Option Explicit
Dim numelements As Integer

Public Function ToDegree(Angle As Single) As Single
    ToDegree = Angle / 3.14159 * 180#
End Function

Public Function GetAngle(dx As Single, dy As Single) As Single
    If dx >= 0# Then
        If dy > 0# Then
            GetAngle = Atn(dx / dy)
        Else
            GetAngle = 3.14159 - Atn(Abs(dx) / Abs(dy))
        End If
    Else
        If dy <= 0# Then
            GetAngle = 3.14159 + Atn(Abs(dx) / Abs(dy))
        Else
            GetAngle = 2 * 3.14159 - Atn(Abs(dx) / Abs(dy))
        End If
    End If
End Function

Public Sub DegMinSec(decang As Single, degang As Single, minang As Single, secang As Single)
    degang = Int(decang)
    decang = decang - degang
    minang = Int(decang * 60#)
    decang = decang * 60# - minang
    secang = decang * 60#
End Sub

Public Function ToRad(Angle As Single) As Single
    ToRad = Angle / 180 * 3.14159
End Function

Public Sub Traverse()
    Dim Azimuth(1 To 6) As Single, Distance(1 To 6) As Single
    Dim index As Integer
    Dim X0 As Single, Y0 As Single
    Dim x(1 To 6) As Single, y(1 To 6) As Single
    Dim deltax(1 To 6) As Single, deltay(1 To 6) As Single
    Dim Length(1 To 6), Perimeter As Single, Angle(1 To 6) As Single
    Dim AngleDeg(1 To 6) As Single
    Dim AngleMin(1 To 6) As Single
    Dim AngleSec(1 To 6) As Single
    Dim Bearing(1 To 6) As Single
    Dim NorthSouth(1 To 6) As String, EastWest(1 To 6) As String
    Dim area As Single
    X0 = 1000#
    Y0 = 1000#
    numelements = 6
    For index = 1 To numelements
        Azimuth(index) = ActiveCell.Offset(index - 1, 0).Value
        Distance(index) = ActiveCell.Offset(index - 1, 1).Value
        x(index) = Distance(index) * Sin(ToRad(Azimuth(index))) + X0
        y(index) = Distance(index) * Cos(ToRad(Azimuth(index))) + Y0
        ActiveCell.Offset(index - 1, 2).Value = x(index)
        ActiveCell.Offset(index - 1, 3).Value = y(index)
    Next index
    Perimeter = 0#
    For index = 1 To numelements - 1
        deltax(index) = x(index + 1) - x(index)
        deltay(index) = y(index + 1) - y(index)
        Length(index) = (deltax(index) ^ 2 + deltay(index) ^ 2) ^ (1 / 2)
        Perimeter = Perimeter + Length(index)
    Next index
    deltax(numelements) = x(1) - x(6)
    deltay(numelements) = y(1) - y(numelements)
    Length(numelements) = (deltax(numelements) ^ 2 + deltay(numelements) ^ 2) ^ (1 / 2)
    Perimeter = Perimeter + Length(numelements)
    Range("A1").Value = Perimeter
    For index = 1 To numelements
        ActiveCell.Offset(index - 1, 4).Value = ToRad(Azimuth(index))
    Next index
End Sub
ActiveCell.Offset(index - 1, 4).Value = Length(index)
Angle(index) = ToDegree(GetAngle(deltax(index), deltay(index)))
Call AzimuthToBearing(Angle(index), Bearing(index), NorthSouth(index), EastWest(index))
Call DegMinSec(Bearing(index), AngleDeg(index), AngleMin(index), AngleSec(index))
ActiveCell.Offset(index - 1, 6).Value = NorthSouth(index)
ActiveCell.Offset(index - 1, 7).Value = AngleDeg(index)
ActiveCell.Offset(index - 1, 8).Value = AngleMin(index)
ActiveCell.Offset(index - 1, 9).Value = AngleSec(index)
ActiveCell.Offset(index - 1, 10).Value = EastWest(index)
Next index
Call CalcArea(deltax, y, area)
Range("A2").Value = area
End Sub

Public Sub AzimuthToBearing(Az As Single, Br As Single, NS As String, EW As String)
Select Case Az
Case Is = 0#
   NS = "N"
   EW = ""
   Br = 0#
Case Is < 90#
   NS = "N"
   EW = "E"
   Br = Az
Case Is = 90#
   NS = ""
   EW = "E"
   Br = 0#
Case Is < 180#
   NS = "S"
   EW = "E"
   Br = 180# - Az
Case Is = 180#
   NS = "S"
   EW = ""
   Br = 0#
Case Is < 270#
   NS = "S"
   EW = "W"
   Br = Az - 180#
Case Is = 270#
   NS = ""
   EW = "W"
   Br = 0#
Case Is < 360#
   NS = "N"
   EW = "W"
   Br = 360# - Az
Case Else
   NS = "N"
   EW = ""
   Br = 0#
End Select
End Sub

Public Sub CalcArea(dx() As Single, y() As Single, area As Single)
Dim index As Integer, incarea As Single
area = 0#
For index = 1 To numelements - 1
   incarea = (y(index) + y(index + 1)) / 2 * dx(index)
   If dx(index) > 0# Then
      area = area + incarea
   Else
      area = area - incarea
   End If
Next index
incarea = (y(1) + y(numelements)) / 2# * dx(numelements)
If dx(numelements) > 0# Then
   area = area + incarea
Else
   area = area - incarea
Else
Module1 - 3

        carea = carea - incarea
    End If

End Sub