# Distance From Shore

**Distance = Distance from shore in nautical miles**. The CalCOFI database includes a calculated distance-from-shore value used primarily for vertical section plots published in the hydrographic data reports. CalCOFI line coastline intercepts, digitized from Google maps, are tabulated below; the latitudes and longitudes are the origins (zero starting point) for vertical section plots.

| Line | Sta (if calculated) | Latitude N | Longitude W |
|------|---------------------|------------|-------------|
| 93.3, 93 | 26.2 | 32.978982 | -117.272386 |
| 90.0 | 27.6 | 33.497430 | -117.741800 |
| 86.7, 87 | 32.1 | 33.917500 | -118.430800 |
| 83.3, 83 | 39.1 | 34.273350 | -119.308200 |
| 80.0 | 50.3 | 34.474500 | -120.476500 |
| 76.7, 77 | 47.23 | 35.148000 | -120.653000 |
| 73.3, 73 | 49.6 | 35.653333 | -121.221667 |
| 70.0 | 50.9 | 36.196667 | -121.723333 |
| 66.7, 67 | 46.9 | 36.903333 | -121.845000 |
| 63.3, 63 | 49.3 | 37.398333 | -122.425000 |
| 60.0 | 49.0 | 37.988333 | -122.815000 |

## Basic Source Code (translated from F77 Fortran)

```
Public Static Sub OLDDS(LAT1!, Lon1!, D!, Liine$)
'C
'C -- ROUTINE TO RETURN DISTANCE IN NAUTICAL MILES GIVEN TWO POSITIONS
'C    IN DECIMAL DEGREES.  USES A METHOD THAT WILL MATCH THE RESULTS
'C    PRODUCED IN THE OLD VERTICAL SECTION PROGRAM.
'C
'C -> LAT0  -- LATITUDE OF FIRST POINT. (DECIMAL DEGREES. SOUTH NEGATIVE.)
'C -> LON0  -- LONGITUDE OF FIRST POINT. (WEST NEGATIVE)
'C -> LAT1  -- LAT. OF SECOND POINT.
'C -> LON1  -- LONG. OF SECOND POINT.
'C <- D     -- DISTANCE (NAUTICAL MILES) BETWEEN POINTS.
'C
  Dim COSD#
'     Real LAT1, LON1, LAT2, LON2, D
'     Real L, DLO, COSD, PI, F, A
      PI# = 3.141592654
      F# = PI# / 180
'C
'     INTRINSIC COS,ABS,SQRT
'C
      'COSD#(A#) = Cos(A# * F#)
      If frmSelData.txtDataIn(6) = "" Then
        LineIn! = (Val(Liine$))
      Else
        LineIn! = Val(frmSelData.txtDataIn(6))
      End If
      D! = 0
      Dist2Orig$ = ""
190
      Select Case CInt(LineIn!)
```

```vb
Case 93, 94 ', 93.3
    Lat0! = 32.969    '93.3 26.2 Google projection 2010; old value 32.95298 from Ralf & Nav Program
    Lon0! = -117.2696 'Google projection 2010; old value -117.2664
Case 90, 91
    Lat0! = 33.49743   '90.0 27.6  Google checked 2010    '4969634422
    Lon0! = -117.7418      '745017037
Case 87, 86 ', 86.7
    Lat0! = 33.9175    '86.7 32.1 Google projection 2010; old .89568     '918195424
    Lon0! = -118.4308  'Google projection 2010; old .419     '431669307
Case 83, 84  ', 83.3
    Lat0! = 34.27335   '83.3 39.1  Google checked 2010  '2739356345
    Lon0! = -119.3082    '310048755
Case 80, 79
    Lat0! = 34.4745     '80.0 50.3 Google projection 2010; old .4631       '4823190269
    Lon0! = -120.4765  'Google projection 2010; old .4739    '485679527
Case 77, 76  ', 76.7
    Lat0! = 35.148     '76.7 47.23  Google checked 2010  '1446809515
    Lon0! = -120.653     '650653673
Case 73  ', 73.3
    Lat0! = 35.6533333333
    Lon0! = -121.221666667
Case 70
    Lat0! = 36.1966666667
    Lon0! = -121.723333333
Case 67  ', 66.7
    Lat0! = 36.9033333333
    Lon0! = -121.845
Case 63
    Lat0! = 37.3983333333
    Lon0! = -122.425
Case 60  ', 57.8, 57.9
    Lat0! = 37.9883333333
    Lon0! = -122.815
Case Else
 Select Case LineIn!
  Case 91.7, 91.6, 91.8, 92
    Lat0! = 33.25773      'from Ralf & Nav Program
    Lon0! = -117.4385
  Case 88.5, 88.6, 88.4, 88
    Lat0! = 33.69985      'from Ralf & Nav Program
    Lon0! = -118.0522
  Case 85.4, 85.5, 85.6
    Lat0! = 34.01588      'from Ralf & Nav Program
    Lon0! = -118.8243
  Case 82, 81.9, 82.1
    Lat0! = 34.435         'from Ralf & Nav Program
    Lon0! = -119.955
  Case 81.7, 81.8, 81
    Lat0! = 34.4174        'from Ralf & Nav Program
    Lon0! = -119.8033
  Case 78.5, 78.4, 78.6
    Lat0! = 34.77699      'from Ralf & Nav Program
    Lon0! = -120.6302
  Case Else
   If LineIn! <> Val(Liine$) And LineIn! <> Val(StndL$) Then
    LineIn! = Val(StndL$)
    GoTo 190
   Else
    Exit Sub
   End If
```

```
    End Select
  End Select
'C
    Lon0! = Lon0! * -1
    L! = Abs(LAT1! - Lat0!)
    DLO! = Abs(Lon1! - Lon0!)
    If (DLO! > 180) Then DLO! = 360 - DLO!
'C
    L! = L! * 60
    DLO! = DLO! * 60
    DLO! = DLO! * Cos(F# * ((Lat0! + LAT1!) / 2))
    D! = (Sqr(L! * L! + DLO! * DLO!)) * -1
    Dist2Orig$ = Format(D!, "#0.00")
    If D! = 0 Then Dist2Orig$ = ""


End Sub
```