HP-33S Calculator Program Closure 6

Traverse Program using Latitude and Longitude and the Gauss Mid-Latitude Formulae

Programmer: Dr. Bill Hazelton

Date: May, 2006.

Line	Instruction	Display	User Programming Instructions
T0001	LBL T		
T0002	SF 10		FLAGS SF .0
T0003	6378137		
T0004	STO A		
T0005	6.699438 E–3		
T0006	STO E		
T0007	ENTER A		EQN RCL E RCL N etc. ENTER to end
T0008	PSE		
T0009	INPUT A		
T0010	ENTER E		EQN RCL E RCL N etc. ENTER to end
T0011	PSE		
T0012	INPUT E		
T0013	1		
T0014	STO M		
T0015	FEET–METERS		EQN RCL F etc. and – (minus).
T0016	PSE		
T0017	INPUT M		
T0018	0		
T0019	STO H		
T0020	STO F		
T0021	STO L		
T0022	ENTER H		EQN RCL E RCL N etc. ENTER to end
T0023	PSE		
T0024	INPUT H		
T0025	ENTER LAT		EQN RCL E RCL N etc. ENTER to end
T0026	PSE		
T0027	INPUT F		
T0028	ENTER LONG		EQN RCL E RCL N etc. ENTER to end
T0029	PSE		
T0030	INPUT L		
T0031	RCL F		
T0032	→HR		
T0033	STO F		
T0034	STO Y		
T0035	RCL L		
T0036	→HR		
T0037	STO L		
T0038	STO X		
T0039	1		

Traverse Program Using Latitude and Longitude

T0040	RCL H		
T0041	RCL× M		
T0042	1.571 E–7		
T0043	×		
T0044	–		
T0045	STO H		
U0001	LBL U		
U0002	AZIMUTH		EQN RCL A RCL Z etc. ENTER to end
U0003	PSE		
U0004	INPUT Z		
U0005	DISTANCE		EQN RCL D RCL I etc. ENTER to end
U0006	PSE		
U0007	INPUT D		
U0008	RCL Z		
U0009	→HR		
U0010	STO Z		
U0011	RCL M		
U0012	STO× D		
U0013	RCL H		
U0014	STO× D		
U0015	RCL Z		
U0016	RCL D		
U0017	$\theta, r \to y, x$		4
U0018	RCL÷ A		
U0019	→ DEG		
U0020	STO S		
U0021	2		
U0022	÷		
U0023	STO B		
U0024	x < > y		
U0025	RCL F		
U0026	RCL+ B		
U0027	COS		
U0028	÷		
U0029	RCL÷ A		
U0030	→ DEG		
U0031	STO T		
U0032	2		
U0033	÷		
U0034	STO C		
U0035	1.004		
U0036	STO Q		
V0001	LBL V		
V0002	RCL F		
V0003	RCL+ B		
V0004	SIN		
V0005	RCL C		

V0006	TAN		
V0007	×		
V0008	RCL B		
V0009	COS		
V0010	÷		
V0011	ATAN		
V0012	STO G		
V0013	1		
V0014	RCL– E		
V0015	RCL× A		
V0016	RCL F		
V0017	RCL+ B		
V0018	SIN		
V0019	RCL× E		
V0020	1		
V0021	x < > y		
V0022	–		
V0023	1.5		
V0024	y^x		
V0025	÷		
V0026	STO R		
V0027	RCL A		
V0028	1		
V0029	RCL F		
V0030	RCL+ B		
V0031	SIN		
V0032	RCL× E		
V0033	–		
V0034	√x		
V0035	÷		
V0036	STO N		
V0037	RCL D		
V0038	RCL÷ R		
V0039	RCL Z		
V0040	RCL+ G		
V0041	COS		
V0042	×		
V0043	→ DEG		
V0044	STO S		
V0045	RCL Z		
V0046	RCL+ G		
V0047	SIN		
V0048	RCL× D		
V0049	RCL÷ N		
V0050	RCL F		
V0051	RCL+ B		
V0052	COS		

Traverse Program Using Latitude and Longitude

V0053	÷		
V0054	→ DEG		
V0055	STO T		
V0056	RCL S		
V0057	2		
V0058	÷		
V0059	STO B		
V0060	RCL T		
V0061	2		
V0062	÷		
V0063	STO C		
V0064	ISG Q		
V0065	GTO V		
V0066	RCL S		
V0067	STO+ F		
V0068	RCL T		
V0069	STO+ L		
V0070	RCL F		
V0071	→ HMS		
V0072	STO F		
V0073	LATITUDE		EQN RCL L RCL A etc. ENTER to end
V0074	PSE		
V0075	VIEW F		
V0076	RCL L		
V0077	→ HMS		
V0078	STO L		
V0079	LONGITUDE		EQN RCL L RCL O etc. ENTER to end
V0080	PSE		
V0081	VIEW L		
V0082	RCL Z		
V0083	RCL+ G		
V0084	RCL+ G		
V0085	→ HMS		
V0086	STO Q		
V0087	AZIMUTH		EQN RCL A RCL Z etc. ENTER to end
V0088	PSE		
V0089	VIEW Q		
V0090	RCL F		
V0091	→ HR		
V0092	STO F		
V0093	RCL L		
V0094	→ HR		
V0095	STO L		
V0096	NEW LINE = 1		EQN RCL N RCL E etc. ENTER to end
V0097	PSE		
V0098	STOP		
V0099	x > 0?		

V0100	GTO U		
V0101	RCL F		
V0102	RCL– Y		
V0103	→ HMS		
V0104	STO B		
V0105	RCL L		
V0106	RCL– X		
V0107	→ HMS		
V0108	STO C		
V0109	MISC LAT		EQN RCL M RCL I etc. ENTER to end
V0110	PSE		
V0111	VIEW B		
V0112	MISC LONG		EQN RCL M RCL I etc. ENTER to end
V0113	PSE		
V0114	VIEW C		
V0115	CF 10		FLAGS CF .0
V0116	STOP		

Notes

(1) Simple computation of co-ordinates around a traverse, together with a simple computations of misclosure, where the traverse point locations are latitudes and longitudes, and the lines have their starting azimuth and linear distance available.

(2) Brief prompts are provided before each requirement for data entry, as well as before results are displayed. Each prompt shows for about 1 second, and is then replaced by the value or request for input.

(3) Co-ordinates of the traverse points (latitude and longitude) are not stored, and so must be written down to record them.

(4) Angles, including latitudes, longitudes and azimuths, are entered and displayed in HP notation, i.e., DDD.MMSS. Internal storage of angles and azimuths is in decimal degrees.

(5) Latitudes are positive North of the Equator, negative South of the Equator. Longitudes are positive to the East of the Greenwich meridian, negative to the West of the Greenwich meridian. This means that in the US, latitudes are positive and longitudes are negative.

(5) The program computes the latitude and longitude of the next traverse point from the azimuth and distance of the line from the current point. This is the classical 'forward' line computation problem in geodesy. The program uses the Gauss Mid-Latitude Formulae for the calculation, iterating each line four times. The program can also be used for stand-alone Gauss Mid-Latitude computations.

(6) The forward azimuth of the line at the end of each line is displayed. This allows the user to compute the azimuth of the next line, using the back azimuth of the current line and the angle measured at the traverse point, if this is required.

(7) The Gauss Mid-Latitude Formulae take care of any spherical excess in the figure, leaving the measured angle misclosure in the resulting azimuths.

(8) The formulae are designed to work with the WGS84/NAD83/GRS80 ellipsoid. If a different ellipsoid is required, the a and e^2 values can be changed at the start of the program. If computations on a spherical figure are required, enter the required radius for a, and set e^2 to zero.

(9) Users can enter an average height for lines, to allow lines to be reduced to the ellipsoid, but this is an average value for the entire traverse. If the lines have already been reduced to the ellipsoid, or this is not required, enter zero for the height.

(10) Users can elect to enter distances in feet by entering 0.3048 when prompted for FEET–METERS early in the program. To use meters, retain a value of 1, the program default. Other units can be used, if a conversion factor to meters is entered instead of the 0.3048. Long lines may need 0.30480061 for US Survey feet, while 0.3048 is used for International feet and approximate work in feet.

(11) The resulting misclosure is expressed in angular terms, in HP notation, and is the amount by which the final latitude and longitude miss the starting values. To convert these values to meters, multiply the number of seconds of latitude by 30, and the number of seconds of longitude by 30 cos ϕ. To get the values in feet multiply by 100 and 100 cos ϕ, respectively, instead. Note that these will be approximate.

Theory

This program uses the Gauss Mid-Latitude formulae to calculate the position of the point at the end of a line, given the starting position (in latitude, ϕ, and longitude, λ), the forward azimuth at the known point, and the distance (in either feet or meters). This is the classical 'forward' problem of geodetic line computation.

For the forward solution, the Gauss Mid-Latitude formulae require iteration to reach a solution, but are the simplest and quickest geodetic formulae for this type of task. The formulae are accurate to better than 0.001" in latitude and longitude (0.3 m, 0.1 ft), for worst-case lines up to 20 miles (32 km). If greater precision is required, use a different geodetic long-line formula (e.g., Robbins' or Rudoe's formulae).

Within the program, when a line's azimuth, θ, and distance, d, are first entered, they are converted to an initial, approximate latitude and longitude differences ($\Delta\phi_0$ and $\Delta\lambda_0$, respectively), using:

$$\Delta\phi_0 = \frac{d \cos\theta}{a} \qquad \Delta\lambda_0 = \frac{d \sin\theta}{a \cos\phi}$$

The mid-latitude of the line, ϕ_m, is computed using: $\phi_m = \phi + \frac{\Delta\phi}{2}$ [1]

The change in azimuth over the length of the line, $\Delta\theta$, is computed using:

$$\tan\frac{\Delta\theta}{2} = \tan\frac{\Delta\lambda}{2} \sin\phi_m \sec\frac{\Delta\phi}{2} \quad [2]$$

The radii of curvature in the meridian and prime vertical at the mid-point of the line, ρ_m and ν_m, respectively, are calculated, using:

$$\rho_m = \frac{a(1 - e^2)}{(1 - e^2 \sin\phi_m)^{\frac{3}{2}}} \quad [3]$$

$$\nu_m = \frac{a}{\sqrt{(1 - e^2 \sin\phi_m)}} \quad [4]$$

Traverse Program Using Latitude and Longitude

The differences in latitude and longitude are then calculated, using:

$$\Delta\phi = \frac{d}{\rho_m}\cos\left(\theta + \frac{\Delta\theta}{2}\right) \quad [5]$$

$$\Delta\lambda = \frac{d}{\nu_m}\sin\left(\theta + \frac{\Delta\theta}{2}\right)\sec\phi_m \quad [6]$$

The values from equations [5] and [6] are returned to equation [1], and the process re-iterated until the changes in latitude and longitude are too small to worry about. In most cases, three iterations are sufficient, but the program uses four iterations, just to be sure.

Distances are converted to ellipsoidal distances using the average height (h_m) for the region (or single line) entered. Entering zero for the height means the height scale factor has no effect on line length. The formula used is as follows, which is good to about 1 in 10,000 if the height is good to within 60 meters. The ellipsoidal distance is equal to the 'horizontal' distance times the height scale factor.

$$\text{Height Scale Factor} = 1 - (h_m \times 0.1571 \times 10^{-6})$$

At the end of each line, the program displays the latitude and longitude of the end point, as well as the forward azimuth of the line at this point. For most lines of any significant length, this will differ from the forward azimuth at the start of the line. By converting this forward bearing to a back bearing (by adding or subtracting 180°), an angle measured at the end point can be used to obtain the forward azimuth of the next line. This is done manually by the user, and is not included in the program, as it is not something that will be needed by all users.

The program stores the initial point values, so that a comparison can be made at the end of a traverse, if desired. The difference is calculated and shown to the user.

Sample Computation

Traverse Data and Results

Point	Line	Azimuth	Distance	Latitude	Longitude
A				40° 02' 25".000	–83° 01' 25".000
	A–B	47° 51' 27"	14,302.785		
B				40° 07' 35".189	–82° 53' 57".427
	B–C	140° 32' 56"	12,821.076		
C				40° 02' 14".812	–82° 48' 14".056
	C–D	235° 28' 29"	15,093.269		
D				39° 57' 37".772	–82° 56' 57".579
	D–E	274° 28' 12"	6,394.974		
E				39° 57' 53".807	–83° 01' 26".012
	E–A	0° 09' 50"	8,383.815		
A				40° 02' 25".001	–83° 01' 25".001

Traverse Program Using Latitude and Longitude

Misclosure	Latitude (ϕ) = 0° 00' 00".001	= 0".0007	= 0.021 m
	Longitude (λ) = –0° 00' 00".001	= –0".0008	= –0.018 m

The angular misclosure around this figure was initially 2", of which about 1" was spherical excess. The linear misclosure when computed by other means is very close to the figures above.

Note that the misclosure is determined from very small differences at the least significant end of a long number, and so is affected by the limitations in the calculator's internal precision. Calculation of the same traverse using different equipment (e.g., a spreadsheet) should give the same results for locations and azimuths, but there may be some small differences in the misclosure.

Storage Registers Used

A a = semi-major axis of ellipsoid = 6378137 m for WGS84/NAD83/GRS80

B $\Delta\phi/2$ = half the latitude difference for the line

C $\Delta\lambda/2$ = half the longitude difference for the line

D Distance, i.e., length of the line

E e^2 = eccentricity of ellipsoid = 0.006699438 for WGS84/NAD83/GRS80

F ϕ, latitude of starting point of each line

G $\Delta\theta/2$ = half the azimuth change for the line

H Height above ellipsoid, then the height-scale factor for line lengths. By default, height = 0

L λ, longitude of starting point of line

M unit to meters conversion factor; by default 1.0 for meters

N ν_m = radius of curvature of the ellipsoid in the prime vertical at the mid-point of the line

Q counter for calculation loop (1.004 by default), then forward azimuth of line

R ρ_m = radius of curvature of the ellipsoid in the meridian at the mid-point of the line

S $\Delta\phi$, the difference in latitude for the line

T $\Delta\lambda$, the difference in longitude for the line

X λ_0, longitude of the initial point of the traverse

Y ϕ_0, the latitude of the initial point of the traverse

Z θ, azimuth at the start of the line

Labels Used

Label **T**	Length = 258	Checksum = 4664
Label **U**	Length = 159	Checksum = 8D51
Label **V**	Length = 471	Checksum = 6AAF

Use the length (LN=) and Checksum (CK=) values to check if program was entered correctly. Use the sample computation to check proper operation after entry.

Traverse Program Using Latitude and Longitude

Running the Program

Press XEQ T

Prompt ENTER A briefly, then A?

Enter a value for ellipsoid semi-major axis, or ignore to retain default value.

Press R/S.

Prompt ENTER E briefly, then E?

Enter value for ellipsoid eccentricity, or ignore to retain default value.

Press R/S.

Prompt FEET-METERS briefly, then M?

Enter unit conversion value, or ignore to retain setting for distances in meters. Enter 0.3048 for International feet; 0.30480061 for US Survey feet; 0.201168 for chains, etc.

Press R/S.

Prompt ENTER H briefly, then H?

Enter average height above the ellipsoid for all lines to be processed. Ignore to retail default value of zero. Enter the height in the units you selected at the FEET-METERS prompt.

Press R/S.

Prompt ENTER LAT briefly, then F?

Enter the latitude (ϕ) of the starting point, in degrees, minutes and seconds, in HP notation (D.MMSSsss). Remember to include a negative sign, if in the southern hemisphere.

Press R/S.

Prompt ENTER LONG briefly, then L?

Enter the longitude (λ) of the starting point, in degrees, minutes and seconds, in HP notation (D.MMSSsss). Remember to include a negative sign, if in the western hemisphere.

Press R/S.

Top of Loop Point

Prompt AZIMUTH briefly, then Z?

Enter azimuth of the line at the starting point (θ) in HP notation.

Press R/S.

Prompt DISTANCE briefly, then D?

Enter the length of the line in the units previous selected.

Traverse Program Using Latitude and Longitude

Press R/S.

The calculator displays RUNNING…… for a while.

Prompt LATITUDE briefly, then F=

The latitude of the far end of the line is displayed, in HP notation.

Press R/S.

Prompt LONGITUDE briefly, then L=

The longitude of the far end of the line is displayed, in HP notation

Press R/S.

Prompt AZIMUTH briefly, then Q=

The forward azimuth of the line at the far end of the line is displayed, in HP notation.

Press R/S.

Prompt NEW LINE=1 briefly, then the calculator stops. Note that there is no “letter?” Prompt at this point. Just enter the required value directly to the X register.

To go on to do the next line in the traverse, enter 1. Pressing R/S will take you to the *Top of Loop Point*, above

To stop the traverse and compute the misclosure, enter 0.

Press R/S.

Prompt MISC LAT briefly, then B=

The misclosure in latitude (difference between start and end latitudes) is displayed, in HP notation.

Press R/S.

Prompt MISC LONG briefly, then C=

The misclosure in longitude (difference between start and end longitudes) is displayed, in HP notation.

Pressing R/S again will clear the flags and stop the program.

The misclosure in latitude will remain in the Y register, and the misclosure in longitude will remain in the X register, on the screen of the calculator. They can now be converted to whatever units interest you, after having converted them to decimal degrees using the → HR function.

Note that this misclosure is being determined from small differences at the least significant end of the calculator’s storage capability, and so will be approximate at best. It will give an idea of the degree of magnitude of the misclosure, rather than an exact amount.