)+180)x E) + (cos(Arg((B-D)+(A-C)i)+180+90)x F)
Goto 0

Example with Diagram

<table>
<thead>
<tr>
<th>POINT-1E</th>
<th>POINT-2E</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.000</td>
<td>0.000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>POINT-1N</th>
<th>POINT-2N</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.000</td>
<td>100</td>
</tr>
</tbody>
</table>

OFFSET +/-
DISTANCE 10.000

Input
"From "co-ordinates: (POINT-1E, POINT-1N)
"To "co-ordinates: (POINT-2E, POINT-2N)
"From base point" DISTANCE
"From base line" OFFSET +/-

Output
Easting&Northing
REQ-E
REQ-N
Introduction
Compute the "Distance&Offset" from base line to any other point with reference of co-ordinates.

Program
Deg:"STN E"?A:"STN N"?B:"POINT E"?C:"POINT N"?D
Lbl 0:"OBS-E"?E:"OBS-N"?F
"DISTANCE":(cos(Arg((B-F)+(A-E)+180-Arg((B-D)+(A-C)+180)+Abs ((B-F)+(A-E)+180)))
"OFFSET":(sin(Arg((B-F)+(A-E)+180-Arg((B-D)+(A-C)+180)+Abs ((B-F)+(A-E)+180)))
Goto 0

Example with Diagram

Input
"From "co-ordinates: (POINT-1E, POINT-1N)
"To "co-ordinates: (POINT-2E, POINT-2N)
"At Station" OBS-E
"At Station" OBS-N

Output
From base point DISTANCE
From base line OFFSET +/-
Introduction

Compute the "Easting&Northing" of two line intersection follow diagram sequence

Program

Lbl 0
Deg:"EAST 1"?B:"NORTH 1"?A:"EAST 2"?D:"NORTH 2"?C:"EAST 3"?F:"NORTH 3"?E:"EAST 4"?H:"NORTH 4"?G→
(A-C)÷(B-D)→I:(E-G)÷(F-H)→J
"PI EAST=":(E-A-(J×F)+(I×B))÷(I-J)→K
"PI NORTH=":A-(I×(B-K))
Goto 0

Example with Diagram

Input
"Easting & Northing "

Output
Intersection Point

INT EAST
INT NORTH
Program
Lbl 0
If E=0:Then 0.0000001→E:If End
If F=0:Then 0.0000001→F:If End
If E=90:Then 90.0000001→E:If End
If F=90:Then 90.0000001→F:If End
If E=180:Then 180.0000001→E:If End
If F=180:Then 180.0000001→F:If End
If E=270:Then 270.0000001→E:If End
If E=270:Then 270.0000001→E:If End
1÷(tan(E))→P
1÷(tan(F))→Q
tan(E)→R
tan(F)→S
((C×Q)-(A×P)+B-D)÷(Q-P)→X
((D×S)-(B×R)+A-C)÷(S-R)→Y
Fix 3
"EASTING=:"X
"NORTHING=:"Y
Lbl 3
Deg:"MENU"
"[1] NEW STANS";"[2] CONTINUE"?M
If M=1:Then Goto 1:If End
If M=2:Then Goto 2:If End
Goto 3
Example with Diagram

![Diagram of Intersections by Bearings]

**Input**
"Easting & Northing "
EAST 1, NORTH 1
EAST 2, NORTH 2
Bearings
AZIMUTH 1
AZIMUTH 2

**Output**
Intersection Point
EASTING
NORTHING
Intersections by Angles

Introduction

Compute the "Easting&Northing" of two line intersection follow diagram sequence

Program

Lbl 0
Deg:"STN-1E"?A:"STN-1N"?B:"STN-2E"?C:"STN-2N"?D:"ANGLE 1"?E:"ANGLE 2"?

180-(E+F)→G:Arg((B-D)+(A-C)i)+180→H:Abs((B-D)+(A-C)i)+I:(I×sin(F))÷(sin(G))→J

"PI EAST=":A+(sin(H-E)×J)
"PI NORTH=":B+(cos(H-E)×J)
Goto 0

Example with Diagram

Input

"Easting & Northing "

Bearings

Output

Intersection Point

STN E1=17.153
STN N1=15.870

STN E2=40.205
STN N2=9.809

PI EAST =31.712
PI NNORTH=22.991

ANGLE-1 40 47'42"

ANGLE-2 28'32"

EAST 1, NORTH 1
EAST 2, NORTH 2

ANGLE 1
ANGLE 2

INT EAST
INT NORTH
Introduction
Compute the "Cross" co-ordinates or change elevation follow the sequence which is in diagram.

Program
Lbl 0
Deg: "EAST 1"?A: "NORTH 1"?B: "EAST 2"?C: "NORTH 2"?D: "EAST 3"?E: "NORTH 3"?F: "EAST 4"?X: "NORTH 4"?Y
(D-B)+(C-A)→H: (E-X)+(F-Y)→G
"CROSS POINT EAST": (F-1÷G×E-B+H×A)÷(H-1÷G)→I
"CROSS POINT NORTH": B+H×(I-A)
Goto 0

Example with Diagram

Input
"Easting & Northing / Chainage & Elevation
EAST 1, NORTH 1
EAST 2, NORTH 2
EAST 3, NORTH 3
EAST 4, NORTH 4

Output
Cross Point
CROSS POINT EAST
CROSS POINT NORTH
Introduction

Compute the "Angle" between two line from a point.

Program

Lbl 0
Deg: "BACKSIGHT"?A:"FORSIGHT"?B
If (B-A)<0:Then 360+(B-A)→C:Else (B-A)→C
"CLOCKWISE=":C►DMS
"ANTI-CLOCKWISE=":360-C►DMS
Goto 0

Example with Diagram

Input
"Observe Station" Horizontal Angle : BACKSIGHT FORSIGHT

Output
Angles
CLOCK WISE
ANTI-CLOCK WISE
Introduction
Compute the “Easting&Northing” with the use of “Azimuth & Distance” no need to re-enter the next station co-ordinates just give the next azimuth and distance it automatically calculate the next point “Easting Northing”.

Program
Deg:”STN E”?A:”STN N”?B:”SCALE FACTOR”?Z→Z
Lbl 0:”POINT AZMUT”?C:”POINT DISTANCE”?D→D
"NEXT EASTING=":(sin(C)×(D×Z))+A→A
"NEXT NORTHING=":(cos(C)×(D×Z))+B→B
Goto 0

Example with Diagram

```
STN E=100.000
STN N=100.000

Input
"From "co-ordinates: (STN E , STN N)
"To "co-ordinates: (AZIMUTH,DISTANCE)
"Curvature correction" IF NO SCALE FACTOR THEN = 1.00000

Output
Bearing & Distance NEXT-EAST
NEXT NORTH

SCALE FACTOR=0.999621
```

Program
Point to Point Easting Northing
Compute the “Easting&Northing” with the use of “Azimuth & Distance” no need to re-enter the next station co-ordinates just give the next azimuth and distance it automatically calculate the next point “Easting Northing”.

```
Deg:”STN E”?A:”STN N”?B:”SCALE FACTOR”?Z→Z
Lbl 0:”POINT AZMUT”?C:”POINT DISTANCE”?D→D
"NEXT EASTING=":(sin(C)×(D×Z))+A→A
"NEXT NORTHING=":(cos(C)×(D×Z))+B→B
Goto 0
```

Example with Diagram

```
STN E=100.000
STN N=100.000

Input
"From "co-ordinates: (STN E , STN N)
"To "co-ordinates: (AZIMUTH,DISTANCE)
"Curvature correction" IF NO SCALE FACTOR THEN = 1.00000

Output
Bearing & Distance NEXT-EAST
NEXT NORTH

SCALE FACTOR=0.999621
```
**Introduction**

Compute the "Bearing&Distance" with the use of "Easting & Northing" no need to re-enter the station co-ordinates again just give the next Easting and Northing its automatically calculate the next point "Bearing Distance".

**Program**

Deg:"STN E"?A:"STN N"?B:"SCALE FACTOR"?Z
Lbl 0:"POINT E"?C:"POINT N"?D
"BEARING=:Arg((B-D)+(A-C)i)+180►DMS
"H-DISTANCE=:Abs ((B-D)+(A-C)i)/Z
C→A:D→B ▶Goto 0

**Example with Diagram**

![Diagram of Point to Point Bearing Distance](image)

**Input**

"From "co-ordinates: (STN E , STN N)
"To "co-ordinates: (AZIMUTH, DISTANCE)
"Curvature correction" IF NO SCALE FACTOR THEN = 1.00000

**Output**

Bearing & Distance

NEXT-EAST

NEXT NORTH
Introduction

Compute the "Co-ordinate" of free Station from two point resection

Program

```
Lbl 0
Deg:"1ST POINT E"?B:"1ST POINT N"?A:"2ND POINT E"?D:"2ND POINT N"?C:
"1ST POINT DIST"?E:"2ND POINT DIST"?F
A-C→X:B-D→Y:Arg (X+Yi)+180→J:Abs (X+Yi)→I:(J+cos⁻¹((I²+E²-F²)/(2×I×E))→T:E*Z→T
"STN E=":B+sin(T)×E→N
"STN N=":A+cos(T)×E→M
"OBS ANGLE=":cos⁻¹((E²+F²-I²)/(2×E×F))→DMS
N-C→X:M-D→Y:"BRG POINT=":Arg (X+Yi)+180→DMS
"Distance=":Abs (X+Yi)
Goto 0
```

Example with Diagram

![Diagram showing 2 Point Resection](image)

Input

- "co-ordinates" 1ST POINT E, 1ST POINT N
- "Distance" 1ST POINT DIST

Output

- "Free Station co-ordinates" STN E
- STN N
Introduction

Compute the "Co-ordinate" of free Station from three point angular resection

Program

Lbl 1
Deg:"1ST POINT E"?A:"1ST POINT N"?B:"2ND POINT E"?C:"2ND POINT N"?D:"3RD POINT E"?E:"3RD POINT N"?F

Lbl 2
"AZIMUTH 1"?G:"AZIMUTH 2"?H:"AZIMUTH 3"?I

(A-C)→J:(B-D)→K:(A-E)→L:(B-F)→M
Arg(K+Ji)+180→N:Arg(M+Li)+180→O
If O-N<0:Then O-N+360→P:Else O-N→P
(C-A)→J:(D-B)→K:(C-E)→L:(D-F)→M
Arg(K+Ji)+180→N:Arg(M+Li)+180→O
If N-O<0:Then N-O+360→Q:Else N-O→Q
(E-A)→J:(F-B)→K:(E-C)→L:(F-D)→M
Arg(K+Ji)+180→N:Arg(M+Li)+180→O
If O-N<0:Then O-N+360→R:Else O-N→R
If I-H<0:Then I-H+360→S:Else I-H→S
If G-I<0:Then G-I+360→T:Else G-I→T
If H-G<0:Then H-G+360→U:Else H-G→U
1÷((1÷Tan(P))-(1÷Tan(S)))→V
1÷((1÷Tan(Q))-(1÷Tan(T)))→W
1÷((1÷Tan(R))-(1÷Tan(U)))→X
((V×A)+(W×C)+(X×E))+((V×E)+(W×D)+(X×F))→M
Fix 4
"EASTING=":M
"NORTHING=":N

Lbl 3
"MENU":[1] NEW STNS:,[2] NEW BRGS"?Z
If Z=1:Then Goto 1:If End
If Z=2:Then Goto 2:If End
Goto 3
Example with Diagram

Input
"Obs Station "co-ordinates:  
(1ST POINT E,1ST POINT N)  
(2ND POINT E,2ND POINT N)  
(3RD POINT E,3RD POINT N)

Output
Free Station Co-ordinates:  EASTING,NORTHING
Introduction
Compute the "Northing Easting" of Circle from three co-ordinates always input co-ordinates in clock wise direction.

Program
Lbl 1 →
Deg:"EAST 1"?A:"NORTH 1"?B:"EAST 2"?C:"NORTH 2"?D:"EAST 3"?E:"NORTH 3" →
(A+C)÷2→G:(B+D)÷2→H:(C+E)÷2→I:(D+F)÷2→J →
If(B-D)=0:Then D+.00001→D:If End →
If(D-F)=0:Then D+.00001→D:If End →
If(C-A)=0:Then C+.00001→C:If End →
If(C-E)=0:Then C+.00001→C:If End →
\[\tan^{-1}\left(\frac{(C-A)}{(D-B)}\right)+90\] →K →
\[\tan^{-1}\left(\frac{(C-E)}{(D-F)}\right)+90\] →L →
1÷(tan(K))→P:1÷(tan(L))→Q →
(tan(K))→R:(tan(L))→S →
((I×Q)-(G×P)+H-J)-(Q-P)→X →
((J×S)-(H×R)+G-I)-(S-R)→Y →
Fix 3 →
"C.P EAST=":X →
"C.P NORTH=":Y →
"RADIUS=":\sqrt{(A-X)^2+(B-Y)^2} →
Goto 1 →

Input
"At "co-ordinates:
1st E=23.7368 N=19.0763
2nd E=27.5617 N=18.7473
3rd E=30.0259 N=15.8035

Output
"Centre" Point Co-ordinates: C.P EAST, C.P NORTH
RADIUS：5.000
Introduction

Compute the "Northing & Easting" of Horizontal Curve at any given Chainage and Offset Program

"BC CHAINAGE"?A:"BC EASTING"?B:"BC NORTHING"?C:
"EC EASTING"?D:"EC NORTHING"?E:"RADIUS"?F
"LONG CORD= Abs((B-D)+(C-E)i)→G
  (sin^-1((G÷2)÷F)×2)→H
"DEFLECTION ANGLE=":H▷DMS
"TANGENT LENGTH=":((G÷2)÷cos(H÷2))
"CURVE LENGTH=":(((F×2)×π)÷360)×H→L
"MID ORDINATE=":((G÷2)×tan(H=4))→I
"APEX=":((G÷2)×tan(H÷2))→J
"RISE=":F-II
"[DIRECTION]=":[1]RIGHT":[2]LEFT"?Z
If Z=1:Then 1→Z:If End
If Z=2:Then -1→Z:If End
Arg((C-E)+(B-D)i)+180→J:J+(90-(H÷2))×Z→J
B+sin(J)×F→X:C+cos(J)×F→Y
Arg((Y-C)+(X-B)i)+180→K
Lbl 0
"OFFSET"?N
If N=0:Then 1→W:If End
If N=0:Then Goto 2:Else Goto 5
Lbl 5
If W=1:Then -1→W:If End
If W=2:Then 1→W:If End
Lbl 2
"[1] EAST NORTH":[2]CHAIN OFFSET"?M
If V=1:Then Goto 3:If End
If V=2:Then Goto 4:If End
Lbl 3
"REQ CHAINAGE"?A
"REQ EASTING=":X+(sin(K+(((H+L)×(M-A))×Z))×(F-(N×Z×W)))
"REQ NORTHING=":Y+(cos(K+(((H+L)×(M-A))×Z))×(F-(N×Z×W)))
Goto 3
Lbl 6
Lbl 4
"OBS EASTING"?T
"OBS NORTHING"?O
Abs((Y-O)+(X-T)i)→P:((F-(N×Z×W)))→S:Arg((Y-O)+(X-T)i)+180→Q
If Z=1:Then Q→K:If End
If Z<0:Then K→Q:If End
If R<0:Then S+360→R:If End
If R<0:Then R:If End
If R>H:Then Goto 7:If End
"CHAINAGE=":(((((F×2)×π)÷360)×R))
"OFFSET=":(((F-(N×Z×W))-P)×Z)
Goto 6
Lbl 7
"PNT NOT IN CURVE"?R
Goto 6
____ __
Simple Horizontal Curve Left & Right E,N&Chain,Off

Example with Diagram

[1] EAST NORTH
CHAINAGE
L-OFFSET
FOR LEFT CURVE OFFSET
WILL BE OPPOSIT
B.C
R-OFFSET
[1] EAST NORTH
THIS OPTION WILL CALCULATE
(EASTING&NORTHING) AT REQUIRED
CHAINAGE AND OFFSET

[2] CHAIN OFFSET
CHAINAGE
L-OFFSET (-ANS)
FOR LEFT CURVE OFFSET
WILL BE OPPOSIT
B.C
R-OFFSET (ANS)
[2] CHAIN OFFSET
THIS OPTION WILL CALCULATE
(CHAINAGE&OFFSET) WITH OBSERVED
EASTING&NORTHING
Introduction
Compute the "Level or Staff reading" for level calculation or Level marking from Bench Mark.

Program
Deg:"BM"?A:"BACK SIGHT"?B
Lbl 0:"F.S-I.S-REQ-LEVEL"?C
"LEVEL/I.S=:/(A+B)-C"
Goto 0

Example with Diagram

Input
"REFERENCE POINT: B.M"
"OBS READING: BACK SIGHT"
"REQUIRED: REQ LEVEL/I.S/F.S"

Output
CALCULATED LEVEL/READING REQ LEVEL/I.S/F.S
Interpolation

Compute the "Level in Slopes" at required chainage.

**Program**

```
Deg:"STRAT ELEV”?A:"START CH”?B:"END ELEV”?C:"END CH”?D ↓
Lbl 0: "REQ CH”?E ↓
"REQ ELEV=":((C-A)÷(D-B)×(E×B))+A ↓
Goto 0 ↓
```

**Example with Diagram**

![Diagram showing chainage and elevation values](image)

**Input**

- "Start" Chainage: START CH
- "Start" Level: START ELEV
- "End" Chainage: END CH
- "End" Level: END ELEV
- "Required" Chainage: REQ CH

**Output**

- Required Elevation: REQ ELEV
**Introduction**

Compute the "Elevation" on Vertical Curve. This program calculates the elements of curve also.

**Program**

Deg:"PVI CH"?A:"PVI ELEV"?B:"CURVE LENGTH"?C:"GRAD-1%"?D:"GRAD-2%"?E

"PVC CH"=":A-(C÷2)→F ▽
"PVC ELEV"=":B-(D÷100)×(C÷2)→G ▽
"PVT CH"=":F+C→H ▽
"PVT ELEV"=":B+((E÷100)×(C÷2))→I ▽
Lbl 0:"REQ CH"?J ▽
"REQ ELEV"=":G+((D÷100)×(J-F))+(((E-D)÷(200×C))×(J-F)^2) ▽
Goto 0

**Example with Diagram**

![Diagram of Vertical Curve]

**Input**

"At"Chainage: PVI CH
"At"Elevation: PVI ELEV
"Curvature" CURVE LENGTH
"PVC to PVI "Slope percentage: GRAD-1%
"PVI to PVT" Slope percentage: GRAD-2%
Required Chainage between Start to End REQ CH

**Output**

Curve Start Chainage PVC CH
Curve Start Elevation PVC ELEV
Curve End Chainage PVT CH
Curve End Elevation PVT ELEV
Elevation on Required Chainage REQ ELEV
Introduction

Compute the "Sides & Angles & Area" of triangle

Program

LbI 3
If Z=1:Then Goto 0:If End
If Z=2:Then Goto 1:If End
If Z=3:Then Goto 2:If End
LbI 0

Deg:"SIDE A"?A:"SIDE B"?B:"SIDE C"?C
"ANGLE A":\( \cos^{-1} \left( \frac{(B)^2+(C)^2-(A)^2}{2\times B\times C} \right) \)→F►DMS
"ANGLE B":\( \cos^{-1} \left( \frac{(A)^2+(C)^2-(B)^2}{2\times A\times C} \right) \)►DMS
"ANGLE C":\( \cos^{-1} \left( \frac{(A)^2+(B)^2-(C)^2}{2\times A\times B} \right) \)►DMS
G"AREA"=(C×B×sin(F))÷2
Goto 3

LbI 1

Deg:"ANGLE A"?A:"SIDE B"?B:"SIDE C"?C
"SIDE A"= \( \sqrt{(B^2+C^2)-(2\times B\times C\times \cos(A))} \)→D
"ANGLE B":\( \cos^{-1} \left( \frac{(D)^2+(C)^2-(B)^2}{2\times D\times C} \right) \)→E:E►DMS
"ANGLE C"=180-(A+E)►DMS
"AREA"=(B×C×sin (A))÷2
Goto 3

LbI 2

Deg:"SIDE A"?A:"ANGLE B"?B:"ANGLE C"?C
"ANGLE A"=180-(B+C)→D►DMS
"SIDE B":(A×sin(B))÷sin(D)→E:E
"SIDE C":(E×sin(C))÷sin(B)→F:F
"AREA"=(F×E×sin(D))÷2
Goto 3
Example with Diagram

Input
"Sides or Angle" Select Option 1 or 2 or 3
[1] Side A, Side B, Side C
[2] "or" Angle A, Side B, Side C

Output
"Sides or Angle" (Angle A, Angle B, Angle C), Area
"or" (Side A, Angle B, Angle C), Area
"or" (Angle A, Side B, Side C), Area

Triangles

Side b=25.000
Side a=22.000
Side c=20
Area =210.256 sqm
New Program:
Press MODE
5:PROG Press 5
1:NEW Press 1
File Name?
[ TEST ] Press 1
File Mode
1:COMP
3:Formula 2:BASE-N
Press 1
Start Prog installation