

**NAME**

**addigits** - add individual digits in a number

**SYNOPSIS**

**addigits** [*options*] [*number*] ...

**DESCRIPTION**

Add individual digits in a *number*.

**OPTIONS**

- a** Additive Persistence, i.e.  
[http://en.wikipedia.org/wiki/Persistence\\_of\\_a\\_number](http://en.wikipedia.org/wiki/Persistence_of_a_number)
- l** Count circles in numbers (0,6,9 = 1; 8 = 2)
- L** Count loops in numbers (0,4,6,9 = 1; 8 = 2)
- m** Multiplicative Persistence, i.e.  
[http://en.wikipedia.org/wiki/Persistence\\_of\\_a\\_number](http://en.wikipedia.org/wiki/Persistence_of_a_number)
- p** Petals Around the Rose (3=2, 5=4, rest are zero), i.e.  
[http://en.wikipedia.org/wiki/Petals\\_Around\\_the\\_Rose](http://en.wikipedia.org/wiki/Petals_Around_the_Rose)
- r** Digital root or recursive, i.e.  
[http://en.wikipedia.org/wiki/Digital\\_root](http://en.wikipedia.org/wiki/Digital_root)
- D lvl** Debug level

**EXAMPLES**

Add digits:

```
$ addigits 1956
21
```

Digital root:

```
$ addigits -r 1956
3
```

Additive and Multiplicative Persistence:

```
$ addigits -a 1956
2
```

```
$ addigits -m 1957
3
```

Petals Around the Rose:

```
$ addigits -p 35264
6
```

Count circles in numbers:

```
$ addigits -l 80177437
```

addigits(1)

addigits(1)

3

Count loops in numbers:

```
$ addigits -L 80177437
```

4

**NAME**

**addletters** - Tool for diddling with letters

**SYNOPSIS**

**addletters** [*options*] *text* ...

**DESCRIPTION**

Tool for diddling with letters.

**OPTIONS**

- a** Use ASCII value of each letter (instead of 1-26)
- b** Print in binary
- B base**  
For **-r -a**, print the value from base N (N=2..32)
- d** Use delta between letters
- l** Output length
- L** Output length of Consonants/Vowels
- m** Multiply them together
- M modulus**  
Modulus to use
- n** Just print out the numerical value of each letter
- N** Shift the numbers: !=1, @==2, #==3, ... )==0
- p** When **-r**, val = val / position
- P** Use phone key value of each letter (instead of 1-26)
- r** Reverse: e.g. **addletters** 18 05 22 05 18 19 05
- R** Digital Root of letter (e.g. 12=3)
- s** Single char: rickrich == 18
- G** German Scrabble weights, add \*<n> bonus; i.e. start\*2
- S** Scrabble weights, add \*<n> bonus; i.e. start\*2
- T** Scrabble tiles
- t total** Output additional amount to add to get 'total'
- v vals** vals is:
  - a-z** A=1, B=2, ... Z=26
  - z-a** A=26, B=25, ... Z=1
  - kay** Kay (Francis Bacon) cipher.
  - l-za-k** A=16, B=17, ... K=26, L=1, M=2, ... Z=15
  - l-az-m** A=12, B=11, ... L=1, M=26, N=25, ... Z=13
  - lino** Linotype machine frequency of letters.
  - morse** Morse code frequency of letters.
  - oldphone**  
Use old phone keys, q=1 z=0

- qwerty** Keyboard qaz=1, wsx=2, ... p=0
- u** Swedish umlauts ÅäÖ instead of german umlauts ÄäÖ
- w** Print single words
- W** Words With Friends points
- x** Print in hex, not decimal
- 0** A=0, B=1, Z=25
- z** A=26, B=25, Z=1
- D lvl** Set Debug level [0]

**EXAMPLE**

Add the letters in 'geocaching':

```
$ addletters geocaching
72
```

Add reverse values by position:

```
$ addletters -r -p 20 30 18 36 70 24 140 64 81 190 33 12 39 112 75
TOFINDTHISCACHE
```

Add German umlauts:

```
$ addletters -n Ää ÖÖ
27 28 29 30 = 114
```

**SEE ALSO**

**lethist(1)**

**NAME**

**atomic-number-to-text** - Atomic numbers to text

**SYNOPSIS**

**atomic-number-to-text** [*options*] [*weight*] ...

**DESCRIPTION**

Atomic numbers to text from stdin. If two or more numbers are separated by +-\* / then calculate the expression. Convert from stdin or command line.

**OPTIONS**

- 1** First character
- 2** Second character
- D lvl** Debug level

**EXAMPLE**

Convert <http://coord.info/GC2HWNM>:

```
$ atomic-number-to-text -1
14 63 99 19 22 72 49 50 96 79 55 67 63 47 52 28 76 45 65 80 87 76
37 22 39 43 74 76 62 68 23 63 41 90 99 63 10 59 76 53 93 81 40 68
88 76 76 93 99 76 28 68 74 63 38 52 34 68 23 63 41 65 70 76 11 63
26 76 75 52 39 34 49 54 78 76 77 41 90 73 74 76 26 76 92 44 30 99
37 76
seekthiscacheatnorthfortytwoseventeenpointzerooneone \
westseventyonefortysixpointtwofourzero
```

**NAME**

**atomic-symbol-to-atomic-number** - Atomic Symbol to Atomic Number

**SYNOPSIS**

*atomic-symbol-to-atomic-number* [*options*] [*symbol*] ...

**DESCRIPTION**

Atomic Symbol to Atomic Number. Convert from stdin or command line.

You can use lowercase for the leading letter; it will convert it to uppercase.

**OPTIONS**

**-D lvl** Debug level

**EXAMPLE**

Convert:

```
$ atomic-symbol-to-atomic-number Na Cl
11 17
```

**NAME**

**atomic-symbol-to-period-or-group** - Atomic Symbol to Period or Group

**SYNOPSIS**

```
atomic-symbol-to-period-or-group [options] [symbol] ...
```

**DESCRIPTION**

Atomic Symbol to a period (row) or a group (column). It will convert from stdin or command line.

You can use lowercase for the leading letter; it will convert it to uppercase.

**OPTIONS**

**-p**     Period (row)  
**-g**     Group (column)  
**-D lvl**  Debug level

**EXAMPLE**

Convert:

```
$ atomic-symbol-to-period-or-group -p H Be Na K  
1 2 3 4
```

```
$ atomic-symbol-to-period-or-group -g v mo re hs  
5 6 7 8
```

**NAME**

**baconian2text** - Convert baconian to text

**SYNOPSIS**

**baconian2text** [*options*] [*baconian*]

**DESCRIPTION**

Convert *baconian* to text. It can read from stdin or command line arguments. 0/1, a/b, or A/B can be used.

**EXAMPLES**

Convert:

```
$ baconian2text -i AABBBAAAAAABBBBAABBBABABBA
happy
```

```
$ echo 'AABBBAAAAAABBBBABB BBBBAAA' | baconian2text
happy
```

**OPTIONS**

**-s**      Swap A/B  
**-i**      Convert i=j and u=v  
**-D lvl**   Debug level

**SEE ALSO**

[http://en.wikipedia.org/wiki/Bacon%27s\\_cipher](http://en.wikipedia.org/wiki/Bacon%27s_cipher)

**NAME**

**balanced-ternary** - Convert balanced-ternary to/from decimal

**SYNOPSIS**

*balanced-ternary* [*options*] [*ternary*] ...

**DESCRIPTION**

Converts *balanced-ternary* to/from decimal. Can use command line arguments or read from stdin.

When converting to decimal, the following letters work:

MINUS	ZERO	PLUS	
-----	-----	-----	
-	0	+	Normal
T	0	1	Wikipedia page
M	Z	P	M is MINUS, Z is ZERO, P is PLUS
Z	M	C	rot-13 of the above

**OPTIONS**

**-e** Encode decimal to *balanced-ternary*

**-D lvl** Debug level

**EXAMPLES**

Convert to decimal, in rot13 and +0-:

```
$ balanced-ternary CZCMCCCMMZCZZMZMCCMMZ +-+0+++00-+---0-0++00-
8321925851
8321925851
```

Convert from decimal:

```
$ balanced-ternary -e 0 1 2 3 4 5 16 -436 523
0
+
+-
+0
++
+--
+--+
-+-0--
+-0++0+
```

**SEE ALSO**

[https://en.wikipedia.org/wiki/Balanced\\_ternary](https://en.wikipedia.org/wiki/Balanced_ternary)

[http://rosettacode.org/wiki/Balanced\\_ternary](http://rosettacode.org/wiki/Balanced_ternary)

**NAME**

**bing2ll** - Bing maps quadkey string to lat/lon

**SYNOPSIS**

**bing2ll** [*options*] *quadkey* ...

**DESCRIPTION**

Bing maps *quadkey* string to lat/lon. Quadkey is a version of Quad Trees.

**OPTIONS**

**-e** Elliptical earth

**-D lvl** Debug level

**EXAMPLE**

Convert <http://coord.info/GC5K63B>:

```
$ bing2ll 0212113121222020033110
0212113121222020033110 53.558640 -113.552070 n53.33.518 w113.33.124
```

NOTE: there should be a 30 feet slop in the geochecker answer. But it is an exact answer if you use GTools for the iphone. For my program, this should be:

```
$ bing2ll 0212113121222020033101333
0212113121222020033101333 53.558618 -113.552118 n53.33.517 w113.33.127
```

**SEE ALSO**

<https://msdn.microsoft.com/en-us/library/bb259689.aspx> <http://intepid.com/posts/484>

**NAME****braille2text** - braille to text translator**SYNOPSIS****braille2text** [*options*] [*braille-cell*] ...**DESCRIPTION**Takes an encoding of a braille **cell**(s) and translates that into text.

Braille cell:

row1	1 4
row2	2 5
row3	3 6

Also takes a two digit octal representation.

**OPTIONS****-D lvl** Debug level**EXAMPLE**

Decode north coordinates:

```
$ echo "1-3-4-5 3-4-5-6 1-5 1 1-4-5 2-4 1-2-4 1-2-4-5 2-4-5" |
  braille2text
n5149670
```

```
$ echo "2-4-5-6 3-4-5-6 1-2-5 2-4-5 2-4-5 1-2 1 1-2-4" |
  braille2text
w800216
```

```
$ braille2text 56 52 72 36 62 64 52 72 36 57 64 52 51 72
northfortyfour
```

**NAME**

**decimal2cryptogram** - Anything to cryptogram

**SYNOPSIS**

**decimal2cryptogram** [*options*] [*thing*] ...

**DESCRIPTION**

Decimal to cryptogram. Actually "decimal" should be "anything" to cryptogram.

**EXAMPLE**

Processing from stdin:

```
$ decimal2cryptogram
81 56 57 58 54 72 55 72 58 66 56 81 41 53 41
42 57 41 41 63 41 55 42 54 58 41 41 81 46 56
55 81 58 81 55 81 41 41 55 42 54 58 48 41 57 56
ABCDEFGHIJFDHBAIJKCIILIGKEDI IAMBGADAGAIIGKEDNICO
```

Then, go online to <http://www.quipqiup.com/> .

Hexadecimal:

```
$ decimal2cryptogram
2c 7f 24 24 5a 20 7c 26 6a 07 63 5a 6a 20
4e 20 63 5a 6a 67 1c 6a 4e 7c 6a 1b 6a 01 01 6a
41 01 64 49 54 34 6a 5c 6a 28 14 01 6a 61 0f 64
14 28 34
ABCCDEFGHIJDHEKEJDHLMHKFHNHOOHPOQRSTHUVVWOHXYQWVT
```

Spaces:

```
$ decimal2cryptogram -s 00
18 24 34 47 00 34 47 00 12 00 35 34
ABCD CD E FC
```

**OPTIONS**

**-s *thing*** Spaces are "*thing*"

**-D *lvl*** Debug level

**SEE ALSO**

<http://www.quipqiup.com/>

**NAME**

**fibonacci-coding** - Decode/encode a fibonacci coding to/from a number

**SYNOPSIS**

**fibonacci-coding** [*options*] *string* ...

**DESCRIPTION**

In mathematics and computing, Fibonacci coding is a universal code which encodes positive integers into binary code words. It is one example of representations of integers based on Fibonacci numbers. Each code word ends with

**OPTIONS**

**-e** Encode decimal to fibonacci coding

**-v** Verbose

**-D lvl** Debug level

**EXAMPLE**

Convert <http://coord.info/GC34BVW>:

```
$ fibonacci-coding \  
10101001001001001001000100010001110000100100010101010010100010100011  
4128162 11202427
```

```
$ fibonacci-coding -e 4128162 11202427  
10101001001001001001000100010001110000100100010101010010100010100011
```

**SEE ALSO**

[https://en.wikipedia.org/wiki/Fibonacci\\_coding](https://en.wikipedia.org/wiki/Fibonacci_coding)

<http://wiki.tcl.tk/12324>

**NAME**

**geo-2gpsdrive** - Enter a file of waypoints into the GpsDrive SQL database.

**SYNOPSIS**

**geo-2gpsdrive** [*options*] *waypoint-file*

**geo-2gpsdrive** [*options*] *waypoint-file latitude longitude*

**DESCRIPTION**

Enter a file of waypoints into the GpsDrive SQL database (if version of gpsdrive is 2.09 or less) OR sqlite3 database (if version of gpsdrive is 2.10 or greater).

This is useful if you have a file of waypoints from geo-nearest that you need to convert into Gpsdrive format plus one or more other formats, such as Cetus plus GpsDrive. Gpsbabel currently doesn't know how to enter waypoints directly into an SQL database (and its not clear to me whether it should be taught how to do this or not).

**OPTIONS**

**-s** Output short names for the caches (gpsbabel option)

**-r radius**

Display only caches with radius (e.g. **-r 25M**)

**-i format**

Input format, **-o?** for possibilities [tabsep]

**-S** Enter waypoints into SQL database

**-d** For **-S**, just delete selected records

**-P** For **-S**, purge all records of type **-t Geocache\***

**-t type** The waypoint type [Geocache]

**-V gpsver**

Version of gpsdrive (2.09 or 2.10+) [2.09]

**-D lvl** Debug level [0]

**-U** Retrieve latest version of this script

Defaults can also be set with variables in file \$HOME/.georc:

```
LAT=latitude;          LON=logitude;
OUTFMT=format;        BABELFLAGS=-s;
SQLUSER=gast;         SQLPASS=gast;          SQLDB=geoinfo;
```

**EXAMPLES**

Display shortnames:

```
geo-2gpsdrive -s caches.tabsep
```

Add caches to a GpsDrive SQL database

```
geo-2gpsdrive -s -S caches.tabsep
```

Purge the existing SQL database of all geocaches, then enter new ones:

```
geo-2gpsdrive -S -P -s caches.tabsep
```

**SEE ALSO**

geo-newest, geo-found, geo-placed, geo-nearest, <http://geo.rkkda.com/>

**NAME**

**geo-2tangogps** - Enter a file of waypoints into the tangogps SQL database.

**SYNOPSIS**

**geo-2tangogps** [*options*] *waypoint-file*

**geo-2tangogps** [*options*] *waypoint-file* *latitude* *longitude*

**DESCRIPTION**

Enter a file of waypoints into the tangogps or FoxtrotGPS SQL database.

This is useful if you have a file of waypoints from geo-nearest that you need to convert into tangogps format plus one or more other formats, such as Cetus plus tangogps. Gpsbabel currently doesn't know how to enter waypoints directly into an SQL database (and its not clear to me whether it should be taught how to do this or not).

**OPTIONS**

**-s** Output short names for the caches (gpsbabel option)

**-r radius**

Display only caches with radius (e.g. **-r 25M**)

**-f** Use FoxtrotGPS instead of tangoGPS for the DB file. Right now, just changes SQLDB to ~/.foxtrotgps/poi.db

**-i format**

Input format, **-o?** for possibilities [tabsep]

**-S** Enter waypoints into SQL database

**-d** For **-S**, just delete selected records

**-P** For **-S**, purge all records of type **-t** Geocache\*

**-t type** The waypoint type [Geocache]

**-X term**

Exclude caches with 'term' [Geocache Found|**-ifound**]

**-D lvl** Debug level [0]

**-U** Retrieve latest version of this script

Defaults can also be set with variables in file \$HOME/.georc:

```
LAT=latitude;          LON=logitude;
OUTFMT=format;        BABELFLAGS=-s;
SQLUSER=gast;         SQLPASS=gast;          SQLDB=~/.tangogps/poi.db
;
```

**EXAMPLES**

Display shortnames:

```
geo-2tangogps -s caches.tabsep
```

Add caches to a tangogps SQL database

```
geo-2tangogps -s -S caches.tabsep
```

Purge the existing SQL database of all geocaches, then enter new ones:

```
geo-2tangogps -S -P -s caches.tabsep
```

**SEE ALSO**

geo-newest, geo-found, geo-placed, geo-nearest, <http://geo.rkkda.com/>

**NAME**

**geo-additional** - Fetch additional waypoints

**SYNOPSIS**

**geo-additional** [*options*] *gid* ...

**DESCRIPTION**

Fetch additional waypoints from a gc id.

**EXAMPLES**

Fetch extra waypoints from FTF HOUNDS MN STYLE - Hal-oween:

```
$ geo-additional GC30V8T
geo-waypoint $FLAGS N44.54.103 W093.34.027 MU30V8T
geo-waypoint $FLAGS N44.54.094 W093.33.896 S130V8T
geo-waypoint $FLAGS N44.54.072 W093.34.100 S230V8T
geo-waypoint $FLAGS N44.54.172 W093.34.070 S330V8T
geo-waypoint $FLAGS N44.54.247 W093.34.079 S430V8T
geo-waypoint $FLAGS N44.54.242 W093.34.050 S530V8T
geo-waypoint $FLAGS N44.54.219 W093.33.973 S630V8T
geo-waypoint $FLAGS N44.54.190 W093.33.947 S730V8T
geo-waypoint $FLAGS N44.54.185 W093.33.936 S830V8T
geo-waypoint $FLAGS N44.54.142 W093.33.766 S930V8T
geo-waypoint $FLAGS N44.54.139 W093.33.765 TE30V8T
```

**OPTIONS**

**-D lvl** Debug level

**NAME**

**geo-algebraic-expressions** - Solve a system of algebraic expressions

**SYNOPSIS**

**geo-algebraic-expressions** [*options*] *eqn* ...

**DESCRIPTION**

Solve a system of algebraic expressions. Needs sympy installed. It's really a helper script to put the *'eqn'*s into Python's syntax.

**OPTIONS**

**-i** Use integers only

**-D lvl** Debug level [0]

**EXAMPLES**

Solve  $a+b=10$ ,  $a-b=7$ :

```
$ geo-algebraic-expressions a+b=10 a-b=7
{b: 3/2, a: 17/2}
```

Solve <http://coord.info/GC5NTN7>:

```
geo-algebraic-expressions -i 22+c=a 23+c=b a+b=111 d+e=22 e+f=c \
f+g=23 3+h=d h+i=e i+j=f j+6=g
{j: 1, c: 33, f: 16, a: 55, d: 5, i: 15, b: 56, e: 17, g: 7, h: 2}
```

**SEE ALSO**

<http://www.sympy.org/> <http://docs.sympy.org/dev/install.html>

**NAME**

**geo-alphabetic** - solve a math puzzle in which letters stand for digits

**SYNOPSIS**

**geo-alphabetic** [*options*] 'string'

**DESCRIPTION**

The four basic operators (+ - \* /) and exponentiation (\*\*) are understood. Assignment is with double equals (==). A single equals (=) is converted to double equals. Likewise, exponentiation with a caret (^) is converted to double star (\*\*). This is because its a python script instead of everything (C, awk, perl, ...) else!

**OPTIONS**

**-1**     Use 1-9 instead of 0-9  
**-D**     lvl    Debug level [0]

**EXAMPLE**

Subtraction:

```
$ geo-alphabetic 'hscoocoh - ddsgooc0 = oifidsh'
hscoocoh - ddsgooc0 == oifidsh
83255258 - 77305520 == 5949738
83255258 - 77345520 == 5909738
```

Long Division (GC369B3):

```
for i in 1 2 3 4 5 6 7 8 9; do
  for j in 1 2 3 4 5 6 7 8 9; do
    geo-alphabetic "infirmary/rummy == iris+(${i}${j}rou/rummy)"
  done
done
for i in 1 2 3 4 5 6 7 8 9; do
  geo-alphabetic "interests/store == none+(${i}srst/store)"
done
```

**SEE ALSO**

<http://www.math.ubc.ca/~israel/applet/metic/metic.html>  
[tate.com/recipes/576615-alphametics-solver/](http://tate.com/recipes/576615-alphametics-solver/)

<http://code.activestatem.com/recipes/576615-alphametics-solver/>



**NAME**

**geo-battleship** - Map the geocheck.org battleship locations

**SYNOPSIS**

**geo-battleship** [*options*] *lat lon less-than greater-than ...*

**DESCRIPTION**

Map the geocheck.org battleship locations. *Lat/lon* is dotted MinDec location. The *less-than* is within meters or feet (**-f**) distance. The *greater-than* is outside meters or feet (**-f**) distance.

**OPTIONS**

**-f**      Feet instead of meters

**-D lvl**   Debug level

**EXAMPLE**

Solve <http://coord.info/GC5MB36>:

```
$ geo-battleship S38.10.565 E146.17.895 4000 3000 \  
  s38.10.606 e146.15.480 2000 1000 \  
  ...
```

**SEE ALSO**

**geo-map(1)**

<http://geocheck.org/>

**NAME**

**geo-char-at** - pick the char at position "n"

**SYNOPSIS**

```
geo-char-at [options] position text ...
geo-char-at [options] < file
geo-char-at [options] -r "values" text ...
```

**DESCRIPTION**

Pick the char at *position* "*position*". "*position*" can be a number, or "page"-*line*"-*char*" in which case "*position*" equals "char", or "m" for the midpoint.

Works for one-time-pad ciphers, where page-line-char is given.

**OPTIONS**

-r "values" Do a range of values. A letter equals A=1, B=2, ...  
 -D *lvl* Debug level

**EXAMPLES**

Some examples:

```
$ geo-char-at 12 "said to her, Do not be afraid, Mary"
n
```

```
$ geo-char-at 14 After they had heard the king
e
```

```
$ geo-char-at 1-2-21 "In the sixth month of Elizabeth's pregnancy"
z
```

```
$ geo-char-at -r "1 3 6" abcdef
acf
```

```
$ geo-char-at -r "/96/95//69/68/./560/559/558" `cat tau`
4323455
```

```
$ geo-char-at -r "shockvalue" 05818841893035873408602622
0188300068
```

```
$ geo-char-at m KnOcK kNoCk WhO iS tHeRe?
w
```

**NAME**

**geo-circles** - Compute the intersection of two circles on the earth

**SYNOPSIS**

**geo-circles** [*options*] *lat1 lon1 radius1 lat2 lon2 radius2*

**DESCRIPTION**

Compute the intersection of two circles on the earth.

lat/lon can be specified in DegDec or dotted MinDec format. radius is in meters (m) or feet (ft) or miles (mi).

N.B. this program was inspired by Rock Johnson's "Gee" series of math caches. Dyl1231, Seabiskit, and I enjoy these very much. Thanks RJ!

**OPTIONS**

**-D lvl** Debug level

**EXAMPLES**

# DegDec input...

```
$ geo-circles -- 44.92592 -93.41415 307      44.92392 -93.41377 114
p3a = 44.923176 -93.414810      n44.55.391 w93.24.889
p3b = 44.923455 -93.412518      n44.55.407 w93.24.751
```

# MinDec input...

```
$ geo-circles -- 44.55.435 -93.24.826 114    44.55.435 -93.24.645 150
p3a = 44.923455 -93.412505      n44.55.407 w93.24.750
p3b = 44.924445 -93.412513      n44.55.467 w93.24.751
```

**NAME**

**geo-clock-angle** - compute the clock angle or the time

**SYNOPSIS**

**geo-clock-angle** [*options*] *time\_or\_angle* ...

**DESCRIPTION**

Compute the clock angle. You can give the time or the angle.

**OPTIONS**

**-h** Use angle of hour hand only

**-D lvl** Debug level

**EXAMPLE**

Compute using the time:

```
$ geo-clock-angle 3:15 15:15
3:15    7.5    352.5
15:15   367.5  -7.5
```

Compute using the angle:

```
$ geo-clock-angle 105 285 97.5 285 105 285
105     2:30    9:30
285     3:30    8:30
97.5    5:45    6:15
285     3:30    8:30
105     2:30    9:30
285     3:30    8:30
```

Compute using the hour hand only:

```
$ geo-clock-angle -h 3:15 8:57:05
TIME           HOURS           MINUTES          (360+M-H)%360
    3:15           97.5000000      90.0000000      -7.5000000
    8:57:05       268.5416667    342.5000000     73.9583333
```

**NAME**

**geo-code** - Geocode an address into a lat/lon

**SYNOPSIS**

```
geo-code [options] address citystate_or_zip [country]
geo-code [options] "" citystate_or_zip [country]
geo-code [options] tele-phone-number
```

**DESCRIPTION**

```
geo-code [options] address citystate_or_zip [country]
```

Convert (geocode) a street address into a latitude/longitude.

```
geo-code [options] "" citystate_or_zip [country]
```

Convert (geocode) a place name into a latitude/longitude.

```
geo-code [options] tele-phone-number
```

Convert (geocode) a phone number into a latitude/longitude.

In either case, the output can be formatted to any of the output file types that gpsbabel supports, or directly imported into the GpsDrive MySQL waypoint database.

Requires:

**curl** <http://curl.haxx.se/>

**gpsbabel**  
<http://gpsbabel.sourceforge.net/>

**OPTIONS****-o format**

Output format, **-o?** for possibilities [gpsdrive] plus "gpsdrive.sql" for direct insertion into MySQL DB plus "degdec" for just Lat.fraq<tab>Long.fraq. plus "mindec" for just DD MM.MMM<tab>DD MM.MMM. plus "map[,geo-map-opts]" to display a geo-map.

**-n name**

The waypoint name, e.g. Bob's House. The default is the street *address*. Percent escapes can be used: %d/%D for DegDec lat/lon, %m/%M for MinDec lat/lon, %a for *address*, %c for *citystate\_or\_zip*, %p for phone

**-s** Output shortened names (a gpsbabel option)

**-t type** The waypoint type, e.g. house, cache, bar [new]

**-q** Quiet. Do not output *address* confirmation on stderr.

**-S** Alias for **-o gpsdrive.sql**

**-a** For SQL, delete existing record only if it matches all fields. Otherwise, delete it if it matches just the name and the type.

**-D level**

Debug level

**-U** Retrieve latest version of this script

**COUNTRIES**

at, be, ca, dk, fr, de, it, lu, nl, es, ch, uk, us, fi, no, pt, se

**EXAMPLES**

Geocode...

```
$ geo-code "123 AnyStreet" 12345
123AnyStreet 42.81020 -73.95070 new

$ geo-code -t house "123 AnyStreet" 12345
123AnyStreet 42.81020 -73.95070 house

$ geo-code -n "Bob's House" -t house "123 AnyStreet" 12345
BobsHouse 42.81020 -73.95070 house

$ geo-code -S -n "Bob" -t house "123 AnyStreet" 12345
[waypoint is added to GpsDrive MySQL database]

$ geo-code 901-555-1212
123AnyStreet 42.81020 -73.95070 new

$ geo-code "Schlossplatz 10" "76131 Karlsruhe" de
Schlossplatz10 49.01294 08.40584 new

$ geo-code "" "Mankato, MN"
MankatoMN 44.16562 -94.00130 new
```

**SEE ALSO**

geo-nearest, geo-waypoint, geo-pg,  
<http://www.rubygeocoder.com/>  
<http://geo.rkkda.com/>

**NAME**

**geo-compare-images** - Compare two images

**SYNOPSIS**

**geo-compare-images** [*options*] *image1 image2*

**DESCRIPTION**

Compare two images using difference and then divide. Display with 'xv', 'display', 'eog', 'Preview' (Mac OS) or 'mspaint.exe' (cygwin).

**OPTIONS**

**-D lvl** Debug level

**EXAMPLE**

Compare:

```
$ geo-compare-images xxx.bmp yyy.png
```

**SEE ALSO**

ImageMagick, xv

**NAME**

**geo-coords** - Convert lat/lon from one format to another

**SYNOPSIS**

**geo-coords** [*options*] *latitude longitude*

**DESCRIPTION**

Convert lat/lon from one format to another. Lat/Lon may be in DegDec, MinDec, or DMS formats.

Acceptable formats for lat/lon are:

- 93.49130            DegDec (decimal degrees)
- W93.49130           DegDec (decimal degrees)
  
- "-93 29.478"        MinDec (decimal minutes)
- "W93 29.478"        MinDec (decimal minutes)
- 93.29.478           MinDec (decimal minutes)
- W93.29.478           MinDec (decimal minutes)
  
- "-93 45 30"         DMS (degrees, minutes, seconds)

**OPTIONS**

- a**     Antipod (opposite side)
- d**     Output DegDec only
- m**     Output MinDec only
- l**     Lat only
- L**     Long only

**EXAMPLE**

Convert DegDec:

```
$ geo-coords n45.12345 w93.12345
45.12345 -93.12345
N45.12345 W93.12345
N45 7' 24.420000" W93 7' 24.420000"
N45.07.407 W93.07.407
```

Convert to antipod:

```
$ geo-coords -a s38.32.329 e58.13.715
38.538816 121.771417
N38.538816 E121.771417
N38 32' 19.737600" E121 46' 17.101200"
N38.32.329 E121.46.285
```

**SEE ALSO**

ll2maidenhead, ll2osg, ll2rd, ll2usng, ll2utm, maidenhead2ll, rd2ll, usng2ll, utm2ll

**NAME**

**geo-correct-coords** - Correct the coords of **cache(s)** on the gc.com site

**SYNOPSIS**

**geo-correct-coords** [*options*] [*gcid lat lon*] ...

**DESCRIPTION**

Correct the coordinates of **cache(s)** on the gc.com site. It can take arguments or read from a file. It can work on traditional, multi, wherigo, mystery, etc., caches, unlike the GC interface.

**EXAMPLES**

Correct GC288HG:

```
$ geo-correct-coords GC288HG n44.51.202 w93.45.232
```

Correct GC numbers in ~/.geo-mystery:

```
$ geo-correct-coords < ~/.geo-mystery
```

**OPTIONS**

**-D lvl** Debug level

**SEE ALSO**

<http://geo.rkkda.com/>

**NAME**

**geo-count** - Count geocache finds or logs

**SYNOPSIS**

**geo-count** [*options*] *user* ...

**geo-count** [*options*] *GCxxxx* ...

**DESCRIPTION**

**geo-count** [*options*] *user* ...

Report and count geocache finds for "user". "user" can be a user name or a user account number.

**geo-count** [*options*] *GCxxxx* ...

Count number of log entries for a cache.

Requires: A free login at <http://www.geocaching.com>.

**curl** <http://curl.haxx.se/>

**OPTIONS**

- b** Include benchmarks in count
- c** Remove cookie file when done
- o** Include counts of items owned
- s** Only print one output line with totals
- h** Print header line
- t** Include counts of travel bugs
- u username**  
Username for <http://www.geocaching.com>
- p password**  
Password for <http://www.geocaching.com>
- D lvl** Debug level [0]
- U** Retrieve latest version of this script

Defaults can also be set with variables in file `$HOME/.georc`:

```
PASSWORD=password; USERNAME=username;
```

**EXAMPLES**

Report cache finds by type for *user* 'Jeremy':

```
geo-count Jeremy
```

Report totals (found, placed, bugs, bugged) for *user* number 3:

```
geo-count -s 3
```

**SEE ALSO**

geo-usernum, geo-found, <http://geo.rkkda.com/>

**NAME**

**geo-countries-states** - List of Countries and States

**COUNTRIES**

Here is a list of the countries used by **geo-newest**, **geo-demand** and **geo-pqs**.

<b>c-code</b>	<b>2letter</b>	<b>3letter</b>	<b>Name</b>
12	.af	.afg	afghanistan
272	.ax	.ala	aland islands
244	.al	.alb	albania
14	.dz	.dza	algeria
245	.as	.asm	american samoa
16	.ad	.and	andorra
17	.ao	.ago	angola
246	.ai	.aia	anguilla
18	.aq	.ata	antarctica
13	.ag	.atg	antigua and barbuda
19	.ar	.arg	argentina
15	.am	.arm	armenia
20	.aw	.abw	aruba
3	.au	.aus	australia
227	.at	.aut	austria
21	.az	.aze	azerbaijan
23	.bs	.bhs	bahamas
29	.bh	.bhr	bahrain
24	.bd	.bgd	bangladesh
25	.bb	.brb	barbados
40	.by	.blr	belarus
4	.be	.bel	belgium
31	.bz	.blz	belize
26	.bj	.ben	benin
27	.bm	.bmu	bermuda
30	.bt	.btu	bhutan
32	.bo	.bol	bolivia
275			bonaire
234	.ba	.bih	bosnia and herzegovina
33	.bw	.bwa	botswana
247	.bv	.bvt	bouvet island
34	.br	.bra	brazil
248	.io	.iot	british indian ocean territories
39	.vg	.vgb	british virgin islands
36	.bn	.brn	brunei
37	.bg	.bgr	bulgaria
216	.bf	.bfa	burkina faso
35	.bi	.bdi	burundi
42	.kh	.khm	cambodia
43	.cm	.cmr	cameroon
5	.ca	.can	canada
239	.cv	.cpv	cape verde
44	.ky	.cym	cayman islands
46	.cf	.caf	central african republic
249	.td	.tcd	chad
6	.cl	.chl	chile

47	.cn	.chn	china
250	.cx	.cxr	christmas island
251	.cc	.cck	cocos (keeling) islands
49	.co	.col	colombia
50	.km	.com	comoros
51	.cg	.cog	congo
48	.ck	.cok	cook islands
52	.cr	.cri	costa rica
53	.hr	.hrv	croatia
238	.cu	.cub	cuba
54			curacao
55	.cy	.cyp	cyprus
56	.cz	.cze	czech republic
257	.cd	.cod	democratic republic of the congo
57	.dk	.dnk	denmark
58	.dj	.dji	djibouti
59	.dm	.dma	dominica
60	.do	.dom	dominican republic
252			east timor
61	.ec	.ecu	ecuador
63	.eg	.egy	egypt
64	.sv	.slv	el salvador
62	.gq	.gnq	equatorial guinea
65	.er	.eri	eritrea
66	.ee	.est	estonia
67	.et	.eth	ethiopia
69	.fk	.flk	falkland islands
68	.fo	.fro	faroe islands
71	.fj	.fji	fiji
72	.fi	.fin	finland
73	.fr	.fra	france
70	.gf	.guf	french guiana
74	.pf	.pyf	french polynesia
253	.tf	.atf	french southern territories
75	.ga	.gab	gabon
76	.gm	.gmb	gambia
78	.ge	.geo	georgia
79	.de	.deu	germany
254	.gh	.gha	ghana
80	.gi	.gib	gibraltar
82	.gr	.grc	greece
83	.gl	.grl	greenland
81	.gd	.grd	grenada
77	.gp	.glp	guadeloupe
229	.gu	.gum	guam
84	.gt	.gtm	guatemala
86	.gg		guernsey
255	.gn	.gin	guinea
85	.gw	.gnb	guinea-bissau
87	.gy	.guy	guyana
89	.ht	.hti	haiti
256	.hm	.hmd	heard island and mcdonald islands
90	.hn	.hnd	honduras

91	.hk	.hkg	hong kong
92	.hu	.hun	hungary
93	.is	.isl	iceland
94	.in	.ind	india
95	.id	.idn	indonesia
96	.ir	.irn	iran
97	.iq	.irq	iraq
7	.ie	.irl	ireland
243	.im		isle of man
98	.il	.isr	israel
99	.it	.ita	italy
100			ivory coast
101	.jm	.jam	jamaica
104	.jp	.jpn	japan
102	.je		jersey
103	.jo	.jor	jordan
106	.kz	.kaz	kazakhstan
107	.ke	.ken	kenya
109	.ki	.kir	kiribati
241	.kw	.kwt	kuwait
108	.kg	.kgz	kyrgyzstan
110	.la	.lao	laos
111	.lv	.lva	latvia
113	.lb	.lbn	lebanon
114	.ls	.lso	lesotho
115	.lr	.lbr	liberia
112	.ly	.lby	libya
116	.li	.lie	liechtenstein
117	.lt	.lyu	lithuania
8	.lu	.lux	luxembourg
258	.mo	.mac	macau
125	.mk	.mkd	macedonia
119	.mg	.mdg	madagascar
129	.mw	.mwi	malawi
121	.my	.mys	malaysia
124	.mv	.mdv	maldives
127	.ml	.mli	mali
128	.mt	.mlt	malta
240	.mh	.mhl	marshall islands
122	.mq	.mtq	martinique
123	.mr	.mrt	mauritania
134	.mu	.mus	mauritius
259	.yt	.myt	mayotte
228	.mx	.mex	mexico
242	.fm	.fsm	micronesia
237	.md	.mda	moldova
130	.mc	.mco	monaco
131	.mn	.mng	mongolia
274	.me		montenegro
135	.ms	.msr	montserrat
132	.ma	.mar	morocco
133	.mz	.moz	mozambique
136	.mm	.mmr	myanmar

137	.na	.nam	namibia
138	.nr	.nru	nauru
140	.np	.npl	nepal
141	.nl	.nld	netherlands
148	.an	.ant	netherlands antilles
142	.kn		nevis and st kitts
41	.nc	.ncl	new caledonia
9	.nz	.nzl	new zealand
144	.ni	.nic	nicaragua
143	.ne	.ner	niger
145	.ng	.nga	nigeria
149	.nu	.niu	niue
260	.nf	.nfk	norfolk island
146	.kp	.prk	north korea
236	.mp	.mnp	northern mariana islands
147	.no	.nor	norway
150	.om	.omn	oman
151	.pk	.pak	pakistan
261	.pw	.plw	palau
276	.ps		palestine
152	.pa	.pan	panama
156	.pg	.png	papua new guinea
262	.py	.pry	paraguay
153	.pe	.per	peru
154	.ph	.phl	philippines
155	.pn	.pcn	pitcairn islands
158	.pl	.pol	poland
159	.pt	.prt	portugal
226	.pr	.pri	puerto rico
160	.qa	.qat	qatar
161	.re	.reu	reunion
162	.ro	.rom	romania
163	.ru	.rus	russia
164	.rw	.rwa	rwanda
277			saba
171	.sh	.shn	saint helena
264	.kn	.kna	saint kitts and nevis
173	.lc	.lca	saint lucia
217	.ws	.wsm	samoa
183	.sm	.smr	san marino
176	.st	.stp	sao tome and principe
166	.sa	.sau	saudi arabia
167	.sn	.sen	senegal
222	.rs		serbia
168	.sc	.syc	seychelles
178	.sl	.sle	sierra leone
179	.sg	.sgp	singapore
182	.sk	.svk	slovakia
181	.si	.svn	slovenia
184	.sb	.slb	solomon islands
185	.so	.som	somalia
165	.za	.zaf	south africa
267	.gs	.sgs	south georgia and sandwich islands

180	.kr	.kor	south korea
278			south sudan
186	.es	.esp	spain
187	.lk	.lka	sri lanka
169	.bl		st barthelemy
170			st eustatius
172	.kn		st kitts
175	.pm	.spm	st pierre miquelon
177			st vince grenadines
174	.mf		st. martin
188	.sd	.sdn	sudan
189	.sr	.sur	suriname
268	.sj	.sjm	svalbard and jan mayen
190	.sz	.swz	swaziland
10	.se	.swe	sweden
192	.ch	.che	switzerland
193	.sy	.syr	syria
194	.tw	.twn	taiwan
195	.tj	.tjk	tajikistan
196	.tz	.tza	tanzania
198	.th	.tha	thailand
200	.tg	.tgo	togo
269	.tk	.tkl	tokelau
201	.to	.ton	tonga
202	.tt	.tto	trinidad and tobago
203	.tn	.tun	tunisia
204	.tr	.tur	turkey
199	.tm	.tkm	turkmenistan
197	.tc	.tca	turks and caicos islands
205	.tv	.tuv	tuvalu
208	.ug	.uga	uganda
207	.ua	.ukr	ukraine
206	.ae	.are	united arab emirates
11	.uk	.gbr	united kingdom
210	.uy	.ury	uruguay
270	.um	.umi	us minor outlying islands
235	.vi	.vir	us virgin islands
211	.uz	.uzb	uzbekistan
212	.vu	.vut	vanuatu
213	.va	.vat	vatican city state
214	.ve	.ven	venezuela
215	.vn	.vnm	vietnam
218	.wf	.wlf	wallis and futuna islands
271	.eh	.esh	western sahara
220	.ye	.yem	yemen
224	.zm	.zmb	zambia
225	.zw	.zwe	zimbabwe

**STATES**

Here is a list of the states used by **geo-newest**, **geo-demand** and **geo-pqs**.

<b>s-code</b>	<b>2letter</b>	<b>Name</b>
189		abruzzo
162		acre
454		aguascalientes

312		aichi
240		akershus
383		akita
60	al	alabama
163		alagoas
2	ak	alaska
63		alberta
433		alsace
164		amapa
165		amazonas
116		andalucia
87		antwerpen
313		aomori
412		aquitaine
119		aragon
3	az	arizona
4	as	arkansas
380		armed forces americas
381		armed forces europe
382		armed forces pacific
113		arquipelago da madeira
114		arquipelago dos acores
247		aust-agder
59		australian capital territory
413		auvergne
95		aveiro
434		bacs-kiskun
135		baden-wuerttemberg
166		bahia
455		baja california
456		baja california sur
287		banksobystricky kraj
435		baranya
190		basilicata
414		basse-normandie
136		bayern
96		beja
436		bekes
137		berlin
359		blekinge
437		borsod-abauj-zemplen
415		bourgogne
91		brabant wallon
97		braga
98		braganca
138		brandenburg
288		bratislavsky kraj
139		bremen
416		bretagne
64		british columbia
93		brussels
438		budapest
258		burgenland

297		busan
244		buskerud
192		calabria
5	ca	california
193		campania
457		campeche
130		cantabria
99		castelo branco
115		castilla y leon
117		castilla-la mancha
121		cataluna
167		ceara
417		centre
133		ceuta
418		champagne-ardenne
486		chatham islands
458		chiapas
314		chiba
459		chihuahua
305		chungcheong buk do
306		chungcheong nam do
460		coahuila
100		coimbra
461		colima
6	co	colorado
127		comunidad de madrid
126		comunidad foral de navarra
123		comunidad valenciana
227		connacht
7	ct	connecticut
419		corse
439		csongrad
298		daegu
301		daejeon
360		dalarna
9	de	delaware
8	dc	district of columbia
168		distrito federal
462		distrito federal
396		dolnoslaskie
385		drenthe
226		dublin
463		durango
215		east midlands
153		eastern cape
219		eastern england
315		ehime
194		emilia-romagna
234		espace mittelland (be/so)
169		espirito santo
101		evora
120		extremadura
102		faro

440		fejer
257		finnmark
395		flevoland
10	fl	florida
420		franche-comte
160		free state
394		friesland
195		friuli-venezia giulia
316		fukui
317		fukuoka
318		fukushima
122		galicia
304		gangwondo
159		gauteng
362		gavleborg
387		gelderland
11	ga	georgia
319		gifu
170		goias
361		gotland
229		graubuenden (gr)
384		groningen
464		guanajuato
103		guarda
465		guerrero
320		gunma
300		gwangju
303		gyeonggido
309		gyeongsang buk do
310		gyeongsang nam do
441		gyor-moson-sopron
88		hainaut
442		hajdu-bihar
363		halland
140		hamburg
421		haute-normandie
12	ha	hawaii
242		hedmark
150		hessen
443		heves
466		hidalgo
321		hiroshima
286		hlavni mesto praha
322		hokkaido
250		hordaland
323		hyogo
324		ibaraki
13	id	idaho
422		ile-de-france
14	il	illinois
299		incheon
15	in	indiana
16	ia	iowa

325		ishikawa
132		islas baleares
128		islas canarias
326		iwate
467		jalisco
364		jamtland
444		jasz-nagykun-szolnok
311		jejudo
307		jeolla buk do
308		jeolla nam do
274		jihocesky kraj
273		jihomoravsky kraj
365		jonkoping
236		jura (ju/ne)
327		kagawa
328		kagoshima
366		kalmar
329		kanagawa
17	ks	kansas
276		karlovarsky kraj
259		karnten
18	ky	kentucky
330		kochi
445		komarom-esztergom
289		kosicky kraj
284		kraj vysocina
275		kralovehradecky kraj
367		kronoberg
397		kujawsko-pomorskie
331		kumamoto
157		kwazulu natal
332		kyoto
131		la rioja
423		languedoc-roussillon
196		lazio
228		leinster
104		leiria
277		liberecky kraj
80		liege
197		liguria
89		limburg
393		limburg
424		limousin
158		limpopo
105		lisboa
400		lodzkie
198		lombardia
220		london
425		lorraine
19	la	louisiana
398		lubelskie
399		lubuskie
90		luxembourg

20	me	maine
401		malopolskie
65		manitoba
171		maranhao
199		marche
21	md	maryland
22	md	massachusetts
172		mato grosso
173		mato grosso do sul
402		mazowieckie
141		mecklenburg-vorpommern
134		melilla
468		mexico
23	mi	michigan
469		michoacan
426		midi-pyrenees
333		mie
174		minas gerais
24	mn	minnesota
25	ms	mississippi
26	mo	missouri
334		miyagi
335		miyazaki
200		molise
27	mt	montana
279		moravskoslezsky kraj
252		more og romsdal
470		morelos
155		mpumalanga
225		munster
336		nagano
337		nagasaki
81		namur
338		nara
471		nayarit
28	ne	nebraska
29	nv	nevada
66		new brunswick
30	nh	new hampshire
31	nj	new jersey
32	nm	new mexico
52		new south wales
33	ny	new york
67		newfoundland and labrador
260		niederosterreich
142		niedersachsen
339		niigata
290		nitriansky kraj
446		nograd
392		noord-brabant
389		noord-holland
255		nordland
427		nord-pas-de-calais

143		nordrhein-westfalen
254		nord-trondelag
232		nordwestschweiz (ag/bl/bs)
368		norrbotnen
34	nc	north carolina
35	nd	north dakota
82		north island
217		north wales
156		north west
212		northeast england
154		northern cape
210		northern scotland
58		northern territory
213		northwest england
72		northwest territories
68		nova scotia
472		nuevo leon
73		nunavut
473		oaxaca
261		oberosterreich
36	oh	ohio
340		oita
341		okayama
342		okinawa
37	ok	oklahoma
278		olomoucky kraj
69		ontario
76		oost-vlaanderen
403		opolskie
243		oppland
378		orebro
38	or	oregon
343		osaka
241		oslo
379		ostergotland
239		ostfold
230		ostschweiz (sg/sh/tg/ai/ar/gl)
386		overijssel
129		pais vasco
175		para
176		paraiba
177		parana
280		pardubicky kraj
428		pays de la loire
39	pa	pennsylvania
178		pernambuco
447		pest
179		piaui
429		picardie
201		piemonte
281		plzensky kraj
404		podkarpackie
405		podlaskie

430		poitou-charentes
406		pomorskie
106		portalegre
107		porto
291		presovsky kraj
70		prince edward island
125		principado de asturias
431		provence-alpes-cote d'azur
474		puebla
202		puglia
62		quebec
54		queensland
475		queretaro
476		quintana roo
124		region de murcia
231		region zuerich (zh)
144		rheinland-pfalz
40	ri	rhode island
432		rhone-alpes
180		rio de janeiro
181		rio grande do norte
182		rio grande do sul
249		rogaland
183		rondonia
184		roraima
145		saarland
146		sachsen
147		sachsen-anhalt
344		saga
345		saitama
262		salzburg
477		san luis potosi
185		santa catarina
108		santarem
186		sao paulo
203		sardegna
71		saskatchewan
148		schleswig-holstein
296		seoul
187		sergipe
109		setubal
346		shiga
347		shimane
348		shizuoka
204		sicilia
478		sinaloa
369		skane
407		slaskie
371		sodermanland
251		sogn og fjordane
448		somogy
479		sonora
253		sor-trondelag

55		south australia
41	sc	south carolina
42	sd	south dakota
223		south east england
86		south island
218		south wales
222		south west england
221		southern england
211		southern scotland
263		steiermark
370		stockholm
282		stredocesky kraj
235		suisse romande (ge/vd/fr)
408		swietokrzyskie
449		szabolcs-szatmar-bereg
480		tabasco
481		tamaulipas
57		tasmania
245		telemark
43	tn	tennessee
238		tessin (ti)
44	tx	texas
149		thuringen
264		tirol
482		tlaxcala
188		tocantins
349		tochigi
350		tokushima
351		tokyo
450		tolna
205		toscana
352		tottori
353		toyama
292		trenciansky kraj
206		trentino-alto adige
293		trnavsky kraj
256		troms
302		ulsan
224		ulster
207		umbria
372		uppsala
283		ustecky kraj
45	ut	utah
388		utrecht
208		valle d'aosta
373		varmland
451		vas
374		vasterbotten
375		vasternorrland
376		vastmanland
377		vastra gotaland
209		veneto
483		veracruz

46	vt	vermont
248		vest-agder
246		vestfold
452		veszprem
110		viana do castelo
53		victoria
112		vila real
47	va	virginia
111		viseu
78		vlaams-brabant
265		vorarlberg
354		wakayama
237		wallis (vs)
409		warminsko-mazurskie
48	wa	washington
216		west midlands
49	wv	west virginia
56		western australia
152		western cape
92		west-vlaanderen
410		wielkopolskie
295		wien
50	wi	wisconsin
51	wy	wyoming
355		yamagata
356		yamaguchi
357		yamanashi
214		yorkshire
484		yucatan
74		yukon territory
485		zacatecas
411		zachodniopomorskie
453		zala
391		zeeland
233		zentralschweiz (zg/sz/lu/ur/ow/nw)
294		zilinsky kraj
285		zlinsky kraj
390		zuid-holland

**EXAMPLES**

```
$ geo-newest germany berlin
$ geo-newest .de berlin
$ geo-newest .deu berlin
$ geo-newest c79 berlin
```

**SEE ALSO**

geo-newest, geo-demand, geo-pqs

**NAME**

**geo-crossword** - Search for words in /usr/share/dict/words

**SYNOPSIS**

**geo-crossword** [*options*] *letters* ...

**DESCRIPTION**

Search for words in /usr/share/dict/words.

If '*letters*' is prefixed by '!' or '^' then complement the search.

**OPTIONS****-w words**

Take words from file

**-D lvl** Debug level

**EXAMPLE**

Search for 4 letter words:

```
$ geo-crossword ade asw . bd
asgd
awed
dabb
dand
darb
dard
daub
daud
```

Search in french:

```
$ geo-crossword -w ~/lib/geo/french aeou mtrcpaeou p ou t aeou mtrcp
amputer
empoter
```

Don't search for 'ilstcg':

```
$ geo-crossword z '!ilstcg' ^ilstcg a ^ilstcg
zakah
zaman
zoeae
zonar
```

**SEE ALSO**

[http://www.a2zwordfinder.com/crossword\\_dictionary.html](http://www.a2zwordfinder.com/crossword_dictionary.html)

<https://packages.ubuntu.com/trusty/wfrench>

<https://packages.ubuntu.com/trusty/wspanish>

**NAME**

**geo-demand** - Perform a Pocket Query

**SYNOPSIS**

```

geo-demand [options]
geo-demand [options] latitude longitude
geo-demand [options] zipcode
geo-demand [options] GCxxxx
geo-demand [options] state [latitude longitude]
geo-demand [options] country
geo-demand -o outfmt ....
geo-demand -k glob-pattern

```

**DESCRIPTION**

Pocket Query with demand by email mode...

```

geo-demand [options]
geo-demand [options] latitude longitude
geo-demand [options] zipcode
geo-demand [options] GCxxxx
geo-demand [options] state
geo-demand [options] country

```

Demand a GPX email of a set of geocaches.

"state" can be al, ak, ..., wy or "allstates"

After the query is entered, this script will start a background process that will wait 20 minutes, and then the query will be deleted. The "-w" option puts that process in the foreground. The "-W" option prevents starting that process at all.

Instant data delivery mode...

```

geo-demand -o outfmt ....

```

Any of the command formats above are allowed, and the -o outfmt option must be specified. In this mode, the data is delivered instantly, just like with geo-nearest, etc.

Delete (kill) PQ's by name

```

geo-demand -k glob-pattern

```

Delete (kill) patterns which match glob-pattern by name.

Requires:

- A subscriber login at <http://www.geocaching.com>.

**- curl**

<http://curl.haxx.se/>

**OPTIONS**

- d N[+-]**  
Difficulty level [1+]
- t N[+-]**  
Terrain level [1+]
- e address**  
Email results to this address [account email address]
- z** Do not unzip the email contents.
- n num** Return "num" caches [500]
- r radius**  
Return caches within radius (mi or km) [100mi]
- w** Wait for query to be removed.
- W** Do not delete query.
- T period**  
Placed within last period (week, month, year)
- T mm/dd/yyyy-mm/dd/yyyy**  
Placed between two dates. Also **-mm/dd/yyyy** (oldest) and **mm/dd/yyyy-** (newest)
- q qualifiers**  
Limit by one or more space/comma separated qualifiers:  
  - Type: these ones OR together....  
 traditional, multi, virtual, letterbox, event,  
 mystery, APE, webcam, earth, gps, wherigo
  - Container: these ones OR together....  
 small, other, none, large, regular, micro, unknown
  - These ones AND together....  
 ifound, notfound, bug, unfound, notowned,  
 new, iown, watchlist, updated, active, notactive,  
 notign, found7, soc, notsoc
- N name/number**  
Set the demand query name or number (1-20) [1]
- a attributes**  
Set attribute values.  
  - [~]scenic, [~]dogs, [~]fee, [~]rappelling, [~]boat,
  - [~]scuba, [~]kids, [~]onehour, [~]climbing,
  - [~]wading, [~]swimming, [~]available, [~]night,
  - [~]winter, [~]cliff, [~]hunting, [~]danger,
  - [~]wheelchair, [~]camping, [~]bicycles,
  - [~]motorcycles, [~]quads, [~]jeeps, [~]snowmobiles,
  - [~]campfires, [~]poisonoak, [~]thorn,
  - [~]dangerousanimals, [~]ticks, [~]mine, [~]parking,
  - [~]public, [~]picnic, [~]horses, [~]water,
  - [~]restrooms, [~]phone, [~]stroller, [~]firstaid,
  - [~]cow, [~]stealth, [~]landf, [~]flashlight, [~]rv,
  - [~]luv, [~]snowshoes, [~]skiis, [~]s-tool,
  - [~]nightcache, [~]parkngrab, [~]abandonedbuilding,
  - [~]hike\_short, [~]hike\_med, [~]hike\_long, [~]fuel,

```
[~]food, [~]wirelessbeacon, [~]partnership,
[~]field_puzzle, [~]hiking, [~]seasonal,
[~]touristok, [~]treeclimbing, [~]frontyard,
[~]teamwork, [~]geotour,
```

~keyword means NOT keyword.

- c** Remove cookie file when done
- u username**  
Username for http://www.geocaching.com
- p password**  
Password for http://www.geocaching.com
- U** Retrieve latest version of this script
- D lvl** Debug level [0]
  - 0: Create and run query, then delete it
  - 1: Create query but do not run or delete it
  - 2: More verbose version of -D1
  - 3: Just show what curl command would be executed

Instant Data Options:

- o format**  
Output format, **-o?** for possibilities [] plus "gpsdrive.sql" for direct insertion into MySQL DB plus "map[,geo-map-opts]" to display a geo-map.
- O filename**  
Output file, if not stdout
- H htmldir**  
Also fetch the printable HTML pages (slowly)
- L logdir**  
Also fetch the plain text log entries (slowly)
- f** Do not report any found or unavailable caches
- F** Report caches found by the login 'username' as unfound

Defaults can also be set with variables in file \$HOME/.georc:

```
PASSWORD=password;  USERNAME=username;
LAT=latitude;      LON=logitude;
```

## EXAMPLES

Nearest 500 caches to my home location:

```
geo-demand
```

Nearest 500 caches to a lat/lon:

```
geo-demand 44.53 -93.56
geo-demand 44.25.234 -93.51.543
```

Nearest 500 caches to a zip code:

```
geo-demand 55344
```

500 caches in a *state*:

```
geo-demand mn
```

500 caches in a *state* using lat/lon:

```
geo-demand mn n43.6 w92
```

500 caches in a *country*:

```
geo-demand iraq
```

500 caches in a *country* by code:

```
geo-demand c12
```

500 caches in a foreign *state*:

```
geo-demand berlin
```

Caches I have not found, and wait until query is deleted before exiting (useful in batch scripts):

```
geo-demand -q notfound -w
```

Generate a query, but do not execute it. Check the gc.com website to see what query would have been run...

```
geo-demand -D1
```

Check website: <https://www.geocaching.com/pocket/>

Append to the ignore list any caches that were ever SOCs:

```
ignore=$HOME/.geo-ignore
geo-demand -o gpsdrive -qsoc mn |
  awk '{print $1}' >> $ignore
  sort -u -o $ignore $ignore
```

Delete patterns which match "mn-":

```
geo_demand -k mn-
```

## SEE ALSO

geo-countries-states geo-newest, geo-found, geo-placed, geo-nearest, <http://geo.rkkda.com/>

**NAME**

**geo-density** - Compute the cache density of a circular area

**SYNOPSIS**

```
geo-density [options]  
geo-density [options] latitude longitude  
geo-density [options] zipcode
```

**DESCRIPTION**

Compute the cache density of a circular area.

**OPTIONS**

**-c** Remove cookie file when done

**-q** Qualifier: unknown

**-r radius**

Radius in miles for computing the density [4]

**-D lvl** Debug level [0]

**-U** Retrieve latest version of this script

Defaults can also be set with variables in file /home/rick/.georc:

```
LAT=latitude;      LON=logitude;
```

**SEE ALSO**

<http://geo.rkkda.com/>

**NAME**

**geo-dist** - compute total distance between a set of waypoints

**SYNOPSIS**

**geo-dist** [*options*] *latitude longitude* [*label* [*symbol*]] ...

**DESCRIPTION**

Compute total distance and bearing between a set of waypoints. Acceptable formats for lat/lon are:

```
-93.49130      DegDec (decimal degrees)
W93.49130      DegDec (decimal degrees)
"-93 29.478"   MinDec (decimal minutes)
"W93 29.478"   MinDec (decimal minutes)
-93.29.478     MinDec (decimal minutes)
W93.29.478     MinDec (decimal minutes)
W 93° 29.478   Cut/paste from gc.com (note it is 3 arguments)
"-93 45 30"    DMS (degrees, minutes, seconds)
```

"*label*" and "*symbol*" are optional, can be any text, and are ignored. They are accepted for compatibility with the command line input format of geo-map.

If a lat/lon of 0/0 appears in the list, it is ignored and a new route is started.

**OPTIONS****-t waypoints**

A file of waypoints to plot in tabsep, GPX, or in extended Tiger format: LONG,LAT:SYMBOL:LABEL:URL

**-i** Incremental

**-g** true/false iff dist <= 2mi

**-v** Use Vincenty instead of 'rough'

**-D lvl** Debug level [0]

**-U** Retrieve latest version of this script

**EXAMPLES**

Two waypoints:

```
$ geo-dist N44.48.938 W093.31.988 N44.49.245 W093.30.507
1.258898mi      2.026km 2026m   6647ft   74.13
```

Two waypoints, Vincenty formula:

```
$ geo-dist -v N44.48.938 W093.31.988 N44.49.245 W093.30.507
1.2632476mi    2.033km 2033m   6670ft   73.75
```

Route in a GPX file:

```
$ geo-dist -t bikeathon/bikewalk.gpx
2.8129474mi    4.527km 4527m   14852ft 175.26
```

Four waypoints:

```
$ geo-dist -i 45 w93 44.59.809 -93.0.269 \
          45.0.184 -93.0.269 45.0.375 -93.00.000

1      0.31006422mi    0.499km 499m   1637ft 225.00
2      0.43123161mi    0.694km 694m   2277ft 0.00
```

geo-dist(1)

geo-dist(1)

3	0.31006422mi	0.499km	499m	1637ft	44.92
TOTAL	1.0513601mi	1.692km	1692m	5551ft	0.00

**SEE ALSO**

geo-code, geo-nearest, geo-pg, geo-waypoint, <http://geo.rkkda.com/>



**NAME**

**geo-fax** - Decode a FAX using 0s and 1s

**SYNOPSIS**

**geo-fax** [*options*]

**DESCRIPTION**

Decode a FAX using 0s and 1s. EOL's (0000000001) are optional and supported.

Only run lengths up to 32 are supported at this time!

**OPTIONS**

**-D lvl** Debug level

**EXAMPLE**

Decode <https://coord.info/GC4XGD1>:

```

$ geo-fax
000111100001111011111010001000011101000011101000011110
100001010000101010001010000101000010000111010000111010000111010
000111100001111011111010001000011110000111010000111010
10000100001110100100001011000101000010000111010000111010
000111100001111011111000011101000011110100001000011110
### ###          ###   ### # # ###
#  #             #   #  # # # #
### ###          ###   ### ### # #
#  #             #   #  # # #
### ###          ### # ###  # ###

```

**SEE ALSO**

<https://www.windsurfnow.co.uk/imitedit/ModHuffman.html>  
/10\_4%20Modified%20Huffman%20Coding.htm

<http://www.iet.unipi.it/m.luise/HTML/SdT>

**NAME**

**geo-firefox** - Display a map of a point using aerial photos

**SYNOPSIS**

**geo-firefox** [*options*] *lat lon*

**DESCRIPTION**

Display a map of a point using Bing, Google, AOL, or MapQuest aerial photos and Firefox.

**OPTIONS**

- b**      Batch mode on stdin.
- a source**  
        source: mapquest, bing, google, aol [google]
- n number**  
        Google maps number [5]
- s**      Google Streetview
- z zoom**  
        Zoom level (max, 1-19) [max]
- D lvl**    Debug level

**EXAMPLES**

```
$ geo-firefox 45.04.337 w93.45.414 #A
$ geo-firefox -z 13 45.03.274 w93.38.288 #B
$ geo-firefox 45.00.601 w93.21.109 #C
$ geo-firefox 44.59.668 w93.15.301 #D
$ geo-firefox 45.035778 w93.512187
$ geo-firefox -s N21.27.588 W157.49.934
$ geo-firefox -b
```

**SEE ALSO**

geo-map, <http://geo.rkkda.com/>

**NAME**

**geo-found** - Fetch a list of geocaches found by a specific user

**SYNOPSIS**

**geo-found** [*options*] [*username*]

**geo-found** [*options*] [*username*] [*lat*] [*lon*]

**DESCRIPTION**

Fetch a list of geocaches found by a specific user. Only unique caches are found (i.e. two or more logs on a cache are listed only once). Archived caches have the *lat/lon* set to 0.0, 0.0.

Requires: A premium member (\$30/yr) OR a basic member (free) login at: <http://www.geocaching.com>  
Visit a cache page and click the "Download to EasyGPS" link at least once so you can read and agree to the license terms. Otherwise, you will not get any waypoint data.

**curl** <http://curl.haxx.se/>

**gpsbabel**  
<http://gpsbabel.sourceforge.net/>

**OPTIONS****-b bookmark**

Use list "bookmark" [none]

**-q query**

geo-nearest: Use PQ list "query" [none]

**-q search**

geo-newest: Use search "search" [none]

**-c** Remove cookie file when done

**-f** Do not report any found or unavailable caches

**-m** Do not report any members-only caches

**-F** Report caches found by the login '*username*' as unfound

**-n num** Return "num" caches [20]

**-s** Output short names for the caches (gpsbabel option)

**-I term** Include only caches with 'term' [\*]

**-X term**

Exclude caches with 'term' [*\_NoThInG\_*] terms: ~ (exclude none), unfound, ifound, soc, unavail, regular, multi, virtual, webcam, event, hybrid, cito

**-r radius**

Display only caches with radius in miles (e.g. **-r 25**) Suffix the value with "km" for kilometers.

**-M mystery**

Use file 'mystery' for unknown/mystery/puzzle caches [*/home/rick/.geo-mystery*]. Awk Format:

*gcid lat lon comment*

i.e.: GC2CBVB n44.45.123 w93.00.321 Final

**-u username**

Username for <http://www.geocaching.com>

**-p password**

Password for <http://www.geocaching.com>

- o format**  
Output format, **-o?** for possibilities [gpsdrive] plus "gpsdrive.sql" for direct insertion into MySQL DB plus "map[,geo-map-opts]" to display a geo-map.
- O filename**  
Output file, if not stdout
- S** Alias for **-o** gpsdrive.sql
- d** For **-S**, just delete selected records
- P** For **-S**, purge all records of type **-t** Geocache\*
- t type** For **-ogpsdrive.sql**, the waypoint type [Geocache]
- H htmldir**  
Also fetch the printable HTML pages (slowly)
- L logdir**  
Also fetch the plain text log entries (slowly) For **-H** or **-L**, the limit is 1500 updated caches/day.
- ! "lpr -Plp"**  
Print HTML pages
- E var=val**  
Set environment "var" to "val" i.e. DATEFMT=0|1
- D lvl** Debug level [0]
- U** Retrieve latest version of this script

## DEFAULTS

Defaults can also be set with variables in file \$HOME/.georc:

```
PASSWORD=password;   USERNAME=username;   SOC=0|1;
LAT=latitude;        LON=logitude;        GEOMYSTERY=/dev/null;
NUM=num;              UTFMT=format;         BABELFLAGS=-s;
SQLUSER=gast;         SQLPASS=gast;         SQLDB=geoinfo;
DATEFMT=[0|1];       CACHE_CACHE_MAX_AGE="1 day";
```

## DATE FORMATS

Geocaching.com date formats that are compatible:

GC Format	Example	Compatible
YYYY-MM-DD	2011-07-13	yes
YYYY/MM/DD	2011/07/13	yes
MM/DD/YYYY	07/13/2011	yes
DD/MM/YYYY	13/07/2011	yes if DATEFMT=1 in \$HOME/.georc
DD.MM.YYYY	13.07.2011	yes if DATEFMT=1 in \$HOME/.georc
DD/Mmm/YYYY	13/Jul/2001	no
DD.Mmm.YYYY	13.Jul.2001	no
Mmm/DD/YYYY	Jul/13/2011	no
DD Mmm YY	13 Jul 11	yes (english only)

Change them here:

<http://www.geocaching.com/account/ManagePreferences.aspx>

## NOTE

A basic member will get caches very slow (20 cache pages per minute) because we have to get the actual cache pages. They will be stored in: ~/.geo/caches/GCXXXX.html. Of course, after running this command, geo-html2gpx could be run.

**EXAMPLES**

Show the most recent 50 caches found by Jeremy:

```
geo-found -s -n50 Jeremy
```

Show the most recent caches found by Jeremy that are with a radius of 15 miles of your home location:

```
geo-found -s -r15M Jeremy
```

Show the most recent caches found by Jeremy that are with a radius of 15 miles of a specific location:

```
geo-found -s -r50 Jeremy N47.20.000 W121.30.000
```

Make a FULL backup of all of my cache logs (can take awhile):

```
geo-found -n9999 -L ifound -otabsep > ifound.tabsep
```

Append an incremental backup of all of my cache logs:

```
DIR=ifound; FILE=$DIR.tabsep
```

```
geo-found -n40 -L $DIR -otabsep >> $FILE
```

```
gpsbabel -itabsep -f$FILE -xduplicate,shortname -otabsep -F$FILE
```

**FILES**

~/georc ~/.geo/caches/

**SEE ALSO**

geo-nearest, geo-newest, geo-keyword, geo-placed, geo-code, geo-waypoint, <http://geo.rkkda.com/>

**NAME**

**geo-gccode2id** - Convert GC codes to the decimal equivalent

**SYNOPSIS**

**geo-gccode2id** [*options*] *GC-code* ...

**DESCRIPTION**

Convert GC codes to the decimal equivalent.

**OPTIONS**

**-g** Print out the gallery for unpublished caches

**-D lvl** Debug level

**EXAMPLE**

Convert codes:

```
$ geo-gccode2id gcc8 gcff gcg000 gczzzz gc10000 GC588H3 GCAG9X3
GCC8      200
GCFE      255
GCG000    65536
GCZZZZ    512400
GC10000   512401
GC588H3   4453031
GCAG9X3   9310266
```

Print out the gallery for an unpublished cache:

```
$ geo-gccode2id -g GC54BET
GC54BET 4336679 http://www.geocaching.com/seek/gallery.aspx?id=4336679
```

**SEE ALSO**

geo-id2gccode

**NAME**

**geo-gid** - Fetch data about geocaches by gc.com GID

**SYNOPSIS**

*geo-gid* [*options*] *gid* ...

**DESCRIPTION**

Fetch data about geocaches by gc.com GID. Only works with caches that are active (not archived).

Requires: A subscriber (0/yr) login at <http://www.geocaching.com>. Visit a cache page and click the "Download to EasyGPS" link at least once so you can read and agree to the license terms. Otherwise, you will not get any waypoint data.

**curl** <http://curl.haxx.se/>

**gpsbabel**  
<http://gpsbabel.sourceforge.net/>

**OPTIONS**

- c** Remove cookie file when done
- f** Do not report any found or unavailable caches
- m** Do not report any members-only caches
- F** Report caches found by the login 'username' as unfound
- s** Output short names for the caches (gpsbabel option)
- u username**  
Username for <http://www.geocaching.com>
- p password**  
Password for <http://www.geocaching.com>
- o format**  
Output format, **-o?** for possibilities [gpsdrive] plus "gpsdrive.sql" for direct insertion into MySQL DB plus "map[,geo-map-opts]" to display a geo-map.
- O filename**  
Output file, if not stdout
- S** Alias for **-o** gpsdrive.sql
- d** For **-S**, just delete selected records
- P** For **-S**, purge all records of type **-t** Geocache\*
- t type** For **-ogpsdrive.sql**, the waypoint type [Geocache]
- H htmldir**  
Also fetch the printable HTML pages (slowly)
- L logdir**  
Also fetch the plain text log entries (slowly) For **-H** or **-L**, the limit is 1500 updated caches/day.
- ! "lpr -Plp"**  
Print HTML pages
- D lvl** Debug level [0]
- U** Retrieve latest version of this script

**DEFAULTS**

Defaults can also be set with variables in file `/home/rick/.georc:`

```
PASSWORD=password;  USERNAME=username;  SOC=0 | 1;
```

geo-gid(1)

geo-gid(1)

```
LAT=latitude;      LON=logitude;      GEOMYSTERY=/dev/null;
NUM=num;           OUTFMT=format;     BABELFLAGS=-s;
SQLUSER=gast;      SQLPASS=gast;      SQLDB=geoinfo;
```

**EXAMPLES**

*geo-gid* GCG000

**SEE ALSO**

geo-newest, geo-found, geo-placed, geo-nearest, <http://geo.rkkda.com/>

**NAME**

**geo-gpx** - Fetch GPX **file(s)** by gc.com waypoint name

**SYNOPSIS**

**geo-gpx** [*options*] *gid* ...

**DESCRIPTION**

Fetch GPX **file(s)** by gc.com waypoint name (i.e. GCxxxx)

If no output format is specified, the GPX data is stored into individual files named *<gid>.gpx*.

If an output format is specified with **-o**, the GPX data is combined into a single file with that format and output into stdout or to the filename specified with the **-O** option.

Requires: A subscriber login at <http://www.geocaching.com>.

**OPTIONS****-o format**

Output format, **-o?** for possibilities [] plus "gpsdrive.sql" for direct insertion into MySQL DB plus "map[,geo-map-opts]" to display a geo-map.

**-O filename**

Output file, if not stdout

**-u username**

Username for <http://www.geocaching.com>

**-p password**

Password for <http://www.geocaching.com>

**-D lvl** Debug level [0]**-U** Retrieve latest version of this script

Defaults can also be set with variables in file \$HOME/.georc:

```
PASSWORD=password;  USERNAME=username;
```

**EXAMPLES**

Get a gc.com style gpx file for a single cache...

```
geo-gpx GC3T7TK
```

Get a gc.com style gpx file for the 20 newest caches...

```
geo-gpx -ogpx -Onewest.gpx $(geo-newest | awk '{print $1}')
```

**SEE ALSO**

*geo-gid*, *geo-newest*, *geo-found*, *geo-placed*, *geo-nearest*, <http://geo.rkkda.com/>

**NAME**

**geo-gpxmail** - Process PQ email using gpx2html

**SYNOPSIS**

**geo-gpxmail** [*options*]

**DESCRIPTION**

Process PQ email using gpx2html. The results are placed under directory '\$PQDIR/<pqname>'. \$PQDIR can be set on the command line with the **-d** option, or in /home/rick/georc. The default is PQDIR=/home/rick/Caches.

<pqname> is determined from the subject line of the PQ email. Which means this script could break at any time due to the whims of Jeremy.com.

If a shell script named '\$PQDIR/<pqname>/preconvert.sh' exists, it will be executed before gpx2html is run. This can be used, for example, to copy other GPX files into the current directory for merging. E.G.:

```
#/bin/sh
cp ../found/found.gpx .
```

If a shell script named '\$PQDIR/<pqname>/postconvert.sh' exists, it will be executed after gpx2html is run. This can be used for example, to convert the gpx files to other formats.

The shell variables \$PQDIR and \$PQNAME are available to the scripts for their internal use.

Here is a typical /home/rick/.procmailrc recipe to use this program:

```
#
#       Automagically unpack geocaching locations
#
:0
* ^Subject:.*GEO] Pocket Query:
| geo-gpxmail -k
```

Requires: A subscriber login at <http://www.geocaching.com>.

**OPTIONS**

- i** Incremental (gpx2html mn-20.gpx)
- k** Kill all gpx2html processes
- d pqdir** Base directory for all PQ's [/home/rick/Caches/]
- D lvl** Debug level

**EXAMPLES**

Request and process the list of caches I have found and place it into directory /home/rick/Caches/found/

```
$ geo-demand -Nfound -qifound
```

Request and process 500 nearest caches I have not found and place it into directory /home/rick/Caches/DemandQuary1/

```
$ geo-demand -qnotfound
```

**SEE ALSO**

geo-demand

**NAME**

**geo-gpxprocess** - Process PQ **download**(s) using geo-pqdownload and gpx2html

**SYNOPSIS**

**geo-gpxprocess** [*options*] **-n** *NAMES*

**DESCRIPTION**

Process PQ **download**(s) using geo-pqdownload and gpx2html. This is good for PQs with 501-1000 caches, since they won't be emailed to you.

The results are placed under directory '\$PQDIR/<pqname>'. \$PQDIR can be set on the command line with the **-d** option, or in /home/rick/.georc. The default is PQDIR=/home/rick/Caches.

*NAMES* is composed of a string of alphabetic letters (a-zA-Z) followed by anything. The BASEPQNAME is the prefix. I.E. *NAMES*="mn-\*" and BASEPQNAME="mn". The default is "mn-\*".

If a shell script named '\$PQDIR/<pqname>/preconvert.sh' exists, it will be executed before gpx2html is run. This can be used, for example, to copy other GPX files into the current directory for merging. E.G.:

```
#/bin/sh
cp ../found/found.gpx .
```

If a shell script named '\$PQDIR/<pqname>/postconvert.sh' exists, it will be executed after gpx2html is run. This can be used for example, to convert the gpx files to other formats.

The shell variables \$PQDIR and \$PQNAME are available to the scripts for their internal use.

Requires: A subscriber login at <http://www.geocaching.com>.

**OPTIONS**

- i** Incremental (gpx2html mn-20.gpx)
- k** Kill all gpx2html processes
- n** *NAMES*  
Download and process just *NAMES* (mn-\*)
- d** *pqdir*  
Base directory for all PQ's [/home/rick/Caches/]
- D** *lvl* Debug level

**EXAMPLES**

Request and process the list of caches from 11/05/2000 to 04/02/2004.

```
$ geo-demand -n 1000 -N mn-00 -T 11/05/2000-04/02/2004
$ sleep 1800
$ geo-gpxprocess -n "mn-*"
```

Crontab for rick:

```
$ crontab -l
18 1 1,14 * * geo-demand -n1000 -qnotsoc -T'11/05/2000-07/06/2005' -Nmn-00 "mn"
18 1 2,15 * * geo-demand -n1000 -qnotsoc -T'07/07/2005-08/24/2006' -Nmn-01 "mn"
...
18 1 12,25 * * geo-demand -n1000 -qnotsoc -T'09/25/2009-12/04/2009' -Nmn-11 "mn"
28 1 * * * geo-demand -n1000 -qnotsoc -T'12/05/2009-03/22/2010' -Nmn-12 "mn"
38 1 * * * geo-demand -n1000 -qnotsoc -T'03/23/2010-05/01/2010' -Nmn-13 "mn"
48 1 * * * geo-demand -n1000 -qnotsoc -T'05/02/2010-' -Nmn-14 "mn"
30 4 * * * geo-gpxprocess -i -n "mn-*"
```

geo-gpxprocess(1)

geo-gpxprocess(1)

**SEE ALSO**

geo-pqdownload, gpx2html

**NAME**

**geohash2ll** - Decode a geohash into lat/lon

**SYNOPSIS**

**geohash2ll** [*options*] *geohash* ...

**DESCRIPTION**

Decode a *geohash* into lat/lon.

**OPTIONS**

**-D lvl** Debug level

**EXAMPLES**

Decode:

```
$ geohash2ll dnr7r3h1c254 u4pruydqqvj
dnr7r3h1c254      35.733333 -79.480533   n35.44.000 w79.28.832
u4pruydqqvj      57.649111 10.407440   n57.38.947 e10.24.446
```

**SEE ALSO**

<http://en.wikipedia.org/wiki/Geohash>

**NAME**

**geo-html2gpx**- Convert gc.com *\*printable\** web pages into GPX

**SYNOPSIS**

**geo-html2gpx** [*options*] [gc-com.html]...

**DESCRIPTION**

Convert gc.com *\*printable\** web pages into GPX, including cache description and all logs.

The *\*printable\** web pages can be fetched using geo-nearest, geo-newest, geo-placed, geo-found, or geo-gid with the **-H** option.

**OPTIONS**

- b** Normalize output by postprocessing with gpsbabel
- e** Encode hints with rot13 (e.g. NORTH = ABEGU)
- i** Incremental, no XML and GPX headers
- I directory**  
Fetch image and spoiler pictures to directory
- l number**  
Maximum number of log entries to be exported [unlimited]
- n** No HTML in descriptions (experimental)
- o FMT**  
Output FMT instead of GPX by using gpsbabel
- u username**  
Indicate found status for username [rickrich]
- w** Do not add "Additional Waypoints" to the GPX output
- z** Do not output waypoints with "zero" coordinates
- E var=val**  
Set environment "var" to "val" i.e. DATEFMT=0|1
- D lvl** Debug level

**DEFAULTS**

Defaults can also be set with variables in file \$HOME/.georc:

```
DATEFMT=[ 0 | 1 ] ;
```

**DATE FORMATS**

Geocaching.com date formats that are compatible:

GC Format	Example	Compatible
YYYY-MM-DD	2011-07-13	yes
YYYY/MM/DD	2011/07/13	yes
MM/DD/YYYY	07/13/2011	yes
DD/MM/YYYY	13/07/2011	yes if DATEFMT=1 in \$HOME/.georc
DD.MM.YYYY	13.07.2011	yes if DATEFMT=1 in \$HOME/.georc
DD/Mmm/YYYY	13/Jul/2001	no
DD.Mmm.YYYY	13.Jul.2001	no
Mmm/DD/YYYY	Jul/13/2011	no
DD Mmm YY	13 Jul 11	yes (english only)

Change them here:

<http://www.geocaching.com/account/ManagePreferences.aspx>

## **EXAMPLES**

Convert into GPX:

```
geo-found -n9999 -H. > /dev/null
geo-html2gpx *.html > found.gpx
```

Convert GC1NPYG into GPX:

```
geo-gid -H. GC1NPYG
geo-html2gpx GC1NPYG.html > GC1NPYG.gpx
```

**NAME**

**geo-htmltbl2db** - Convert HTML tables into text

**SYNOPSIS**

**geo-htmltbl2db** [*options*] [*html-file*]

**OPTIONS****-F OFS**

Output field separator string [space].

**-t nth** Process nth table only

**-v FMT1=str**

Sprintf style format for field1. Use FMT2...FMT16 for other fields. A "\*" in the format, such as "%\*s", means use the width of the column in the first row to replace the "\*". "%\*.\*s" and "%-\*.s" also work.

**-v FMT=str**

Default format for all columns [%s].

**-v FCOL=num**

First column to process [1]

**-v LCOL=num**

Last column to process [max]

**-v FROW=num**

First row to process [1]

**-v LROW=num**

Last row to process [max]

**-v FTBL=num**

First table to process [1]

**-v LTBL=num**

Last table to process [max]

**-v TSEP=str**

Separate multiple tables with "str" []

**-h bool** Output table header (<th>) lines [1]

**-s search**

Process after /search/ string []

**-D level**

Set debugging level [0]

**NAME**

**geo-id2gcode** - Convert decimal IDs to GC codes

**SYNOPSIS**

**geo-id2gcode** [*options*] *ID* ...

**DESCRIPTION**

Convert decimal IDs to GC codes

**OPTIONS**

**-D lvl** Debug level

**EXAMPLE**

Convert IDs:

```
$ geo-id2gcode 20 255 65536 512400 512401 4453031 9310266
20      GC14      http://coord.info/GC14
255     GCFF      http://coord.info/GCFF
65536   GCG000    http://coord.info/GCG000
512400  GCZZZZ    http://coord.info/GCZZZZ
512401  GC10000   http://coord.info/GC10000
4453031 GC588H3    http://coord.info/GC588H3
9310266 GCAG9X3    http://coord.info/GCAG9X3
```

Fictitious IDs:

```
$ geo-id2gcode 4467676856 146520045
4467676856      GC5123456      http://coord.info/GC5123456
146520045       GC543210       http://coord.info/GC543210
```

**SEE ALSO**

geo-gcode2id

**NAME**

**geo-incomplete-coords** - Print out incomplete coordinates

**SYNOPSIS**

**geo-incomplete-coords** [*options*]

**DESCRIPTION**

Print out incomplete coordinates. Variables can be:

abcd fghijklm opqr tuv xyz

Up to three variables can be specified.

**OPTIONS**

**-u** Unique coordinates. "**-u -u**" is really unique.

**-v let=vals**

Restrict the set of 'vals' for 'let' (e.g. **-v a=13579**)

**-D lvl** Debug level

**EXAMPLES**

a = 0-9:

```
$ geo-incomplete-coords n45.00.a12 w93.25.912
n45.00.012 w93.25.912
n45.00.112 w93.25.912
n45.00.212 w93.25.912
n45.00.312 w93.25.912
n45.00.412 w93.25.912
n45.00.512 w93.25.912
n45.00.612 w93.25.912
n45.00.712 w93.25.912
n45.00.812 w93.25.912
n45.00.912 w93.25.912
```

Use in geo-map:

```
$ geo-map -s0 $(geo-incomplete-coords n45.00.a12 w93.25.b12)
```

Restrict values:

```
$ geo-incomplete-coords -v a=12 -v b=45 -v c=78 n45.00.abc w93.25.abc
n45.00.147 w93.25.147
n45.00.148 w93.25.148
n45.00.157 w93.25.157
n45.00.158 w93.25.158
n45.00.247 w93.25.247
n45.00.248 w93.25.248
n45.00.257 w93.25.257
n45.00.258 w93.25.258
```

Unique coords:

```
$ geo-incomplete-coords -u -v a=05689 -v b=05689 n41.32.a1a w73.47.b26
n41.32.010 w73.47.526
n41.32.010 w73.47.626
n41.32.010 w73.47.826
```

```
n41.32.010 w73.47.926
n41.32.515 w73.47.026
n41.32.515 w73.47.626
n41.32.515 w73.47.826
n41.32.515 w73.47.926
n41.32.616 w73.47.026
n41.32.616 w73.47.526
n41.32.616 w73.47.826
n41.32.616 w73.47.926
n41.32.818 w73.47.026
n41.32.818 w73.47.526
n41.32.818 w73.47.626
n41.32.818 w73.47.926
n41.32.919 w73.47.026
n41.32.919 w73.47.526
n41.32.919 w73.47.626
n41.32.919 w73.47.826
```

Really unique coords:

```
$ geo-incomplete-coords -u -u n41.32.00a w73.47.00b
n41.32.005 w73.47.006
n41.32.005 w73.47.008
n41.32.005 w73.47.009
n41.32.006 w73.47.005
n41.32.006 w73.47.008
n41.32.006 w73.47.009
n41.32.008 w73.47.005
n41.32.008 w73.47.006
n41.32.008 w73.47.009
n41.32.009 w73.47.005
n41.32.009 w73.47.006
n41.32.009 w73.47.008
```

**NAME**

**geo-intersect** - Compute the intersection of two lines

**SYNOPSIS**

**geo-intersect** [*options*] *point1 point2 point3 point4*

**geo-intersect** [*options*] *point1 bearing1 point2 bearing2*

**DESCRIPTION**

Compute the intersection of two lines. You can use two forms. Line segment *point1-point2* and line segment *point3-point4* computes it by "X marks the spot". Line segment *point1 bearing1* and line segment *point2 bearing2* computes it by heading.

**OPTIONS**

- b** Bearing
- p** Planar. Disregard curvature of the surface of the earth.
- D lvl** Debug level

**EXAMPLE**

Compute the intersection by lines:

```
$ geo-intersect \  
    45.04.337 w93.45.414 45.03.274 w93.38.288 \  
    45.00.601 w93.21.109 44.59.668 w93.15.301  
N45.02.199 W93.31.139
```

Compute the intersection by bearings:

```
$ geo-intersect 45 -93 315 45 -94 45  
n45.21.018 w93.30.000
```

```
$ geo-intersect N 45 27.671 W 75 37.390 232 N 45 27.915 W 75 38.192 134  
n45.27.537 w75.37.634
```

**NAME**

**geo-keyword** - Fetch geocaches with **keyword(s)**

**SYNOPSIS**

*geo-keyword* [*options*] *keyword* ...

**DESCRIPTION**

Fetch geocaches with *keyword(s)*.

Requires: A premium member (\$30/yr) OR a basic member (free) login at: <http://www.geocaching.com>  
Visit a cache page and click the "Download to EasyGPS" link at least once so you can read and agree to the license terms. Otherwise, you will not get any waypoint data.

**curl** <http://curl.haxx.se/>

**gpsbabel**  
<http://gpsbabel.sourceforge.net/>

**OPTIONS****-b bookmark**

Use list "bookmark" [none]

**-q query**

geo-nearest: Use PQ list "query" [none]

**-q search**

geo-newest: Use search "search" [none]

**-c** Remove cookie file when done

**-f** Do not report any found or unavailable caches

**-m** Do not report any members-only caches

**-F** Report caches found by the login 'username' as unfound

**-n num** Return "num" caches [20]

**-s** Output short names for the caches (gpsbabel option)

**-I term** Include only caches with 'term' [\*]

**-X term**

Exclude caches with 'term' [**-unavail**] terms: ~ (exclude none), unfound, ifound, soc, unavail, regular, multi, virtual, webcam, event, hybrid, cito

**-r radius**

Display only caches with radius in miles (e.g. **-r 25**) Suffix the value with "km" for kilometers.

**-M mystery**

Use file 'mystery' for unknown/mystery/puzzle caches [/home/rick/.geo-mystery]. Awk Format:

gcid lat lon comment

i.e.: GC2CBVB n44.45.123 w93.00.321 Final

**-u username**

Username for <http://www.geocaching.com>

**-p password**

Password for <http://www.geocaching.com>

- o format**  
Output format, **-o?** for possibilities [gpsdrive] plus "gpsdrive.sql" for direct insertion into MySQL DB plus "map[,geo-map-opts]" to display a geo-map.
- O filename**  
Output file, if not stdout
- S** Alias for **-o** gpsdrive.sql
- d** For **-S**, just delete selected records
- P** For **-S**, purge all records of type **-t** Geocache\*
- t type** For **-ogpsdrive.sql**, the waypoint type [Geocache]
- H htmdir**  
Also fetch the printable HTML pages (slowly)
- L logdir**  
Also fetch the plain text log entries (slowly) For **-H** or **-L**, the limit is 1500 updated caches/day.
- ! "lpr -Plp"**  
Print HTML pages
- E var=val**  
Set environment "var" to "val" i.e. DATEFMT=0|1
- D lvl** Debug level [0]
- U** Retrieve latest version of this script

## DEFAULTS

Defaults can also be set with variables in file \$HOME/.georc:

```
PASSWORD=password;  USERNAME=username;  SOC=0|1;
LAT=latitude;      LON=logitude;      GEOMYSTERY=/dev/null;
NUM=num;           UTFMT=format;      BABELFLAGS=-s;
SQLUSER=gast;     SQLPASS=gast;      SQLDB=geoinfo;
DATEFMT=[0|1];   CACHE_CACHE_MAX_AGE="1 day";
```

## DATE FORMATS

Geocaching.com date formats that are compatible:

GC Format	Example	Compatible
YYYY-MM-DD	2011-07-13	yes
YYYY/MM/DD	2011/07/13	yes
MM/DD/YYYY	07/13/2011	yes
DD/MM/YYYY	13/07/2011	yes if DATEFMT=1 in \$HOME/.georc
DD.MM.YYYY	13.07.2011	yes if DATEFMT=1 in \$HOME/.georc
DD/Mmm/YYYY	13/Jul/2001	no
DD.Mmm.YYYY	13.Jul.2001	no
Mmm/DD/YYYY	Jul/13/2011	no
DD Mmm YY	13 Jul 11	yes (english only)

Change them here:

<http://www.geocaching.com/account/ManagePreferences.aspx>

## NOTE

A basic member will get caches very slow (20 cache pages per minute) because we have to get the actual cache pages. They will be stored in: ~/.geo/caches/GCXXXX.html. Of course, after running this command, geo-html2gpx could be run.

geo-keyword(1)

geo-keyword(1)

**EXAMPLES**

*geo-keyword* Big Stone Lake

**FILES**

~/georc ~/geo/caches/

**SEE ALSO**

geo-nearest, geo-newest, geo-found, geo-placed, geo-code, geo-map, geo-waypoint, <http://geo.rkkda.com/>

**NAME**

**geo-lewis-and-clark** - Encode/decode Lewis and Clark cipher

**SYNOPSIS**

**geo-lewis-and-clark** [*options*]

**DESCRIPTION**

Encode/decode Lewis and Clark cipher, a.k.a. Jefferson Wheel cipher.

The case is significant. Uppercase translates to a letter (A..Z), lowercase yields a number (0..9).

**OPTIONS**

- d** Decode
- e** Encode.
- p passwd**  
Password. (Artichokes)
- D lvl** Debug level

**EXAMPLES**

Encode then decode:

```
$ echo "Hi! How are you?" | geo-lewis-and-clark -e  
I&! AXZ IFP CGV?
```

```
$ echo "I&! AXZ IFP CGV?" | geo-lewis-and-clark  
HI! HOW ARE YOU?
```

Numbers and letters:

```
$ echo jtz JQL pmi | geo-lewis-and-clark  
926 AND 124
```

**SEE ALSO**

<http://lewisandclarktrail.com/legacy/secretcode.htm>

**NAME**

**geo-loran-c** - Brute force solve of Loran-C problems

**SYNOPSIS**

**geo-loran-c** [*options*] *lata lona latb lonb latc lonc NSa NSb NSc*

**DESCRIPTION**

Brute force solve of Loran-C problems

**OPTIONS****-x delta**

Delta X [1000]

**-y delta**

Delta Y [1000]

**-D lvl** Debug level

**EXAMPLE**

```
# http://coord.info/GC34FYA
```

```
$ geo-loran-c n44.52.351 w93.30.996 \  
    n44.52.300 w93.30.930 \  
    n44.52.413 w93.30.915 \  
    -119.970997 -52.8685748 +172.839572  
44.872560 -93.515775  
44.52.354 -93.30.947
```

**NAME**

**geo-map** - Create and display a map centered about a lat/lon

**SYNOPSIS**

**geo-map** [*options*] *latitude longitude* [*label* [*symbol*]] ...

**DESCRIPTION**

Create and display a map centered about a *latitude/longitude*. Lat/Lon may be in DegDec, MinDec, or DMS formats.

I believe that fair use allows you to use the mapblast and expedia maps for yourself, but you CANNOT republish those maps. The tiger and terraserver/toposerver maps have no restrictions.

Acceptable formats for lat/lon are:

-93.49130	DegDec (decimal degrees)
W93.49130	DegDec (decimal degrees)
"-93 29.478"	MinDec (decimal minutes)
"W93 29.478"	MinDec (decimal minutes)
-93.29.478	MinDec (decimal minutes)
W93.29.478	MinDec (decimal minutes)
W 93° 29.478	Cut/paste from gc.com (note it is 3 arguments)
"-93 45 30"	DMS (degrees, minutes, seconds)

"*label*" can be any text and will be displayed by the waypoint. The default *label* is the coordinates in Min-Dec format, and can be explicitly selected with the *label* "@".

"*symbol*" can be these tiger-style symbols

cross, redstar, bluestar

<clr>pin

<clr>dot<size>

<clr> is red, grn, blu, org, pur, mag, brn, lgr, cyn, gry, wht  
e.g. redden10

"*symbol*" can also be these extensions:

cross,<color>,<size>

dot,<color>,<diameter>

<color> is any color allowed by convert(1)

<size> is the length in pixels of the crosses or the diameter of the dot.

circle,<color>,<radius>

circle,<color>,<radius>,<thick>

<radius> is in pixels, meters(m), kilometers(km), feet(ft), or miles(mi).

gc

Do geocaching.com circle of radius 0.1 miles

puzzle

Do geocaching.com circle of radius 2.0 miles

line,<color>,<thick>

Draw a line from the previous point

hline,<color>,<thick>

Draw a horizontal line

vline,<color>,<thick>

Draw a vertical line

xhair,<color>,<thick>

```

    Draw a crosshair
    <filename>.{gif,jpg,png}
    <filename>.{gif,jpg,png},xsize,ysize
    <filename>.{gif,jpg,png},xsize,ysize,xoff,yoff
    geocache-event geocache-hybrid geocache-multi geocache-regular
    geocache-unknown geocache-virtual geocache-webcam geocache-moving
    geocache-ifound-event geocache-ifound-hybrid geocache-ifound-multi
    geocache-ifound-regular geocache-ifound-unknown geocache-ifound-virtual
    geocache-ifound-webcam geocache-ifound-moving
    geocache-unfound-event geocache-unfound-hybrid
    geocache-unfound-multi geocache-unfound-regular
    geocache-unfound-unknown geocache-unfound-virtual
    geocache-unfound-webcam geocache-unfound-moving

```

The default *symbol* is "cross,red,10" and can be explicitly selected with the *symbol* "@".

## OPTIONS

### -a number

Use map source number/name: [gmap]

#### 1 mapblast/vicinity

2 expedia 3 tiger

#### 4 terraserver

5 toposerver (free USGS)

#### 6 gc 7 gc-icons

8 multimap (worldwide) 9 multimap-aerial (UK only)

13 tscom OR citipix OR globex OR tscom:citipix OR tscom:airphoto OR tscom:digitalglobe OR tscom:globex OR tscom:getmapping OR tscom:getmappingultra. Best is 22544:1 unless a terraserver.com member who sets TSCOM\_EMAIL and TSCOM\_PW in \$HOME.georc.

20 osm OR osmmapnik OR osmapnik

21 osmstatic

30 aolterra

40 gmap (Google Maps) OR gbike

41 gstatic 42 gstatic-hybrid 43 gstatic-terrain 44 gstatic-aerial

TMS: may not be available/current everywhere 91 tms-osm (OSM Tile Map Server tile.openstreetmap.org) 92 tms-osmcycle OR tms-ocm

- tms-transport OR tms-trans (experimental)
- tms-openptmap OR tms-pt
- tms-openrailwaymap OR tms-rail

#### 93 tms-osmde

(Roads German style)

- tms-humanitarian OR tms-hot 94 tms-mapquest OR tms-mq
- tms-openaerial OR tms-mqoa 95 tms-maptoolkit OR tms-mtk 96 tms-gpsies
- tms-thunderforest OR tms-outdoors

#### 97 tms-terrain

(Stamen, US only)

**98 tms-toner**

(Stamen)

**99 tms-watercolor**

(Stamen)

**- tms-gif-\***

(GIF TMS, append base URL)

- tms-jpg-\*
- tms-png-\*

**-a black**

Black map

**-a white**

White map

**-a gray** Gray map (for no map at all)**-a url** Don't generate a map, instead output a URL link.**-a file.png** Overlay existing gif or png image with waypoints.**-c** Label map with coordinates**-C** Force 1st comand line coordinate to be the center**-m** Do not display **marker**(s) (symbols)**-s scale** Map scale NNNNN:1 [0]

Units modifiers: K = 1,000 and M = 1,000,000

N.B. A 1024 pixel map at a scale of 10K is 2.26 miles.

Or specify the scale by image resolution: NNNmpp = meters/pixel, NNNfpp or NNNft = feet/pixel, NNNipp or NNNin = inches/pixel (6in res for some sources)

**-s 0** Autoscale. Use bounding box of waypoints.**-r radius**

Minimum 'radius' (square circle) for autoscaled map. Units are in degrees unless suffixed with km or mi.

**-R radius**

Maximum 'radius' (square circle) for autoscaled map. Units are in degrees unless suffixed with km or mi.

**-S symbol**Set the default *symbol* [cross,red,10]**-W width**

Width of image in pixels [1280]

**-H height**

Height of image in pixels [1024]

**-o file** Save map in file, do not display it. Also:**-o www**

Upload: put-rkkda rkkda/tmp 111.jpg

**-o www:file** Upload: put-rkkda rkkda/tmp file**-h file** Write an HTML imagemap to file. Requires **-t** and **-o**. If the file is +file, then append the map to the file.

- i** Use smaller icons and labels. Drop coordinates from *label*.
- t waypoints**  
A file of waypoints to plot in tabsep, GPX, LOC, geo-mystery, or in extended Tiger format:  
  
LONG, LAT:SYMBOL:LABEL:URL  
The map will be centered about the 1st command line coordinate. If there isn't one, it will be centered about the bounding box of the coordinates.
- g mins[,color]** Add a lat/lon grid every minutes (decimal allowed). Suffix mins with "d" for degrees. Grid lines are red unless "color" is specified.
- T title** Title to put on image.
- F footer**  
Footer to put on image.  
  
Escapes for -T and -F:  
%a positional params  
%A entire command line
- B** Show km bar scale
- b** Show mi bar scale **-j dir[,amt]** Jog the center of the picture to n/s/e/w/ne/se/nw/sw by 80%
- P file** Output gpsbabel polygon (square) to file
- D lvl** Debug level [0]
- U** Retrieve latest version of this script

Defaults can also be set with variables in file \$HOME/.georc:

```
MAPSRC=number; MAPSCALE=scale; MAPWIDTH=width; MAPHEIGHT=height;
MAPTEXTBG=white #Can also use #rrggbbaa and "none" for no box
MAPTEXTFG=black #Can also use #rrggbbaa
```

## EXAMPLES

A single waypoint displayed on a map, *label* is lat/lon:

```
geo-map 45.50.501 W93.23.609
```

Two waypoints, map scale determined automatically:

```
geo-map -s0 N44.48.938 W093.31.988 riley cross \
N44.49.245 W093.30.507 yogurt redstar
```

Many waypoints from a Tiger-style waypoint file:

```
geo-map -s0 -t /tmp/mngca/TwinCities.tiger
```

A mailable URL from a Tiger-style waypoint file:

```
geo-map -aurl -s0 -t /tmp/mngca/TwinCities.tiger
```

An HTML imagemap from a Tiger-style waypoint file:

```
geo-map -s0 -t test.tiger -h test.html -o test.png
```

A GIANT imagemap of Twin Cities area caches:

```
geo-map -a3 -s30k -W7400 -H7000 -m -o map.png 45 -93.25
geo-nearest -ogpx -n700 45 -93.25 > tc700.gpx
```

```
geo-map -a map.png -t tc700.gpx -s30k -o big.png -h big.html 45 -93.25
```

A google map with 0.1mi circles:

```
geo-map -S gc -a gmap -t ~/Caches/xxx.gpx -o www:xxx.html
```

A triangle with the centroid:

```
geo-map -aosm -s0 N29.29.730 W98.39.806 \  
N29.29.652 W98.39.943 a line,red,1 \  
N29.29.793 W98.39.954 b line,red,1 \  
N29.29.730 W98.39.806 c line,red,1 \  
n29.29.725 w98.39.901 e dot,red,1
```

## SEE ALSO

geo-code, geo-nearest, geo-pg, geo-waypoint, <http://geo.rkkda.com/>

**NAME**

**geo-morse** - Morse decoder

**SYNOPSIS**

*geo-morse* [*options*] [*morse*] ...

**DESCRIPTION**

Morse decoder. Reads from the command line or stdin.

**OPTIONS**

- a** Use American *morse* code instead of International *morse* code
- e** Encode *morse*
- s** Swap (dot is dash, dash is dot)
- D lvl** Debug level [0]

**EXAMPLES**

Decode (note the "--" to protect the shell):

```
$ geo-morse -- -.--- --- ..- .. / -- . ... .. -.- .  
your message
```

Encode and decode:

```
$ geo-morse -e abcdefghijklmnopqrstuvwxyz sos hello | geo-morse  
abcdefghijklmnopqrstuvwxyz sos hello
```

**NAME**

**geo-myfinds** - Schedule a Pocket Query containing your finds

**SYNOPSIS**

**geo-myfinds** [*options*]

**DESCRIPTION**

Schedule a Pocket Query containing your finds. GC limits them to every 3 days.

Crontab Entry:

```
# 3AM on the 1st, ..., 25th of the month (i.e. 4 days)
0 3 1,5,9,13,17,21,25 * * geo-myfinds
0 11 1,5,9,13,17,21,25 * * geo-pqdownload -n "My*" -z
```

Requires:

- A premium subscriber login at <http://www.geocaching.com>.

- **curl**

<http://curl.haxx.se/>

**OPTIONS**

-**u username**

Username for <http://www.geocaching.com>

-**p password**

Password for <http://www.geocaching.com>

-**U** Retrieve latest version of this script

-**D lvl** Debug level [0]

Defaults can also be set with variables in file `$HOME/.georc`:

```
PASSWORD=password; USERNAME=username;
```

**SEE ALSO**

[geo-demand](#), [geo-newest](#), [geo-found](#), [geo-placed](#), [geo-nearest](#), [geo-pqdownload](#), <http://geo.rkkda.com/>

**NAME**

**geo-mystery** - Copy tabsep from in to out, obeying ~/.geo-mystery

**SYNOPSIS**

**geo-mystery** [*options*]

**DESCRIPTION**

Copy tabsep from in to out, obeying ~/.**geo-mystery**

**OPTIONS**

**-m** Only mystery

**-D lvl** Debug level

**NAME**

**geo-nearest** - Fetch a list of nearest geocaches

**SYNOPSIS**

```

geo-nearest [options]
geo-nearest [options] latitude longitude
geo-nearest [options] latitude longitude cache-type
geo-nearest [options] zipcode
geo-nearest [options] u=<username>
geo-nearest [options] ul=<username>
geo-nearest [options] pq=<pocket-query>
geo-nearest [options] tx=<bookmark-id>
geo-nearest [options] -b bookmark
geo-nearest [options] guid=<bookmark-id>

```

**DESCRIPTION**

Fetch a list of nearest geocaches.

Requires: A premium member (\$30/yr) OR a basic member (free) login at: <http://www.geocaching.com>  
 Visit a cache page and click the "Download to EasyGPS" link at least once so you can read and agree to the license terms. Otherwise, you will not get any waypoint data.

**curl** <http://curl.haxx.se/>

**gpsbabel**  
<http://gpsbabel.sourceforge.net/>

**OPTIONS**

**-b** *bookmark*

Use list "*bookmark*" [none]

**-q** *query*

**geo-nearest:** Use PQ list "*query*" [none]

**-q** *search*

geo-newest: Use search "*search*" [none]

**-c**

Remove cookie file when done

**-f**

Do not report any found or unavailable caches

**-m**

Do not report any members-only caches

**-F**

Report caches found by the login 'username' as unfound

**-n num** Return "num" caches [20]

**-s**

Output short names for the caches (gpsbabel option)

**-I term** Include only caches with 'term' [\*]

**-X term**

Exclude caches with 'term' [**-unavail**] terms: ~ (exclude none), unfound, ifound, soc, unavail, regular, multi, virtual, webcam, event, hybrid, cito

**-r radius**

Display only caches with radius in miles (e.g. **-r 25**) Suffix the value with "km" for kilometers.

- M mystery**  
Use file 'mystery' for unknown/mystery/puzzle caches [/home/rick/.geo-mystery]. Awk Format:  
gcid lat lon comment  
i.e.: GC2CBVB n44.45.123 w93.00.321 Final
- u username**  
Username for http://www.geocaching.com
- p password**  
Password for http://www.geocaching.com
- o format**  
Output format, **-o?** for possibilities [gpsdrive] plus "gpsdrive.sql" for direct insertion into MySQL DB plus "map[,geo-map-opts]" to display a geo-map.
- O filename**  
Output file, if not stdout
- S** Alias for **-o** gpsdrive.sql
- d** For **-S**, just delete selected records
- P** For **-S**, purge all records of type **-t** Geocache\*
- t type** For **-ogpsdrive.sql**, the waypoint type [Geocache]
- H htmldir**  
Also fetch the printable HTML pages (slowly)
- L logdir**  
Also fetch the plain text log entries (slowly) For **-H** or **-L**, the limit is 1500 updated caches/day.
- ! "lpr -Plp"**  
Print HTML pages
- E var=val**  
Set environment "var" to "val" i.e. DATEFMT=0|1
- D lvl** Debug level [0]
- U** Retrieve latest version of this script

**DEFAULTS**

Defaults can also be set with variables in file \$HOME/.georc:

```
PASSWORD=password;   USERNAME=username;   SOC=0|1;
LAT=latitude;        LON=logitude;       GEOMYSTERY=/dev/null;
NUM=num;              UTFMT=format;        BABELFLAGS=-s;
SQLUSER=gast;         SQLPASS=gast;        SQLDB=geoinfo;
DATEFMT=[0|1];       CACHE_CACHE_MAX_AGE="1 day";
```

**DATE FORMATS**

Geocaching.com date formats that are compatible:

GC Format	Example	Compatible
YYYY-MM-DD	2011-07-13	yes
YYYY/MM/DD	2011/07/13	yes
MM/DD/YYYY	07/13/2011	yes
DD/MM/YYYY	13/07/2011	yes if DATEFMT=1 in \$HOME/.georc
DD.MM.YYYY	13.07.2011	yes if DATEFMT=1 in \$HOME/.georc
DD/Mmm/YYYY	13/Jul/2001	no
DD.Mmm.YYYY	13.Jul.2001	no
Mmm/DD/YYYY	Jul/13/2011	no

```
DD Mmm YY    13 Jul 11    yes (english only)
```

Change them here:

```
http://www.geocaching.com/account/ManagePreferences.aspx
```

## NOTE

A basic member will get caches very slow (20 cache pages per minute) because we have to get the actual cache pages. They will be stored in: `~/geo/caches/GCXXXX.html`. Of course, after running this command, `geo-html2gpx` could be run.

## EXAMPLES

Nearest 20 caches, display shortnames:

```
geo-nearest -s
```

Search nearest 500 caches for virtual caches not yet found:

```
geo-nearest -n500 -Ivirtual -Xifound
```

Add nearest 50 caches to a GpsDrive SQL database

```
geo-nearest -n50 -f -s -S
```

Purge the existing SQL database of all geocaches, and fetch 200 fresh ones...

```
geo-nearest -S -P -s -n200
```

640x480 map of nearest caches using map source 2:

```
geo-nearest -omap, "-a2 -W640 -H480"
```

Copy two cachers:

```
geo-nearest -n200 -Xifound -udyll1231 -pPW | awk '{print $1}' >1.foo
geo-nearest -n200 -Xifound -urickrich -pPW |awk '{print $1}' >2.foo
geo-gid -otabsep $(comm -12 1.foo 2.foo) >both
```

Fetch by owner placed:

```
geo-nearest u=team-deadhead
```

Fetch by owner found:

```
geo-nearest ul="AAA+of+MichigAn&sortdir=asc&sort=placed"
```

Fetch by tx method:

```
# nearby caches of this (puzzle) type, that I haven't found
geo-nearest -n500 -f -otabsep tx=40861821-1835-4e11-b666-8d41064d03fe |
    geo-mystery >> Caches/rick.ts
```

Also, `tx=webcam`, `tx=earth`, `tx=multi`, `tx=event`, `tx=virtual`, `tx=letter`, `tx=unknown`, `tx=trad` (`tx=reg` is an alias).

Fetch by *cache-type* method:

```
# nearby puzzles, that I haven't found from my HOME lat/lon
geo-nearest -n500 -f -otabsep '$LAT' '$LON' unknown |
    geo-mystery >> Caches/rick.ts
```

Also, cache-type is webcam, earth, multi, event, virtual, letter, unknown, trad (reg is an alias).

Fetch a *bookmark* list:

```
geo-nearest -b acro
or
geo-nearest guid=baae5bf9-4315-4874-b7fb-ac84c9585641
```

Fetch a PQ query:

```
geo-nearest -q "Needs Maintenance"
or
geo-nearest pq=08be103b-ffd1-4e27-992f-616e144e24df
```

## FILES

~/georc ~/.geo/caches/

## SEE ALSO

geo-newest, geo-found, geo-placed, geo-keyword, geo-code, geo-map, geo-waypoint, <http://geo.rkkda.com/>

**NAME**

**geo-newest** - Fetch a list of newest geocaches

**SYNOPSIS**

**geo-newest** [*options*] [*country*] [*state*]

**geo-newest** [*options*] [*state*]

**geo-newest** [*options*] [*state*] [*lat*] [*lon*]

**DESCRIPTION**

Fetch a list of newest geocaches. "*state*" is only available for USA.

Requires: A premium member (\$30/yr) OR a basic member (free) login at: <http://www.geocaching.com>  
Visit a cache page and click the "Download to EasyGPS" link at least once so you can read and agree to the license terms. Otherwise, you will not get any waypoint data.

**curl** <http://curl.haxx.se/>

**gpsbabel**  
<http://gpsbabel.sourceforge.net/>

**OPTIONS****-b bookmark**

Use list "bookmark" [none]

**-q query**

geo-nearest: Use PQ list "query" [none]

**-q search**

**geo-newest**: Use search "search" [none]

**-c** Remove cookie file when done

**-f** Do not report any found or unavailable caches

**-m** Do not report any members-only caches

**-F** Report caches found by the login 'username' as unfound

**-n num** Return "num" caches [20]

**-s** Output short names for the caches (gpsbabel option)

**-I term** Include only caches with 'term' [\*]

**-X term**

Exclude caches with 'term' [**-unavail**] terms: ~ (exclude none), unfound, ifound, soc, unavail, regular, multi, virtual, webcam, event, hybrid, cito

**-r radius**

Display only caches with radius in miles (e.g. **-r 25**) Suffix the value with "km" for kilometers.

**-M mystery**

Use file 'mystery' for unknown/mystery/puzzle caches [/home/rick/.geo-mystery]. Awk Format:

gcid lat lon comment

i.e.: GC2CBVB n44.45.123 w93.00.321 Final

**-u username**

Username for <http://www.geocaching.com>

**-p password**

Password for <http://www.geocaching.com>

- o format**  
Output format, **-o?** for possibilities [gpsdrive] plus "gpsdrive.sql" for direct insertion into MySQL DB plus "map[,geo-map-opts]" to display a geo-map.
- O filename**  
Output file, if not stdout
- S** Alias for **-o** gpsdrive.sql
- d** For **-S**, just delete selected records
- P** For **-S**, purge all records of type **-t** Geocache\*
- t type** For **-ogpsdrive.sql**, the waypoint type [Geocache]
- H htmdir**  
Also fetch the printable HTML pages (slowly)
- L logdir**  
Also fetch the plain text log entries (slowly) For **-H** or **-L**, the limit is 1500 updated caches/day.
- ! "lpr -Plp"**  
Print HTML pages
- E var=val**  
Set environment "var" to "val" i.e. DATEFMT=0|1
- D lvl** Debug level [0]
- U** Retrieve latest version of this script

## DEFAULTS

Defaults can also be set with variables in file \$HOME/.georc:

```
PASSWORD=password;   USERNAME=username;   SOC=0|1;
LAT=latitude;        LON=logitude;        GEOMYSTERY=/dev/null;
NUM=num;              UTFMT=format;         BABELFLAGS=-s;
SQLUSER=gast;        SQLPASS=gast;         SQLDB=geoinfo;
DATEFMT=[0|1];       CACHE_CACHE_MAX_AGE="1 day";
```

## DATE FORMATS

Geocaching.com date formats that are compatible:

GC Format	Example	Compatible
YYYY-MM-DD	2011-07-13	yes
YYYY/MM/DD	2011/07/13	yes
MM/DD/YYYY	07/13/2011	yes
DD/MM/YYYY	13/07/2011	yes if DATEFMT=1 in \$HOME/.georc
DD.MM.YYYY	13.07.2011	yes if DATEFMT=1 in \$HOME/.georc
DD/Mmm/YYYY	13/Jul/2001	no
DD.Mmm.YYYY	13.Jul.2001	no
Mmm/DD/YYYY	Jul/13/2011	no
DD Mmm YY	13 Jul 11	yes (english only)

Change them here:

<http://www.geocaching.com/account/ManagePreferences.aspx>

## NOTE

A basic member will get caches very slow (20 cache pages per minute) because we have to get the actual cache pages. They will be stored in: ~/.geo/caches/GCXXXX.html. Of course, after running this command, geo-html2gpx could be run.

**EXAMPLES**

Add newest 50 caches to a GpsDrive SQL database

```
geo-newest -n50 -f -s -S MN
```

Purge the existing SQL database of all geocaches, and fetch 200 fresh ones...

```
geo-newest -S -P -s -n200 MN
```

Create a GPX file of all caches in MN, including all logs. This will take several hours to run, and should only be run at night.

```
geo-newest -X "" -n2000 -D1 -H html MN > junk  
geo-html2gpx -b html/*.html > all-mn.gpx
```

Fetch *country* Iraq:

```
geo-newest -s Iraq
```

Fetch *country* Germany, *state* Berlin:

```
geo-newest -s Germany Berlin
```

Fetch *country* Germany, *state* Berlin by code:

```
geo-newest -s c79 s137
```

Fetch by search:

```
geo-newest -f -q unknown
```

Also, tx=webcam, tx=earth, tx=multi, tx=event, tx=virtual, tx=letter, tx=unknown, tx=trad (tx=reg is an alias).

**FILES**

```
~/georc ~/geo/caches/
```

**SEE ALSO**

geo-countries-states geo-nearest, geo-found, geo-placed, geo-keyword, geo-code, geo-map, geo-waypoint,  
<http://geo.rkkda.com/>

**NAME**

**geo-nonogram** - Nonogram solver

**SYNOPSIS**

**geo-nonogram** [*options*] skel.nono ...

**DESCRIPTION**

Nonogram solver using a program by Steven Simpson.

**OPTIONS**

**-v** Clear screen and print the grid while it is being solved in case of Warning: puzzle has imbalance

**-p png-file**

Convert and image it into a PNG png-file

**-P pdf-file**

Convert and image it into a PDF pdf-file

**-D lvl** Debug level

**EXAMPLE**

Solve GC6EQAP.nono:

```

$ geo-nonogram GC6EQAP.nono
ââ   ââ   ââââââââ   ââââââââ   ââ
ââ   ââ   ââ   ââ   ââ   ââ
ââââââââ   ââââââââ   ââââââââ   ââââââââââââ
      ââ       ââ       ââ   ââ   ââ
      ââ       ââ   ââââââââ   ââââââââââââ

ââ       ââââââââââ   ââââââââââ   ââââââââââ
ââ       ââ   ââ   ââ   ââ   ââ   ââ
ââââââââââ   ââââââââââ   ââ   ââ   ââ   ââ
ââ   ââ       ââ   ââ   ââ   ââ   ââ
ââââââââââ       ââ   ââââââââââ   ââââââââââ

ââââââââââ   ââââââââ   ââââââââââ   ââââââââââ
ââ   ââ       ââ   ââ   ââ   ââ       ââ
ââââââââââ       ââ   ââ   ââ   ââ   ââââââââââ
      ââ       ââ   ââ   ââ   ââ
      ââ       ââ   ââââââââââ   ââââââââââ

ââââââââââ   ââââââââââ   ââââââââââ
ââ   ââ   ââ       ââ   ââ       ââ   ââ
ââââââââââ   ââââââââââ   ââââââââââ       ââ
ââ   ââ       ââ   ââ   ââ   ââ   ââ   ââ
ââââââââââ   ââââââââââ   ââââââââââ       ââââââââ

```

**FORMAT**

Format of a ".nono":

```

$ cat skel.nono
title "skeleton"
by "skeletin"
width 20
height 20

```

geo-nonogram(1)

geo-nonogram(1)

rows  
1 2 3  
1 2  
etc.

columns  
1 1  
2 1 3  
etc.

**SEE ALSO**

<http://www.comp.lancs.ac.uk/~ss/nonogram/auto>

<http://www.research.lancs.ac.uk/portal/en/people/Steven-Simpson/>

<https://forge.comp.lancs.ac.uk/svn-repos/nonograms/nonolib/trunk/>

<https://forge.comp.lancs.ac.uk/svn-repos/nonograms/nonogram/trunk/>

<http://webpbn.com/export.cgi>

pbnsolve-wrapper

**NAME**

**geo-ocr** - Do OCR on an image file

**SYNOPSIS**

**geo-ocr** [*options*] *image*

**DESCRIPTION**

Do OCR on an *image* file using tesseract.

**OPTIONS**

**-l lang** Language, including: pigpen (pig), pigpen2, dancing-men (men), aurebesh (starwars)

**-D lvl** Debug level

**EXAMPLES**

Convert dancing men:

```
$ geo-ocr -l men xxx.jpg
Tesseract Open Source OCR Engine v3.02.02 with Leptonica
NORTHFIFTYNINEDEGREESFO
RTY SEVENDOTNINEHUNDREDS
EVENTYFIVEEASTHSEVENTEEN
DEGREESTHIRTYSEVE NDOTEI
GHTY FIVEDOTDOT DOTDOTDOT
```

**FILES**

/usr/share/tesseract/tessdata/\*.traineddata

/usr/local/share/tessdata/\*.traineddata

**SEE ALSO**

<http://www.rkkda.com/sw/ocr/>

**NAME**

**geo-phone2word** - Convert telephone numbers to **word(s)**

**SYNOPSIS**

**geo-phone2word** [*options*] [*numbers*]

**DESCRIPTION**

Convert telephone *numbers* to **word(s)**. It can use command line "*numbers*" or stdin. Also, there are ROT-13 versions of zero, one, ... ninety and north, south, east, west, hundred, etc.

**OPTIONS**

- a** Use ancient text, 1st digit is place, 2nd digit is 1, 2, 3, 4  
i.e. MTS Audio Response Unit (IBM 7772) developed in 1964
- c** Use count instead
- D lvl** Debug level

**EXAMPLE**

```

Convert:
$ geo-phone2word 43246
idaho

$ geo-phone2word -c 66 666 777 8 44
north

$ geo-phone2word 7243 227 488345 48243627 78884 8427377 282745 227
7243: said rage sage paid page raid four(rot13)
227: bar bas cap bbq cbs abs car bbs one(rot13)
488345: thirty(rot13)
48243627: thousand(rot13)
78884: eight(rot13)
8427377: hundred(rot13)
282745: ninety(rot13)
227: bar bas cap bbq cbs abs car bbs one(rot13)

$ geo-phone2word -a
N8191326281930744392R33
63738193T744392H814273
32327432833262636232
W324341428193E94327363
S819163T74328332628191
32628193
twentysixfortysixthreesevenoneeightyzerotwoseventwenty
    
```

**SEE ALSO**

\$HOME/lib/geo/english.dic  
/usr/lib/geo/english.dic

**NAME**

**geo-placed** - Fetch a list of geocaches placed by a user

**SYNOPSIS**

**geo-placed** [*options*] [*username*]

**geo-placed** [*options*] [*username*] [*lat*] [*lon*]

**DESCRIPTION**

Fetch a list of geocaches placed by a specific user.

Requires: A premium member (\$30/yr) OR a basic member (free) login at: <http://www.geocaching.com>  
Visit a cache page and click the "Download to EasyGPS" link at least once so you can read and agree to the license terms. Otherwise, you will not get any waypoint data.

**curl** <http://curl.haxx.se/>

**gpsbabel**  
<http://gpsbabel.sourceforge.net/>

**OPTIONS****-b bookmark**

Use list "bookmark" [none]

**-q query**

geo-nearest: Use PQ list "query" [none]

**-q search**

geo-newest: Use search "search" [none]

**-c** Remove cookie file when done

**-f** Do not report any found or unavailable caches

**-m** Do not report any members-only caches

**-F** Report caches found by the login '*username*' as unfound

**-n num** Return "num" caches [20]

**-s** Output short names for the caches (gpsbabel option)

**-I term** Include only caches with 'term' [\*]

**-X term**

Exclude caches with 'term' [*\_NoThInG\_*] terms: ~ (exclude none), unfound, ifound, soc, unavail, regular, multi, virtual, webcam, event, hybrid, cito

**-r radius**

Display only caches with radius in miles (e.g. **-r 25**) Suffix the value with "km" for kilometers.

**-M mystery**

Use file 'mystery' for unknown/mystery/puzzle caches [*/home/rick/.geo-mystery*]. Awk Format:

*gcid lat lon comment*

i.e.: GC2CBVB n44.45.123 w93.00.321 Final

**-u username**

Username for <http://www.geocaching.com>

**-p password**

Password for <http://www.geocaching.com>

- o format**  
Output format, **-o?** for possibilities [gpsdrive] plus "gpsdrive.sql" for direct insertion into MySQL DB plus "map[,geo-map-opts]" to display a geo-map.
- O filename**  
Output file, if not stdout
- S** Alias for **-o** gpsdrive.sql
- d** For **-S**, just delete selected records
- P** For **-S**, purge all records of type **-t** Geocache\*
- t type** For **-ogpsdrive.sql**, the waypoint type [Geocache]
- H htmdir**  
Also fetch the printable HTML pages (slowly)
- L logdir**  
Also fetch the plain text log entries (slowly) For **-H** or **-L**, the limit is 1500 updated caches/day.
- ! "lpr -Plp"**  
Print HTML pages
- E var=val**  
Set environment "var" to "val" i.e. DATEFMT=0|1
- D lvl** Debug level [0]
- U** Retrieve latest version of this script

## DEFAULTS

Defaults can also be set with variables in file \$HOME/.georc:

```
PASSWORD=password;   USERNAME=username;   SOC=0|1;
LAT=latitude;        LON=logitude;        GEOMYSTERY=/dev/null;
NUM=num;              UTFMT=format;         BABELFLAGS=-s;
SQLUSER=gast;         SQLPASS=gast;         SQLDB=geoinfo;
DATEFMT=[0|1];       CACHE_CACHE_MAX_AGE="1 day";
```

## DATE FORMATS

Geocaching.com date formats that are compatible:

GC Format	Example	Compatible
YYYY-MM-DD	2011-07-13	yes
YYYY/MM/DD	2011/07/13	yes
MM/DD/YYYY	07/13/2011	yes
DD/MM/YYYY	13/07/2011	yes if DATEFMT=1 in \$HOME/.georc
DD.MM.YYYY	13.07.2011	yes if DATEFMT=1 in \$HOME/.georc
DD/Mmm/YYYY	13/Jul/2001	no
DD.Mmm.YYYY	13.Jul.2001	no
Mmm/DD/YYYY	Jul/13/2011	no
DD Mmm YY	13 Jul 11	yes (english only)

Change them here:

<http://www.geocaching.com/account/ManagePreferences.aspx>

## NOTE

A basic member will get caches very slow (20 cache pages per minute) because we have to get the actual cache pages. They will be stored in: ~/.geo/caches/GCXXXX.html. Of course, after running this command, geo-html2gpx could be run.

**EXAMPLES**

List the most recent 50 caches placed by dyl1231:

```
geo-placed -s -n50 dyl1231
```

List the most recent caches placed by dyl1231 that are with a radius of 15 miles of your home location:

```
geo-placed -s -r15M dyl1231
```

List the most recent caches placed by dyl1231 that are with a radius of 15 miles of a specific location:

```
geo-placed -s -r50 dyl1231 N47.20.000 W121.30.000
```

Display a map of the 20 newest caches placed by dyl1231:

```
geo-placed -omap,-a2 -F dyl1231
```

Make a backup copy of all of my caches placed (can take awhile):

```
geo-placed -n999 -H descdir -L logdir -otabsep > placed.tabsep
```

**FILES**

```
~/.georc ~/.geo/caches/
```

**SEE ALSO**

geo-found, geo-nearest, geo-newest, geo-keyword, geo-code, geo-waypoint, <http://geo.rkkda.com/>

**NAME**

**geo-pocket-query-newest** - newest pocket queries...

**SYNOPSIS**

**geo-pocket-query-newest** [*options*]

**DESCRIPTION**

Newest pocket queries based on first word is equal to RE [A-Z][A-Z]. Runs "firefox /tmp/newest.html".

Requires:

- A premium subscriber login at <http://www.geocaching.com>.

**- curl**

<http://curl.haxx.se/>

**OPTIONS****-u username**

Username for <http://www.geocaching.com>

**-p password**

Password for <http://www.geocaching.com>

**-U** Retrieve latest version of this script

**-D lvl** Debug level [0]

Defaults can also be set with variables in file `$HOME/.georc`:

```
PASSWORD=password; USERNAME=username;
```

**EXAMPLES**

i.e.:

```
$ geo-pocket-query-newest
AZ puzzles
FL puzzles
HI puzzles
IA puzzles
MB Manitoba Puzzles
MI puzzles - UP
MN puzzles
MT puzzles
ND puzzles
NE puzzles
NL puzzles
ON Ontario Puzzles
SD puzzles
SO SOCs
WI Milwaukee puzzles
WI puzzles
WY puzzles
```

**SEE ALSO**

geo-demand, geo-newest, geo-found, geo-placed, geo-nearest, geo-pqdownload, <http://geo.rkkda.com/>

**NAME**

**geo-poi** - Lookup places in Place Guide or POI Factory files

**SYNOPSIS**

**geo-poi** [*options*] pg.pdb ...

**geo-poi** [*options*] lat lon

**geo-poi** [*options*] last

**geo-poi** [*options*] place.csv ...

**DESCRIPTION**

**geo-poi** [*options*] pg.pdb ...

Lookup place locations in Mapopolis in pg.pdb Place Guide files, and format them for output in any of the output file types that gpsbabel supports, or directly import them into the GpsDrive MySQL waypoint database.

**geo-poi** [*options*] lat lon

Like the above, except derive the implied list of searched PDB files by consulting an index of placeguide PDB files in /home/rick/.geopoi. A PDB index can be produced with a command like this:

```
for i in */*.pdb; do pgpdb2txt -r `pwd`/$i; done > ~/.geopoi
```

**geo-poi** [*options*] last

Like the above, except determine the current lat/lon from GpsDrive's lastlong/lastlat values in the \$HOME/.gpsdrive/gpsdriverc file.

**geo-poi** [*options*] places.csv ...

Lookup place locations in POI Factory places.csv files, and format them for output in any of the output file types that gpsbabel supports, or directly import them into the GpsDrive MySQL waypoint database.

Requires:

**curl** <http://curl.haxx.se/>

**gpsbabel**  
<http://gpsbabel.sourceforge.net>

**OPTIONS****-c category**

Select category [\*] Can use shell-style \* globbing to match the category

**-n name**

Select name [\*] Can use shell-style \* globbing to match the name

**-l**

List available categories in .pdb files and exit With -c, list raw records in category and exit

**-t type** The waypoint type to output [<category-in-pdb-file>]

**-o format**

Output format, -o? for possibilities [gpsdrive] plus "gpsdrive.sql" for direct insertion into MySQL DB plus "txt" for raw text records

- o mindec** Output *lat/lon* in MinDec (44.56.123) mode.
- S** Alias for **-o** gpsdrive.sql
- d** For **-S**, just delete selected records0
- r radius** Radius value for implied list of PDB files [5]
- D lvl** Debug level
- U** Retrieve latest version of this script

**EXAMPLES**

- geo-poi -cHotels** FredericksburgCityVA-PG.pdb
- geo-poi -cRest\* -t** Restaurant FredericksburgCityVA-PG.pdb
- geo-poi -S -c** Dunn ~/poi/Dunn\_Brothers\_Coffee.csv

**SEE ALSO**

geo-code, geo-waypoint, geo-nearest, <http://geo.rkkda.com/>

**NAME**

**geo-polygon** - Compute the centroid of a polygon

**SYNOPSIS**

**geo-polygon** [*options*] *lat0 lon0 lat1 lon1 lat2 lon2 ...*

**DESCRIPTION**

Compute the centroid of a polygon.

Acceptable formats for lat/lon are:

- 93.49130            DegDec (decimal degrees)
- W93.49130           DegDec (decimal degrees)
- "-93 29.478"        MinDec (decimal minutes)
- "W93 29.478"        MinDec (decimal minutes)
- 93.29.478          MinDec (decimal minutes)
- W93.29.478          MinDec (decimal minutes)
- W 93° 29.478        Cut/paste from gc.com (note it is 3 arguments)
- "-93 45 30"         DMS (degrees, minutes, seconds)

By default, UTM is used for the calculation. With **-m**, a spherical earth is used for the calculation.

**OPTIONS**

- m**     Midpoint from <http://www.geomidpoint.com/>
- M**     Center of gravity (midpoint) with weights
- t**     Do various centers of triangles iff you have sympy installed
- D lvl** Debug level

**EXAMPLES**

MinDec input for a 9 sided polygon:

```
$ geo-polygon n42.00.126 w83.58.037 \
    n42.00.318 w83.57.586 \
    n42.00.120 w83.57.668 \
    n42.00.097 w83.56.769 \
    n41.59.988 w83.56.852 \
    n41.59.599 w83.56.686 \
    n41.59.987 w83.57.700 \
    n41.59.749 w83.57.744
n41.59.970 w83.57.331
```

A Triangle:

```
$ geo-polygon N42.33.767 W83.07.617 N42.33.736 W83.07.594 \
    N42.33.736 W83.07.640
n42.33.746 w83.07.617
```

Tetrahedron using a triangle:

```
$ geo-polygon -m -- -17.6829061279 175.937938962 \
    26.7978332041 72.5860786255 \
    -52.1980005058 -15.0157003564
Normal:            s40.01.127 e96.50.457
Antipod:           n40.01.127 w83.09.543
```

A triangle using **-t**:

```
$ geo-polygon -t N29.42.976 W82.29.858 N29.37.676 W82.24.744 \  
N29.44.891 W82.21.642  
Center:          n29.41.849 w82.25.415  
Circumcenter:   n29.41.919 w82.25.134  
Incenter:       n29.41.795 w82.25.556  
Orthocenter:    n29.41.707 w82.25.976
```

Center of gravity (midpoint) with weights:

```
$ geo-polygon -M 34.663400 135.531433 8805 \  
35.016783 135.677433 2644 \  
34.586766 135.773233 1443  
Normal:         n34.43.640 e135.35.303  
Antipod:        s34.43.640 w44.24.697
```

### SEE ALSO

<http://www.sympy.org/en/index.html>

<http://www.geomidpoint.com/>

**NAME**

**geo-pqdownload** - Perform a Pocket Query **download(s)**

**SYNOPSIS**

**geo-pqdownload** [*options*]

**DESCRIPTION**

Pocket Query download. For PQ's from 501 to 1000 waypoints, because the gc site does not email them (as of May 10, 2010). Go figure!!!

**OPTIONS**

- c** Remove cookie file when done
- d** Delete the files from the server
- n NAME**  
Search for NAMEs, globbing allowed
- t name**  
Construct name using strftime specifiers, PLUS %+ for the actual name. I.E. **-t %m%d-%+**
- z** Unzip the files
- u username**  
Username for http://www.geocaching.com
- p password**  
Password for http://www.geocaching.com
- U** Retrieve latest version of this script
- D lvl** Debug level [0]

Defaults can also be set with variables in file \$HOME/.georc:

```
PASSWORD=password;  USERNAME=username;
LAT=latitude;      LON=logitude;
```

**EXAMPLES**

Download all files:

```
$ geo-pqdownload -d -z
mn-09 http://www.geocaching.com/pocket/downloadpq.ashx?g=ba2e0520...
mn-28 http://www.geocaching.com/pocket/downloadpq.ashx?g=41a95f02...
mn-29 http://www.geocaching.com/pocket/downloadpq.ashx?g=cff93db9...
mn-30 http://www.geocaching.com/pocket/downloadpq.ashx?g=e5049240...
$ ls *.zip *.gpx
mn-09.gpx      mn-28.gpx      mn-29.gpx      mn-30.gpx
mn-09-wpts.gpx  mn-28-wpts.gpx  mn-29-wpts.gpx  mn-30-wpts.gpx
mn-09.zip      mn-28.zip      mn-29.zip      mn-30.zip
```

Download "My Finds.." files:

```
$ geo-pqdownload -d -z -n "My*"
```

Download "My Finds.." files prefixed with year-month-day:

```
$ geo-pqdownload -d -z -n "My*" -t %Y-%m-%d-%+
$ ls *My*
2011-05-06-My Finds Pocket Query.zip
```

Download all files *except* the "My\*" file:

```
$ geo-pqdownload -n "!(My*)"
```

**SEE ALSO**

geo-countries-states geo-newest, geo-found, geo-placed, geo-nearest, strftime, <http://geo.rkkda.com/>

**NAME**

**geo-pqs** - Run PQ's to get all caches in a state or country

**SYNOPSIS**

**geo-pqs** [*options*] *state\_or\_country*

**geo-pqs** [*options*] *country state*

**DESCRIPTION**

Run pocket queries to get all caches in a *state* or *country*.

**OPTIONS**

**-c** Crontab output

**-w** Weekly crontab output, with **-c**

**-l PQLIM**

Limit PQ size to PQLIM (500)

**-n NUM**

Limit total caches to NUM (99999)

**-d N[+-]**

Difficulty level [1+]

**-t N[+-]**

Terrain level [1+]

**-f** Do not report any found or unavailable caches

**-q qualifiers**

Limit by one or more space/comma separated qualifiers:

Type: these ones OR together....

traditional, multi, virtual, letterbox, event,  
mystery, webcam, locationless, trash

Container: these ones OR together....

small, other, none, large, regular, micro, unknown

These ones AND together....

ifound, notfound, bug, unfound, notowned,  
new, iown, watchlist, updated, active, notactive,  
notign, found7, soc, notsoc

**-D lvl** Debug level

**EXAMPLES**

List PQs:

```
$ geo-pqs mn
geo-demand -n500 -T'11/05/2000-11/10/2003' -Nmn-00 mn #499 caches
geo-demand -n500 -T'11/11/2003-10/17/2004' -Nmn-01 mn #497 caches
geo-demand -n500 -T'10/18/2004-07/27/2005' -Nmn-02 mn #498 caches
geo-demand -n500 -T'07/28/2005-01/28/2006' -Nmn-03 mn #494 caches
geo-demand -n500 -T'01/29/2006-05/09/2006' -Nmn-04 mn #500 caches
geo-demand -n500 -T'05/10/2006-07/27/2006' -Nmn-05 mn #500 caches
geo-demand -n500 -T'07/28/2006-10/05/2006' -Nmn-06 mn #498 caches
geo-demand -n500 -T'10/07/2006-12/23/2006' -Nmn-07 mn #492 caches
geo-demand -n500 -T'12/24/2006-03/31/2007' -Nmn-08 mn #495 caches
geo-demand -n500 -T'04/01/2007-05/27/2007' -Nmn-09 mn #487 caches
geo-demand -n500 -T'05/28/2007-07/13/2007' -Nmn-10 mn #494 caches
```

```
geo-demand -n500 -T'07/14/2007-' -Nmn-11 mn #107 caches
```

Crontab for rick:

```
$ crontab -l
34 1 * * 0      geo-demand -n500 -T'11/05/2000-11/10/2003' -Nmn-00 mn
34 1 * * 1      geo-demand -n500 -T'11/11/2003-10/17/2004' -Nmn-01 mn
34 1 * * 2      geo-demand -n500 -T'10/18/2004-07/27/2005' -Nmn-02 mn
34 1 * * 3      geo-demand -n500 -T'07/28/2005-01/28/2006' -Nmn-03 mn
34 1 * * 4      geo-demand -n500 -T'01/26/2006-05/09/2006' -Nmn-04 mn
34 1 * * 5      geo-demand -n500 -T'05/10/2006-07/27/2006' -Nmn-05 mn
34 1 * * 6      geo-demand -n500 -T'07/28/2006-10/05/2006' -Nmn-06 mn
39 1 * * 1,3,5  geo-demand -n500 -T'10/07/2006-12/23/2006' -Nmn-07 mn
39 1 * * 0,2,4,6 geo-demand -n500 -T'12/24/2006-03/31/2007' -Nmn-08 mn
44 1 * * 1,3,5  geo-demand -n500 -T'04/01/2007-05/27/2007' -Nmn-09 mn
44 1 * * 0,2,4,6 geo-demand -n500 -T'05/28/2007-07/13/2007' -Nmn-10 mn
49 1 * * *      geo-demand -n500 -T'07/14/2007-' -Nmn-11 mn #107 caches 07/23/07
```

Filter:

```
$ geo-pqs -qtrad,small,regular,large -d2- mn
```

Country and State:

```
$ geo-pqs Australia "Northern Territory"
```

## SEE ALSO

geo-demand, geo-countries-states

**NAME**

**geo-procmail** - procmailrc script for geocaching

**SYNOPSIS**

**geo-procmail** [*options*]

**DESCRIPTION**

This is a procmailrc script for geocaching. It will turn the "http://www.geocaching.com/seek/cache\_details.aspx?..." into "http://www.geocaching.com/seek/cdpf.aspx?" so that you can get the print-friendly pages.

**EXAMPLE**

In \$HOME/.procmailrc:

```
#
#      GEO: Print friendly, decrypt
#
:0f
* ^Subject:.*GEO] Notify: Surfer Joe
| geo-procmail
```

**OPTIONS**

**-D lvl** Debug level

**NAME**

**geo-project** - Project a waypoint

**SYNOPSIS**

**geo-project** [*options*] *lat1 lon1 distance bearing*

**DESCRIPTION**

Project a waypoint.

lat/lon can be specified in DegDec or dotted MinDec format.

*distance* is in miles unless suffixed with mil, engchain, chain, fathom, rod, furlong, hand, link, pace, fizzy, smoot, verst, in, ft, yd, km, or m.

*bearing* is in compass degrees unless suffixed with mil, grad, or rad, or n, nne, ne, ene, e, ese, se, sse, s, ssw, sw, wsw, w, wnw, nw, nnw. If the *bearing* is a negative number, then calculate in the reverse to:from instead of from:to.

**OPTIONS**

- e** Use WGS 1984 ellipsoid calculation method [default]
- u** Use UTM calculation method
- s rad** Use spherical calculation method with radius = rad
- l** Output decimal latitude only (for scripts)
- L** Output decimal longitude only (for scripts)
- D lvl** Debug level

**EXAMPLES**

Project a waypoint 13147.2 feet at 38 degrees:

```
$ geo-project 44.47.151 -93.14.094 13147.2ft 38
wp = 44.814260 -93.203712      n44.48.856 w93.12.223
```

Project a spherical waypoint 402.31 meters at 228.942 degrees:

```
$ geo-project -s 6378000 N42.43.919 W84.28.929 402.31m 228.942
wp = 42.729609 -84.485860      n42.43.777 w84.29.152
```

**SEE ALSO**

[https://en.wikipedia.org/wiki/Earth\\_ellipsoid](https://en.wikipedia.org/wiki/Earth_ellipsoid)

**NAME**

**geo-rehides** - Output a new GPX file containing just rehides

**SYNOPSIS**

**geo-rehides** *finder-name* file.gpx > rehides.gpx

Requires: A *subscriber login* at <http://www.geocaching.com>.

**DESCRIPTION**

Output a new GPX file containing just rehides as far as *finder-name* is concerned (e.g. date placed > date found).

**NAME**

**geo-rotate-text** - Rotate text CW, CCW, or flip

**SYNOPSIS**

**geo-rotate-text** [*options*]

**DESCRIPTION**

Rotate text clockwise, counter-clockwise, or flip around.

**OPTIONS**

**-cw** Rotate clockwise  
**-ccw** Rotate counter-clockwise  
**-f** Flip  
**-D lvl** Debug level

**EXAMPLES**

Assume "xxx" has the lines in it:

```
abcd  
efgh
```

Do nothing:

```
$ geo-rotate-text -ccw < xxx | geo-rotate-text -cw  
abcd  
efgh
```

Clockwise:

```
$ geo-rotate-text -cw < xxx  
ea  
fb  
gc  
hd
```

Counter-clockwise:

```
$ geo-rotate-text -ccw < xxx  
dh  
cg  
bf  
ae
```

Flip:

```
$ geo-rotate-text -f < xxx  
ae  
bf  
cg  
dh
```

**NAME**

**geo-sdt** - Replace Size, Difficulty, Terrain from a PQ file

**SYNOPSIS**

**geo-sdt** [*options*] pq.ts

**DESCRIPTION**

Replace Size, Difficulty, Terrain in a tabsep file from a Pocket Query. Read it from stdin and write it to stdout.

This is used for geo-nearest/geo-newest.

**OPTIONS**

**-H** Also do hints

**-D lvl** Debug level

**EXAMPLES**

Replace:

```
gpsbabel -i gpx -f ~/Caches/mn.gpx -o tabsep -F ~/Caches/mn.ts  
geo-nearest -otabsep | geo-sdt ~/Caches/mn.ts > ~/Caches/rick.ts
```

**SEE ALSO**

**update-caches(1)**

**NAME**

**geo-soon** - Outputs a list of submitted but unapproved caches

**SYNOPSIS**

**geo-soon** [*options*]

**geo-soon** [*options*] *lat lon*

**DESCRIPTION**

Outputs a list of submitted but unapproved caches. This script first finds the cache number of the most recently submitted cache, anywhere in the world. Then, beginning "**-n num**" (1000) caches before that, it fetches the LOC information for caches that have been submitted but are not yet (and may never be) approved. It filters these caches to the specified radius around your *lat/lon*.

The exploit that makes this possible is that the LOC info is returned even though the caches aren't approved.

The amount of information available for these caches is very limited. The GC id, cache name and owner, and *lat/lon* are all that you get.

It would be foolish, and unethical, to search for these caches before they are approved. They could be puzzles, multis, virtuals -- there is no way to know. But, you might use this information to keep a closer vigil on new approvals. Whether that is ethical or not is up to you. I'm just the toolsmith -- if a tool \*can\* be written, I'm inclined to write it. Kinda like a gun maker.

Another use is to check up on your approver, to see how long approvals are taking.

IMPORTANT: The **-r** radius flag limits the output to your area! Otherwise, this command will take a long time to run.

**OPTIONS**

**-c** Remove cookie file when done

**-n num** Search within the last 'num' caches [1000]

**-s** Output short names for the caches (gpsbabel option)

**-r radius**  
Show only caches within radius (e.g. **-r 25M**) [35]

**-u username**  
Username for <http://www.geocaching.com>

**-p password**  
Password for <http://www.geocaching.com>

**-o format**  
Output format, **-o?** for possibilities [gpsdrive] plus "gpsdrive.sql" for direct insertion into MySQL DB plus "map[,geo-map-opts]" to display a geo-map.

**-O filename**  
Output file, if not stdout

**-S** Alias for **-o gpsdrive.sql**

**-d** For **-S**, just delete selected records

**-P** For **-S**, purge all records of type **-t Geocache-soon\***

**-t type** For **-ogpsdrive.sql**, the waypoint type [Geocache-soon]

**-D lvl** Debug level [0]

**-U** Retrieve latest version of this script

Defaults can also be set with variables in file \$HOME/.georc:

```
PASSWORD=password;  USERNAME=username;  SOC=0|1;
LAT=latitude;      LON=logitude;
NUM=num;           OUTFMT=format;      BABELFLAGS=-s;
SQLUSER=gast;      SQLPASS=gast;      SQLDB=geoinfo;
```

## EXAMPLES

**geo-soon**

**geo-soon** N33.48.566 W117.50.099

## SEE ALSO

geo-newest, geo-found, geo-placed, geo-nearest, <http://geo.rkkda.com/>

**NAME**

**geo-state** - Get a state

**SYNOPSIS**

*geo-state* [*options*] *state*

**DESCRIPTION**

Get *state* by:

```
cd ~/Caches
geo-newest -n4000 -H $ss/tmp $SS >/dev/null
geo-html2gpx $ss/tmp/*.html >$ss/$ss.gpx
(cd $ss; gpx2html)
geo-2gpsdrive -s -S -igpx $ss/$ss.gpx
```

**OPTIONS**

**-D lvl** Debug level

**EXAMPLE**

*geo-state* sd

**NAME**

**geo-sub** - Do a substitution (caesar) cipher for all shifts.

**SYNOPSIS**

**geo-sub** [*options*] [*text ...*] [*file*]

**DESCRIPTION**

Do a substitution (caesar) cipher for all shifts. a-z is the default, option **-n** selects 0-9, **-b** selects a-z1234567890.

You can use command line arguments for "*text*", or a filename, or standard input.

**OPTIONS**

- a** addletters output
- o opt** addletters *options*
- n** Select 0-9
- b** Select a-z1-90
- c** Circular rotation
- D lvl** Debug level

**EXAMPLES**

```
# the cache is here

$ geo-sub bpm kikpm qa pmzm
Shift  Substitution cipher (Caesar)
0:      bpm kikpm qa pmzm
1:      cqn ljlqn rb qnan
2:      dro mkmro sc robo
3:      esp nlmsp td spcp
4:      ftq omotq ue tqdq
5:      gur pnpur vf urer
6:      hvs qoqvs wg vsfs
7:      iwt rprwt xh wtgt
8:      jxu sqsxu yi xuhu
9:      kyv trtyv zj yviv
10:     lzw usuzw ak zwjw
11:     max vtvax bl axkx
12:     nby wuwby cm byly
13:     ocz xvxcz dn czmz
14:     pda ywyda eo dana
15:     qeb zxzeb fp ebob
16:     rfc ayafc gq fcpc
17:     sgd bzbgd hr gdqd
18:     the cache is here      <--- here it is
19:     uif dbdif jt ifsf
20:     vjg ecejg ku jgtg
21:     wkh fdfkh lv khuh
22:     xli gegli mw livi
23:     ymj hfhmj nx mjwj
24:     znk igink oy nkxk
25:     aol jhjol pz olyl
26:     bpm kikpm qa pmzm
```

# http://coord.info/GC2T23M

\$ geo-sub rsvxl jsvxc xmwjjâ° jtwyd.chur qpg irxu

Shift Substitution cipher (Caesar)

- 0: rsvxl jsvxc xmwjjâ° jtwyd.chur qpg irxu
- 1: stwym ktwyd ynxxkâ° kuxze.divs rqh jsyv
- 2: tuxzn luxze zoyllâ° lvyaf.ejwtsri ktzw
- 3: uvyao mvyaf apzmmâ° mwzbg.fkxutsj luax
- 4: vwzbp nwzbg bqannâ° nxach.glyvutk mvby
- 5: wxacq oxach crbooâ° oybdi.hmzwvul nwcz
- 6: xybdr pybdi dscppâ° pzcej.inaxwvm oxda
- 7: yzces qzcej etdqqâ° qadfk.jobyxwn pyeb
- 8: zadft radfk fuerrâ° rbegl.kpczyxo qzfc
- 9: abegu sbegl gvfssâ° scfhl.lqdayzp ragd
- 10: bcfhv tcfhm hwgttâ° tdgin.mrebazq sbhe
- 11: cdgiw udgin ixhuuâ° uehjo.nsfcbartcif
- 12: dehjx vehjo jyivvâ° vfikp.otgdcbs udjg
- 13: efiky wfikp kzjwwâ° wgjllq.puhe dct vekh
- 14: fgjllz xgjllq lakxxâ° xhkmr.qvifedu wfli
- 15: ghkma yhkmr mblyyâ° yilns.rwjgfev xgmj
- 16: hilnb zilns ncmzzâ° zjmot.sxkhgfw yhnk
- 17: ijmoc ajmot odnaaâ° aknpu.tylihgx ziol
- 18: jknpd bknpu peobbâ° bloqv.uzmjihy ajpm
- 19: kloqe cloqv qfpccâ° cmprw.vankjiz bkqn
- 20: lmpfr dmpwr rgqddâ° dnqxs.wbolkja clro
- 21: mnqsg enqsg shreeâ° eorty.xcpmlkb dmsp <--- part 2
- 22: north forty tisffâ° fpsuz.ydqnmlc entq <--- part 1
- 23: opsui gpsuz ujtggâ° gqtva.zero nmd four <--- part 3,5
- 24: pqtvj hqtva vkuhhâ° hruwb.afspone gpvs <--- part 4
- 25: qruwk iruwb wlviiâ° isvxc.bgtqpof hqwt
- 26: rsvxl jsvxc xmwjjâ° jtwyd.chur qpg irxu

# Circular

\$ echo NPTWL KUYBH DHA QSVHVWL | geo-sub -c

Left: NORTH FORTY TWO DEGREES

Right: NQVZP PAFJQ NSM DGKXMOE

**NAME**

**geo-suffix** - Replace name with name/TypeSizeDiffTerr/gcid/LatLon

**SYNOPSIS**

**geo-suffix** [*options*] [*file*]

**DESCRIPTION**

Process a "tabsep" format on stdin or "*file*" and produce a "tabsep" format on stdout. Replace name with name/TypeSizeDiffTerr/gcid/LatLon.

```
Type      Tr, Mu, Un, etc.
Size      Mi, Sm, Re, etc.
Diff      1, 1+, 2, 2+, etc.
Terr      1, 1+, 2, 2+, etc.
gcid      GC1H6YH equals 1H6YH
Lat       last 3 digits of latitude
Lon       last 3 digits of longitude
```

Also, print only "Events" if they are one day before or on the actual day.

For TomTom, nuvi, etc.

**EXAMPLES**

Convert to TomTom:

```
geo-suffix < example.ts |
  gpsbabel -i tabsep -f -
    -o tomtom -F /mnt/tomtom/USA_and_Canada/geocaching.ov2
```

Name change:

```
geo-suffix ~/xxx.ts | awk -F "      " '{ print $3 }'
```

```
OBG:NoMoreM/TrSm22+/1H6YH/132376
OBG:MonsterM/TrSm22/1HBZ5/100511
GreeniesandG/TrSm22/1HBZM/970265
OBG:LoveThe/TrRe1+1+/14XXG/932547
OBG:LovePoti/TrUn22/1HC0G/893424
OBG:Treasure/TrUn22/1HC1J/052037
MarkSpitzenH/TrRe22/1HC1T/968696
Psychedelia:N/TrUn1+1+/R70X/916066
Plato'sFiveG/UnRe2+2+/1H5EY/469495
```

**OPTIONS**

**-D lvl** Debug level

**NAME**

**geo-text2numbers** - grep for numbers in text

**SYNOPSIS**

**geo-text2numbers** [*options*] [*file*] ...

**DESCRIPTION**

grep for numbers in text using stdin or files.

**OPTIONS**

**-p pat** grep for 'pat' instead of numbers

**-D lvl** Debug level

**EXAMPLE**

Grep for `http://coord.info/GC51TNR`:

```
$ geo-text2numbers
```

```
If this event takes place we would be elated. If our favorite freighter  
sails again please phone us. it's even possible it would come this way.  
We would love to see her again.
```

```
seven
```

```
four
```

```
eight
```

```
one
```

```
seven
```

```
two
```

Grep using pattern:

```
$ geo-text2numbers -p '<em>.' GC6KWN9.html
```

```
<em>n
```

```
<em>t
```

```
<em>h
```

```
<em>r
```

```
<em>e
```

```
<em>e
```

```
<em>s
```

```
<em>e
```

```
<em>v
```

```
<em>e
```

```
<em>n
```



```

1 0 0 0 0 0 1 0 1 0 1 1 0 0 1 0 0 0 0 0 1
1 0 1 1 1 0 1 0 1 0 1 1 0 0 1 0 1 1 1 0 1
1 0 1 1 1 0 1 0 0 1 0 1 1 0 1 0 1 1 1 0 1
1 0 1 1 1 0 1 0 1 1 1 1 0 0 1 0 1 1 1 0 1
1 0 0 0 0 0 1 0 1 1 1 0 0 0 1 0 0 0 0 0 1
1 1 1 1 1 1 1 0 1 0 1 0 1 0 1 1 1 1 1 1 1
0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0
1 1 0 1 0 0 1 1 0 0 1 0 0 0 1 1 1 0 1 1 0
1 1 0 1 0 1 0 0 0 1 0 0 1 0 0 0 0 1 0 0 1
1 1 0 1 1 0 1 0 0 0 1 0 0 0 1 0 0 1 1 0 0
1 1 1 1 0 1 0 0 1 0 0 1 0 0 1 1 0 0 0 1 0
1 1 0 0 1 0 1 1 1 0 0 0 1 1 0 1 0 1 1 0 1
0 0 0 0 0 0 0 0 1 0 1 1 1 1 0 0 0 0 0 0 0
1 1 1 1 1 1 1 0 1 0 0 0 0 1 1 0 1 0 1 1 1
1 0 0 0 0 0 1 0 0 1 0 0 1 1 0 0 0 1 0 0 1
1 0 1 1 1 0 1 0 0 1 1 1 1 0 1 0 1 1 0 1 0
1 0 1 1 1 0 1 0 1 0 0 0 0 1 1 0 1 1 0 1 1
1 0 1 1 1 0 1 0 0 0 1 0 1 0 1 1 0 1 1 0 1
1 0 0 0 0 0 1 0 1 0 0 0 1 0 0 0 1 0 1 0 1
1 1 1 1 1 1 1 0 1 1 1 0 1 1 0 1 0 1 1 0 0

```

Convert GC4XPY4 and image to a PDF *file*:

```

$ geo-text2qrcode -P xxx.pdf GC4XPY4.txt
Then: http://online-barcode-reader.inliteresearch.com/

```

Convert GC4XPY4 and image to a PNG *file*:

```

$ geo-text2qrcode -p xxx.png GC4XPY4.txt
Then: http://zxing.org/w/decode.jspx
Or: http://online-barcode-reader.inliteresearch.com/

```

**SEE ALSO**

<http://online-barcode-reader.inliteresearch.com/> <http://zxing.org/w/decode.jspx>

**NAME**

**geo-thumbnails** - Recursively extract image thumbnails

**SYNOPSIS**

**geo-thumbnails** [*options*] *image* ...

**DESCRIPTION**

Recursively extract *image* thumbnails from the *image(s)*.

**OPTIONS**

**-D lvl** Debug level

**EXAMPLE**

For <http://coord.info/GC51JZN>:

```
$ geo-thumbnails n.jpg w.jpg
n-t-00.jpg n-t-01.jpg n-t-02.jpg n-t-03.jpg n-t-04.jpg n-t-05.jpg
w-t-00.jpg w-t-01.jpg w-t-02.jpg w-t-03.jpg w-t-04.jpg w-t-05.jpg
```

**SEE ALSO**

<http://www.sno.phy.queensu.ca/~phil/exiftool/>

**NAME**

**geo-timed-cache** - Timed cache password fetcher

**SYNOPSIS**

**geo-timed-cache** [*options*] *dir url*

**DESCRIPTION**

Timed cache password fetcher. If '*dir*' is a relative path, then add /home/rick/proj/caches/**geo-timed-cache**/ to '*dir*'. The URL will be fetched to 'password.NNNNN.<suffix>', where NNNNN is 00001, 00002, etc. and <suffix> is gif, jpg, png, html, or txt. Only the passwords which are different are saved.

**EXAMPLE**

In a crontab:

```
$ vi ~/.crontab/every5
geo-timed-cache Stay-Up-All-Night \
    http://techmanski.net/geocaching/GC5BZFR/image.php
```

Standalone:

```
$ geo-timed-cache -w 5 GC5PPGE http://gc5ppge.16mb.com
Sat Oct 7 15:35:32 CDT 2017:
    /home/rick/proj/caches/geo-timed-cache/GC5PPGE/password.00001.html
Sat Oct 7 15:40:33 CDT 2017:
    /home/rick/proj/caches/geo-timed-cache/GC5PPGE/password.00002.html
[control-c to terminate]
```

**OPTIONS****-w time**

Wait time in minutes before doing a 'snapshot'. [0] The program will never exit - use control-c to terminate it. Note: this option should be used outside of a crontab entry!

**-D lvl** Debug level

**SEE ALSO**

crontab (1)

**NAME**

**geo-triangulation** - Compute the "point" by measuring angles

**SYNOPSIS**

**geo-triangulation** [*options*] *lat0 lon0 bear0 lat1 lon1 bear1* [ *lat2 lon2 bear2* ]

**DESCRIPTION**

Compute the "point" by measuring angles from/to locations.

lat/lon can be specified in DegDec or dotted MinDec format.

The bearing is in degrees from the **location(s)** to the "point", or negative degrees to the **location(s)** from the "point". I.E. the **abs(bearing)** plus 180 mod 360. The bearing can be suffixed with mil or grad.

N.B. this program was inspired by Rock Johnson's "Gee" series of math caches. Dyl1231, Seabiskit, and I enjoy these very much. Thanks RJ!

**OPTIONS**

**-f** Pretend that the world is flat and 1 degree latitude == 1 degree longitude

**-D lvl** Debug level

**EXAMPLES**

MinDec input, from the locations to the "point" ...

```
$ geo-triangulation N41.09.810 W105.22.693 336.25 \  
                    N41.09.882 W105.22.868 61.74 \  
                    N41.10.101 W105.22.416 235.82  
0-1:   N41.09.926 W105.22.760  
0-2:   N41.09.925 W105.22.760  
1-2:   N41.09.931 W105.22.746  
Ave:   .927 .755
```

MinDec input, from the "point" to the locations ...

```
$ geo-triangulation N41.09.810 W105.22.693 -156.25 \  
                    N41.09.882 W105.22.868 -241.74 \  
                    N41.10.101 W105.22.416 -55.82  
0-1:   N41.09.926 W105.22.760  
0-2:   N41.09.925 W105.22.760  
1-2:   N41.09.931 W105.22.746  
Ave:   .927 .755
```

Two points ...

```
$ geo-triangulation N41.09.810 W105.22.693 -156.25 \  
                    N41.09.882 W105.22.868 -241.74  
0-1:   N41.09.926 W105.22.760
```

**NAME**

**geo-trilateration** - Compute the intersection of three circles

**SYNOPSIS**

**geo-trilateration** [*options*] *lat0 lon0 rad0 lat1 lon1 rad1 lat2 lon2 rad2*

**DESCRIPTION**

Compute the intersection of three circles on the earth.

lat/lon can be specified in DegDec or dotted MinDec format. radius is in meters (m), kilometers (km), feet (ft) or miles (mi).

N.B. this program was inspired by Rock Johnson's "Gee" series of math caches. Dyl1231, Seabiskit, and I enjoy these very much. Thanks RJ!

**OPTIONS**

- f** Pretend that the world is flat and 1 degree latitude == 1 degree longitude
- D lvl** Debug level

**EXAMPLES**

# DegDec input...

```
$ geo-trilateration 44.92342 -93.41253 382 \
    44.92335 -93.41165 398 \
    44.55.502 -93.24.795 205
p3a = 44.920119 -93.413749      n44.55.207 w93.24.825
p3b = 44.926875 -93.412695      n44.55.613 w93.24.762 <--
p3a = 44.926874 -93.412796      n44.55.612 w93.24.768 <--
p3b = 44.926326 -93.415098      n44.55.580 w93.24.906
p3a = 44.926875 -93.412745      n44.55.613 w93.24.765 <--
p3b = 44.925423 -93.415801      n44.55.525 w93.24.948
```

# MinDec input...

```
$ geo-trilateration 44.53.200 w93.36.000 370m \
    44.53.000 w93.36.200 262m \
    44.53.200 w93.36.200 453m
p3a = 44.885602 -93.604417      n44.53.136 w93.36.265
p3b = 44.883374 -93.600012      n44.53.002 w93.36.001 <--
p3a = 44.890036 -93.600031      n44.53.402 w93.36.002
p3b = 44.883374 -93.600025      n44.53.002 w93.36.002 <--
p3a = 44.883374 -93.600012      n44.53.002 w93.36.001 <--
p3b = 44.883339 -93.606647      n44.53.000 w93.36.399
```

# Flat World...

```
$ geo-trilateration -f \
    N 45 04.033 W 093 03.667 0.015611742375526 \
    N 45 03.491 W 093 04.787 0.00836557828246395 \
    N 45 04.655 W 093 04.569 0.0116429978957274
p3a = 45.065950 -93.076676      n45.03.957 w93.04.601 <--
p3b = 45.055799 -93.071764      n45.03.348 w93.04.306
p3a = 45.082210 -93.065466      n45.04.933 w93.03.928
p3b = 45.065952 -93.076676      n45.03.957 w93.04.601 <--
p3a = 45.065952 -93.076681      n45.03.957 w93.04.601 <--
p3b = 45.066548 -93.079864      n45.03.993 w93.04.792
```

**SEE ALSO**

<http://en.wikipedia.org/wiki/Trilateration>

**NAME**

**geo-ts2geko** - Convert tabsep to geko tabsep

**SYNOPSIS**

**geo-ts2geko** [*options*]

**DESCRIPTION**

Convert tabsep to geko tabsep. I.E. change the GID and remove events that are not within 2 days.

**OPTIONS**

**-D lvl** Debug level

**NAME**

**geo-uniq** - unique the tabsep database

**SYNOPSIS**

**geo-uniq** [*options*] [*file*]

**DESCRIPTION**

Unique the tabsep database based on GC codes. Last one wins. Used for update-caches in the incremental mode. This is for people who updated the coordinates for a cache.

**EXAMPLES**

Usage:

```
geo-uniq ~/Caches/rick.ts > rick.ts
mv rick.ts ~/Caches/rick.ts
```

**OPTIONS**

**-D lvl** Debug level

**NAME**

**geo-usernum** - Given a username, print the user account number

**SYNOPSIS**

**geo-usernum** [*options*] [*username*] ...

**DESCRIPTION**

Given a *username*, print the user account number. If no usernames are given on the comand line, then read usernames from stdin, one per line.

Requires: A free login at <http://www.geocaching.com>.

**curl** <http://curl.haxx.se/>

**OPTIONS****-a aliases**

Tab separated alias file [/home/rick/.geo-alias]

**-c** Remove cookie file when done

**-d dbfile**

Database file to cache lookups [/home/rick/.**geo-usernum**]

**-f** Force website lookup

**-u username**

Username for <http://www.geocaching.com>

**-p password**

Password for <http://www.geocaching.com>

**-s sleep** Time to sleep between page fetches [6]

**-v** Verbose. Print account, *username*, and URL

**-D lvl** Debug level [0]

**-U** Retrieve latest version of this script

Defaults can also be set with variables in file \$HOME/.georc:

```
PASSWORD=password;  USERNAME=username;
```

**EXAMPLE**

Decode the authors name:

```
$ geo-usernum -v rickrich
82192 rickrich http://www.geocaching.com/profile/?u=rickrich
```

**SEE ALSO**

geo-count, geo-found, geo-placed, <http://geo.rkkda.com/>

**NAME**

**geo-waypoint** - Convert a lat/lon into a waypoint using gpsbabel

**SYNOPSIS**

**geo-waypoint** [*options*] *latitude longitude name*

**DESCRIPTION**

Convert a *latitude/longitude* into a waypoint using gpsbabel. Lat/Lon may be in DegDec, MinDec, or DMS formats.

Acceptable formats for lat/lon are:

- 93.49130            DegDec (decimal degrees)
- W93.49130           DegDec (decimal degrees)
  
- "-93 29.478"        MinDec (decimal minutes)
- "W93 29.478"        MinDec (decimal minutes)
- 93.29.478           MinDec (decimal minutes)
- W93.29.478           MinDec (decimal minutes)
  
- "-93 45 30"         DMS (degrees, minutes, seconds)

**OPTIONS**

**-o format**

Output format, **-o?** for possibilities [gpsdrive] plus "gpsdrive.sql" for direct insertion into MySQL DB

**-S**     Alias for **-o** gpsdrive.sql

**-d**     For **-S**, just delete selected records

**-t type** The waypoint type [Geocache]

**-V gpsver**

Version of gpsdrive (2.09 or 2.10+) [2.09]

**-D lvl** Debug level [0]

**-U**     Retrieve latest version of this script

Defaults can also be set with variables in file \$HOME/.georc:

- NUM=num;                    OUTFMT=format;            BABELFLAGS=-s;
- SQLUSER=gast;                SQLPASS=gast;             SQLDB=geoinfo;

**EXAMPLES**

Enter a lat/lon into the GpsDrive 2.09 waypoint SQL database:

```
geo-waypoint -S "45 50.501" "-93 23.609" MultiCacheLeg2
```

Enter a lat/lon into the GpsDrive 2.11 waypoint SQL database:

```
geo-waypoint -V 2.11 -S "45 50.501" "-93 23.609" MultiCacheLeg2
```

**SEE ALSO**

geo-code, geo-pg, geo-nearest, <http://geo.rkkda.com/>

**NAME**

**geo-wordsearch** - Perform a Word Search

**SYNOPSIS**

**geo-wordsearch** [*options*] [*file*]

**DESCRIPTION**

Perform a Word Search on stdin or "*file*". You can do this two ways:

- The matrix followed by blank line, then the words.
- The matrix in 1 character words, then the words.

You can use letters or numbers. Comments are: "# any text".

**OPTIONS**

- b** No backwards
- d** No diagonals
- l** Lowercase words
- N** Append numerical words (NORTH, ONE, SIXTY, ...)
- r** Rotate-13
- u** Uppercase words
- w** Append 4+ letter words from /home/rick/lib/geo/english.dic
- W** Append 4+ letter words from /usr/share/dict/words
- D lvl** Debug level

**EXAMPLES**

First method:

```
$ geo-wordsearch
CRCOCRCCCVLV
ELRLELCCLOVER
LVOVOCOVOICE
VCOVRVRVRLRV
CLOVEREEEEVC
COERVVRVRLC
LVVCOOOVOLL
OEERLRLEVOLV
VRCCCCCEVCVC
ECLOVEREVOLC
REVOLCREVOLC
```

```
CLOV
VER
```

Second method:

```
$ geo-wordsearch
S Y C I S U M T P H C A N N E K S G L
F N A G S I X E A N H N O O M R T H N
F I O D O R T Y R T A W O I I D E E G
T C R I I R E E A S M T H S T I W R T
S N E E T L Y F D I P V E E R Y P O I
D E E L W A O N E T A O N M E K I S S
```

N C P V E O R H S E G S E A H V E N S  
 E N F A E B R O S E N V R K T E N R W  
 I A E E R S R K C G E S S E A T E E I  
 R D S I G T R A S E E H T R F Z Y T T  
 F W T O D E Y A T V D O G S I R T E E  
 S S I N G I N G E E F I C T F E T Y F  
 O U V S E I B A B Y R P E A F O S I B  
 N T I F A M I L Y O W P N N C T E A F  
 O S T R E A M E R S P E O U A H N R N  
 I N I S N O O L L A B C N H E N I Y K  
 R Y E Y A D S R A E Y W E N E H C N X  
 P J S N O I T U L O S E R R C K X H G  
 B V T H G I N D I M E F S H O R N S C  
 APPETIZERS BABIES BALLOONS BANNERS CELEBRATE  
 CHAMPAGNE CONFETTI DANCE DECORATIONS FAMILY  
 FATHERTIME FESTIVITIES FIREWORKS FRIENDS GEOCACHING  
 GSIX HATS HOLIDAY HORNS KISS  
 MIDNIGHT MUSIC NEWYEARS DAY NEWYEARS EVE NEWYEARS EVENT  
 NOISEMAKERS PARADES PARTY RESOLUTIONS SINGING  
 STREAMERS

Original:

S Y C I S U M T P H C A N N E K S G L  
 F N A G S I X E A N H N O O M R T H N  
 F I O D O R T Y R T A W O I I D E E G  
 T C R I I R E E A S M T H S T I W R T  
 S N E E T L Y F D I P V E E R Y P O I  
 D E E L W A O N E T A O N M E K I S S  
 N C P V E O R H S E G S E A H V E N S  
 E N F A E B R O S E N V R K T E N R W  
 I A E E R S R K C G E S S E A T E E I  
 R D S I G T R A S E E H T R F Z Y T T  
 F W T O D E Y A T V D O G S I R T E E  
 S S I N G I N G E E F I C T F E T Y F  
 O U V S E I B A B Y R P E A F O S I B  
 N T I F A M I L Y O W P N N C T E A F  
 O S T R E A M E R S P E O U A H N R N  
 I N I S N O O L L A B C N H E N I Y K  
 R Y E Y A D S R A E Y W E N E H C N X  
 P J S N O I T U L O S E R R C K X H G  
 B V T H G I N D I M E F S H O R N S C

Words Found:

S Y C I S U M P C N E  
 F N A G S I X A H O M N  
 I O D R A I I E  
 T C R I I A M S T W  
 S N E E T L D P E R Y  
 D E E L W A O E A M E K I S S  
 N C P V E O R H S G A H S  
 E N F A E B R O N R K T R  
 I A E R S R K C G E S E A E I  
 R D S T R A S E E R F Z T  
 F T Y A T V D O S I T  
 S I N G I N G E E C T E  
 V S E I B A B Y E A F S B

I F A M I L Y   W P   N C T   A  
 S T R E A M E R S P E O   A H N  
 I S N O O L L A B C N H   N I  
 E Y A D S R A E Y W E N E   N  
 S N O I T U L O S E R R   G  
 T H G I N D I M   S H O R N S

Words Left:

                          T   H   A N       K S G L  
                           E   N   N O       R T H  
 F           O R T Y   T   W O       D E   G  
                   R E E   S   T H       I   R T  
                   Y F   I   V E       P O I  
                   N   T   O N  
                           E   S E       V E N  
                   S E   V       E N   W  
                   E                   S       T   E  
                   I G                   H T       Y   T  
 W   O D E                   G       R   E E  
 S                    F I       F   T Y F  
 O U                   R P       O   I  
 N T                   O       N       E   F  
 O                                   U       R N  
 I N                                   E       Y K  
 R Y                                   H C   X  
 P J                                   C K X H  
 B V                   E F                   C

Horizontal:

THANKSGLENNORTHFORTYTWODEGREESTHIRTYFIVEPOINTONESEVENSEVENWESTEIG  
 HTYTWODEGREESFIFTYFOURPOINTONEFOURNINEYKRYHCXPJCKXHBVEFC

Vertical:

FSONOIRPBWUTNYJVEIOOGDRRETEYTEYEFNSHNTSITEEOFREANWTVOSVHIPFNOOHE  
 NESTGNUMFECKRDIVETROHKSTEPENYTECXGHRONEEYIRYHLGTIWTEFFNKXC

Overlap:

Horizontal:       EITSVESRAEYAEWENAE  
 Vertical:        EEVAESREAIEYWTEANS

Words: 32 out of 31

- 1 APPETIZERS
- 1 BABIES
- 1 BALLOONS
- 1 BANNERS
- 1 CELEBRATE
- 1 CHAMPAGNE
- 1 CONFETTI
- 1 DANCE
- 1 DECORATIONS
- 1 FAMILY
- 1 FATHERTIME
- 1 FESTIVITIES
- 1 FIREWORKS
- 1 FRIENDS
- 1 GEOCACHING
- 1 GSIX
- 1 HATS
- 1 HOLIDAY

- 1 HORNS
- 1 KISS
- 1 MIDNIGHT
- 1 MUSIC
- 1 NEWYEARS DAY
- 2 NEWYEARSEVE
- 1 NEWYEARSEVENT
- 1 NOISEMAKERS
- 1 PARADES
- 1 PARTY
- 1 RESOLUTIONS
- 1 SINGING
- 1 STREAMERS

**SEE ALSO**

<http://www.newocr.com/>  
<http://coord.info/GC2BBF0>

<http://coord.info/GC38HJE>

<http://coord.info/GC35J4T>

**NAME**

**geo-zipcode** - Translate zip code to city and state

**SYNOPSIS**

**geo-zipcode** [*options*] *zip* ...

**DESCRIPTION**

Translate *zip* code to city and state.

**OPTIONS**

**-D lvl** Debug level

**EXAMPLE**

Convert cities:

```
$ geo-zipcode 05345 50212 84763 67485 15639
05345   Newfane, VT 05345, USA
50212   Ogden, IA 50212, USA
84763   Rockville, UT 84763, USA
67485   Tipton, KS 67485, USA
15639   Hunker, PA 15639, USA
```

e.g. first letter is "NORTH".

**NAME**

**ggl2ll** - google QRST string to Lat/lon

**SYNOPSIS**

**ggl2ll** [*options*] *qrst* ...

**DESCRIPTION**

Google QRST string to Lat/lon.

**OPTIONS**

**-D lvl** Debug level

**EXAMPLE**

Convert <http://coord.info/GC4RXYK> :

```
$ ggl2ll tqsqtstssqssrsrstsrttrtqsrqrr  
tqsqtstssqssrsrstsrttrtqsrqrr 41.591999 -73.834784 n41.35.520 w73.50.087
```

**NAME**

**gpx2html** - GPX to HTML converter

**SYNOPSIS**

**gpx2html** [*options*] [<gpx-file> ...]

**DESCRIPTION**

**gpx2html** is an application that generates simplified and compact HTML pages from GPX files obtained from Pocket Queries at geocaching.com.

**OPTIONS**

**-a**      Use old style index\_names.html

**-h|-?**    Help

**SEE ALSO**

<http://www.fizzymagic.net/Geocaching/gpx2html/gpx2html.html>

**NAME**

**gpx-finders** - Output the finders from a GPX file

**SYNOPSIS**

**gpx-finders** [*options*]

**DESCRIPTION**

**gpx-finders** [*options*]

Output the finders from a GPX file.

Options:

-D lvl          Debug level

**gpx-finders -H** [*options*]

Add headers to stdin

Options:

-D lvl          Debug level

**EXAMPLES**

Finder count:

```
$ gpx-finders /home/rick/proj/caches/Backups/mn30.gpx | wc -l
4480
```

Finders:

```
$ gpx-finders mn30.gpx | sort -n -t'          ' -k2 | gpx-finders -H
```

**NAME**

**gpx-fff** - Display FTF cache logs for a finder from a GPX file

**SYNOPSIS**

**gpx-fff** [*options*] *finder-name* *gpx-file*

**USAGE**

**-D level**

Debug level

**EXAMPLES**

**NAME**

**gpx-loghistory** - print all logs from a GPX file in reverse cron order

**SYNOPSIS**

**gpx-loghistory** [*options*] file.gpx ...

**USAGE**

- f fspec** Output into page/day format. %d format specifier needed.
- F** Output Found logs only.
- H** Output HTML **page(s)**.
- n num** Stop after "num" logs. [no limit] With **-f**, stop after "num" days
- u file** Save finders to file
- D level**  
Debug level [0]

**NAME**

**gpx-logs** - Display cache logs for a finder from a GPX file

**SYNOPSIS**

**gpx-logs** [*options*] *finder-name* *gpx-file*

**USAGE**

**-s pattern**

Print only logs with "pattern"

**-D level**

Debug level

**EXAMPLES**

**NAME**

**gpx-merge** - GPX file merge

**SYNOPSIS**

`gpx-merge [options] file(s) ...`

**DESCRIPTION**

GPX file merge. Output to stdout.

**OPTIONS**

**-D lvl** Debug level

**NAME**

**gpx-photos** - Fetch hi-res PNG aerial photos from a GPX file

**SYNOPSIS**

**gpx-photos** [*options*] *gpx-file*

**DESCRIPTION**

Fetch hi-res PNG format aerial photos for every cache in a GPX file.

**OPTIONS**

- f** Force image download even if it already exists
- W width**  
Width of image in pixels [500]
- H height**  
Height of image in pixels [500]
- a mapsrc**  
Source for photos, ala geo-map [gstatic-aerial]
- s scale** Scale of photos, ala geo-map {0.5fpp}
- S time** Time to sleep between fetches [0]
- D lvl** Debug level

**NAME**

**gpx-stats** - Compute stats from a GPX file

**SYNOPSIS**

**gpx-stats** [*options*]

**DESCRIPTION**

**gpx-stats** [*options*]

Compute stats from a GPX file

Options:

-l	Sort by # of logs
-a	Sort by cache age
-f	Sort by log frequency
-D lvl	Debug level

**gpx-stats -H** [*options*]

Add headers to stdin

Options:

-l	Sort by # of logs
-a	Sort by cache age
-f	Sort by log frequency
-t	Top caches only
-D lvl	Debug level

**EXAMPLES**

Statistics:

```
gpx-stats all-mn.gpx
```

```
gpx-stats -l all-mn.gpx | gpx-stats -H
```

```
gpsbabel -igpx -fall-mn.gpx -x radius,lat=45,lon=-93.5,distance=20 \  
-ogpx -Ftc.gpx
```

```
gpx-stats -l tc.gpx | gpx-stats -H -t
```

**NAME**

**gpx-unfound** - Filter a GPX file removing found caches

**SYNOPSIS**

```
gpx-unfound [options] finder-name < in.gpx > out.gpx
```

**DESCRIPTION**

Filter a GPX file removing found caches.

Requires:

```
curl    http://curl.haxx.se/
```

**OPTIONS**

**-D lvl** Debug level

**EXAMPLES**

Filter caches unfound by "rickrich":

```
geo-unfound rickrich <~/Caches/mn.gpx > xxx
```

**SEE ALSO**

geo-newest, geo-nearest, geo-found, geo-placed, geo-code, geo-map, geo-waypoint, <http://geo.rkkda.com/>

**NAME****lethist** - Letter histogram**SYNOPSIS****lethist** [*options*] [*words*] ...**DESCRIPTION**Letter histogram from <stdin> or from '*words*'.**EXAMPLE**

Letter histogram:

```
$ lethist | sort -k2 -n -r
1 5 - 8 ) ) W 5 - ( + ) ) ; 4 8 W 5 ; 8 ( * + ; 8 ; W + 0 5 ( 3 8 9
? 0 ; 6 ; ( ? * K 8 ! ; ( 8 8 ) W 6 ; 4 5 1 5 0 0 8 * + * 8 6 * 2 8
; W 8 8 * ; 4 8 ! 8 5 ; 4 ) 4 8 5 ! 6 ) 6 * ; 4 8 9 6 ! ! 0 8 + 1 ;
4 8 ; + . 0 6 9 2 + 1 ; 4 8 1 5 0 0 8 * ; ( 8 8 3 + ; 4 8 ( 8 1 ( +
9 ; 4 8 ! 8 5 ; 4 ) 4 8 5 ! ) 4 + + ; 5 2 8 8 0 6 * 8 ; 4 6 ( ; : 1
8 8 ; + ? ; ; + ; 4 8 ) + ? ; 4.
```

```
$ lethist "Cottonwood trees are, perhaps, the best shade trees"
```

**OPTIONS**

**-t**      Print total  
**-D lvl**   Set Debug level [0]

**SEE ALSO****addletters(1)**

**NAME**

**ll2geohash** - Lat/long to geohash

**SYNOPSIS**

**ll2geohash** [*options*] *lat lon ...*

**DESCRIPTION**

Lat/long to geohash.

<http://en.wikipedia.org/wiki/Geohash>

**EXAMPLE**

Convert:

```
$ ll2geohash n35.44.000 w79.28.832
dnr7r3h1c254
```

**OPTIONS**

**-D lvl** Debug level

**NAME**

**ll2ggl** - Lat/lon to google maps

**SYNOPSIS**

**ll2ggl** [*options*] *lat lon*

**DESCRIPTION**

Lat/*lon* to google maps.

**OPTIONS****-z zoom**

Zoom factor from 0 (small) to 17 (large)[2]

**-D lvl** Debug level**EXAMPLE**

Convert *lat/lon*:

```
$ ll2ggl 53.558614 -113.552123
tqtrrrrsrtrttttqssrsrqr
```

**SEE ALSO**

<http://intepid.com/posts/484>

**NAME**

**ll2maidenhead** - Lat/long to Maidenhead (Grid Squares)

**SYNOPSIS**

**ll2maidenhead** [*options*] *lat lon ...*

**DESCRIPTION**

Lat/long to Maidenhead Locator System a.k.a. Grid Squares.

**EXAMPLES**

# DegDec input...

```
$ ll2maidenhead 7.47194 47.22470
LJ37OL
```

# MinDec input...

```
$ ll2maidenhead n45.00.000 w93.30.000
EN35GA
```

# Copied from gc.com...

```
$ ll2maidenhead N 44° 59.989 W 093° 22.881
EN34HX
```

# Batch ...

```
$ cat <<EOF |
> 40.806862      -96.681679
> 39.7391536    -104.9847034
> 33.5206608    -86.80249
> 39.114053     -94.6274636
> 32.802955     -96.769923
> 41.0814447    -81.5190053
> 46.1381676    -122.9381672
> 43.0730517    -89.4012302
> EOF
> while read lat lon; do
>     ll2maidenhead $lat $lon
> done
EN10PT
DM79MR
EM63OM
EM29QC
EM12OT
EN91FB
CN86MD
EN53HB
```

**OPTIONS**

**-D lvl** Debug level

**NAME**

**ll2osg** - Lat/long to British National Grid

**SYNOPSIS**

**ll2osg** [*options*] *lat lon ...*

**DESCRIPTION**

Lat/long to British National Grid, a.k.a. Ordnance Survey Grid.

[http://www.dorcus.co.uk/carabus/ll\\_ngr.html](http://www.dorcus.co.uk/carabus/ll_ngr.html)

**EXAMPLE**

Convert from DegDec:

```
$ ll2osg 53.8826137384 -2.9290438893
338937 443358 SD 38937 43358
```

Convert from MinDec:

```
$ ll2osg N53.52.957 W2.55.743
338936 443358 SD 38936 43358
```

**OPTIONS**

**-D lvl** Debug level

**NAME**

**ll2rd** - Lat/Lon to RD (Dutch)

**SYNOPSIS**

**ll2rd** [*options*] *lat lon ...*

**DESCRIPTION**

Lat/Lon to RD (Dutch).

<http://www.dekoepel.nl/pdf/Transformatieformules.pdf>

**EXAMPLE**

Convert from DegDec:

```
$ ll2rd n52.01234 e5.01234
129264 447175
```

Convert from MinDec:

```
$ ll2rd n52.01.201 e4.23.057
86160 448438
```

**OPTIONS**

**-D lvl** Debug level

**NAME**

**ll2usng** - Lat/long to US National Grid

**SYNOPSIS**

**ll2usng** [*options*] *lat lon ...*

**DESCRIPTION**

Lat/long to US National Grid.

Also known as "Military Grid Reference System" (MGRS).

<http://dhost.info/usngweb/>

**EXAMPLE**

Convert:

```
$ ll2usng 44 -93.5
15T VJ 59913 71994
```

**OPTIONS**

**-D lvl** Debug level

**NAME****maidenhead2ll** - Maidenhead (Grid Squares) to Lat/long**SYNOPSIS****maidenhead2ll** [*options*] *grid* ...**DESCRIPTION**

Maidenhead (a.k.a. Grid Squares) to Lat/long

**EXAMPLES**

Convert:

```
$ maidenhead2ll EN10PT DM79MR
EN10PT  40.812500 -96.708330   n40.48.750 w96.42.500
DM79MR  39.729170 -104.958330  n39.43.750 w104.57.500
```

Convert, full precidsion:

```
$ maidenhead2ll -f EN10PT
EN10PT  40.812500 -96.708330   n40.48.750 w96.42.500
```

Convert, full precision:

```
$ maidenhead2ll en35lb60ub29
EN35LB60UB29  45.041997 -93.026319   n45.02.520 w93.01.579
```

**OPTIONS**

**-f** Full precision  
**-D lvl** Debug level

**NAME**

**mayan-long-count** - mayan long count

**SYNOPSIS**

**mayan-long-count** [*options*] [*integer*]...

**DESCRIPTION**

Perform a Mayan long count on stdin or the listed **integer**(s). Output both a vigesimal (base-20) and a long count (base 20 and base 18) date.

**OPTIONS**

**-D lvl** Debug level

**EXAMPLE**

From the command line:

```
$ mayan-long-count 6 6 6 6
50526 = 6*8000 + 6*400 + 6*20 + 6*1
45486 = 6*7200 + 6*360 + 6*20 + 6*1
```

**NAME**

**mngca** - Fetch MnGCA cache counts and upload to website

**SYNOPSIS**

**mngca** [*options*]

**DESCRIPTION**

Fetch MnGCA cache counts and upload to website.

Run this on Mondays and Fridays in the wee hours

**OPTIONS**

- i** Just recreate the web pages from existing data
- m l,l** Override moving123 lat/lon
- D lvl** Debug level

**NAME**

**mngca-logs** - Create a webpage of recent logs

**SYNOPSIS**

**mngca-logs** [*options*] [*gpx-files*]

**DESCRIPTION**

Create a webpage of recent logs.

This is meant to be called from cron every 5 minutes.

**OPTIONS**

- f** Force regeneration of web page
- v** View-only, do not post the pages
- D lvl** Debug level

**NAME**

**mngca-newmap** - Create a map of newest caches

**SYNOPSIS**

**mngca-newmap** [*options*]

**DESCRIPTION**

Create a map of newest caches for the MnGCA.

**OPTIONS**

- a0** Use tiger for the maps and use tiger to place the markers on the map.
- a num** Use geo-map and map source "num" for the maps. [3]
- g** Do not include geocaching.com caches
- n** Do not include navicache.com caches
- o** Do not include opencaching.com caches
- k** Do not include opencaching.us caches
- v** View-only, do not post the maps
- T dir** Temp directory name for results [tmp/mngca]
- D lvl** Debug level

**NAME**

**navaho-code-talkers** - Translate Navaho into English

**SYNOPSIS**

**navaho-code-talkers** [*options*] [*words*] ...

**DESCRIPTION**

Translate Navaho into English. Read from the command line or stdin.

**OPTIONS**

**-s** Short version and don't read dashes from DB.

**-D lvl** Debug level

**EXAMPLE**

From the command line:

```
$ navaho-code-talkers A-KHA TSAH GAH A-KHA MOASI KLIZZIE-YAZZIE
O      A-KHA    OIL
N      TSAH    NEEDLE
R      GAH     RABBIT
O      A-KHA    OIL
C      MOASI   CAT
K      KLIZZIE-YAZZIE  KID
```

From stdin:

```
$ navaho-code-talkers
  ATSAH-BE-YAZ WOZ-CHEIND TAH-CHILL NAHL-KIHD
  ATSAH-BE-YAZ BE-NE-TA-TSOSIE DA-AHL-ZHIN TAH-TSOSIE
  GHAW-JIH ATSAH-BE-YAZ
JANUARY ATSAH-BE-YAZ    SMALL EAGLE
FEBRUARY          WOZ-CHEIND      SQUEEKY VOICE
MARCH    TAH-CHILL      SMALL PLANT
DEGREE   NAHL-KIHD      DEGREE
JANUARY  ATSAH-BE-YAZ    SMALL EAGLE
JULY     BE-NE-TA-TSOSIE SMALL HARVEST
PERIOD   DA-AHL-ZHIN    PERIOD
MAY      TAH-TSOSIE     SMALL PLANT
SEPTEMBER          GHAW-JIH      HALF
JANUARY  ATSAH-BE-YAZ    SMALL EAGLE
```

Short version:

```
$ navaho-code-talkers -s
  CHUO YEHVES DAHNESTSA KLESH AWOH BETKAH BEHBHKEASCHINIGH BETKAH
  YEHVES ACHIN BETKAH ACHIN NEAHSJAH DAHNESTSA DAH LIN BETKAH SEIS
  BETKAH MAE TLOCHIN SHIDA DAHNESTSA
FIRST NUMBER IN NORTH IS FOUR
```

```
$ navaho-code-talkers -s < GC6CHA4.txt |
  navaho-code-talkers -s | navaho-code-talkers -s
FIRST NUMBER IN NORTH IS FOUR
THE LAST NUMBER IN WEST IS ZERO
```

**SEE ALSO**

[http://www.geocaching.com/geocache/GCHGFK\\_ah-na-sozi](http://www.geocaching.com/geocache/GCHGFK_ah-na-sozi)

[https://www.geocaching.com/geocache/GC6CHA4\\_insane-hides-2](https://www.geocaching.com/geocache/GC6CHA4_insane-hides-2)

<http://www.history.navy.mil/faqs/faq61-4.htm>

[https://en.wikipedia.org/wiki/Code\\_talker](https://en.wikipedia.org/wiki/Code_talker)

**NAME**

**nc-nearest** - Fetch a list of nearest geocaches

**SYNOPSIS**

**nc-nearest** [*options*]

**nc-nearest** [*options*] [*lat*] [*lon*]

**DESCRIPTION**

Fetch a list of nearest geocaches from navicache.com.

Requires:

**curl** <http://curl.haxx.se/>

Options:

- f** Do not report any found or unavailable caches
- F** Report caches found by the login 'username' as unfound
- T datespec**  
Return caches placed or modified since 'datespec', which can be any date accepted by the **date(1)** command.
- n num** Return "num" caches [20]
- s** Output short names for the caches (gpsbabel option)
- I term** Include only caches with 'term' [\*]
- X term**  
Exclude caches with 'term' [**-unavail**]. Terms: unfound, ifound, unavail, regular, multi, virtual, webcam, event, hybrid, moving
- r radius**  
Display only caches with radius (e.g. **-r 25M**)
- u username**  
Username for <http://www.navicache.com>
- p password**  
Password for <http://www.navicache.com>
- o format**  
Output format, **-o?** for possibilities [gpsdrive]. Plus "gpsdrive.sql" for direct insertion into MySQL DB. Plus "map[,geo-map-opts]" to display a geo-map.
- O filename**  
Output file, if not stdout
- S** Alias for **-o gpsdrive.sql**
- d** For **-S**, just delete selected records
- P** For **-S**, purge all records of type **-t Geocache\***
- t type** For **-ogpsdrive.sql**, the waypoint type [Geocache]
- H htmdir**  
Also fetch the printable HTML pages (slowly)
- L logdir**  
Also fetch the plain text log entries (slowly)
- D lvl** Debug level [0]
- U** Retrieve latest version of this script

Defaults can also be set with variables in file \$HOME/.georc:

```
PASSWORD=password;  USERNAME=username;
LAT=latitude;      LON=logitude;
NUM=num;           OUTFMT=format;      BABELFLAGS=-s;
SQLUSER=gast;      SQLPASS=gast;          SQLDB=geoinfo;
```

## EXAMPLES

Add nearest 50 caches to a GpsDrive SQL database

```
nc-nearest -n50 -f -s -S MN
```

Purge the existing SQL database of all geocaches, and fetch 200 fresh ones...

```
nc-nearest -S -P -s -n200 MN
```

## SEE ALSO

geo-newest, geo-nearest, geo-found, geo-placed, geo-code, geo-map, geo-waypoint, nc-newest,  
<http://geo.rkkda.com/>

**NAME**

**nc-newest** - Fetch a list of newest geocaches

**SYNOPSIS**

**nc-newest** [*options*] [*state*]

**nc-newest** [*options*] [*state*] [*lat*] [*lon*]

**DESCRIPTION**

Fetch a list of newest geocaches.

Requires:

**curl** <http://curl.haxx.se/>

Options:

- f** Do not report any found or unavailable caches
- F** Report caches found by the login 'username' as unfound
- T datespec**  
Return caches placed or modified since 'datespec', which can be any date accepted by the **date(1)** command.
- n num** Return "num" caches [20]
- s** Output short names for the caches (gpsbabel option)
- I term** Include only caches with 'term' [\*]
- X term**  
Exclude caches with 'term' [**-unavail**]. Terms: unfound, ifound, unavail, regular, multi, virtual, webcam, event, hybrid, moving
- r radius**  
Display only caches with radius (e.g. **-r 25M**)
- u username**  
Username for <http://www.navicache.com>
- p password**  
Password for <http://www.navicache.com>
- o format**  
Output format, **-o?** for possibilities [gpsdrive]. Plus "gpsdrive.sql" for direct insertion into MySQL DB. Plus "map[,geo-map-opts]" to display a geo-map.
- O filename**  
Output file, if not stdout
- S** Alias for **-o gpsdrive.sql**
- d** For **-S**, just delete selected records
- P** For **-S**, purge all records of type **-t Geocache\***
- t type** For **-ogpsdrive.sql**, the waypoint type [Geocache]
- H htmdir**  
Also fetch the printable HTML pages (slowly)
- L logdir**  
Also fetch the plain text log entries (slowly)
- D lvl** Debug level [0]
- U** Retrieve latest version of this script

Defaults can also be set with variables in file `$HOME/.georc`:

```
PASSWORD=password;  USERNAME=username;
LAT=latitude;       LON=logitude;
NUM=num;            OUTFMT=format;      BABELFLAGS=-s;
SQLUSER=gast;       SQLPASS=gast;          SQLDB=geoinfo;
```

## EXAMPLES

Add newest 50 caches to a GpsDrive SQL database

```
nc-newest -n50 -f -s -S MN
```

Purge the existing SQL database of all geocaches, and fetch 200 fresh ones...

```
nc-newest -S -P -s -n200 MN
```

## SEE ALSO

geo-newest, geo-nearest, geo-found, geo-placed, geo-code, geo-map, geo-waypoint, <http://geo.rkkda.com/>

**NAME**

**negadecimal** - Convert to/from negadecimal, negabinary, ...

**SYNOPSIS**

**negadecimal** [*options*] *num* ...

**DESCRIPTION**

Convert to/from **negadecimal**, negabinary, negaternary... all negative bases from 2 to 10.

**OPTIONS**

**-2,-3,...,-9**      negabinary, negaternary, ...  
**-d**            To/From in decimal  
**-n**            From **negadecimal**  
**-D lvl**        Debug level

**EXAMPLE**

To **negadecimal**:

```
$ negadecimal 864
944
```

From **negadecimal**:

```
$ negadecimal -n 944 5948692 8769481
864 4132512 7351321
```

From base **-4** to decimal:

```
$ negadecimal -4 -n -d 12113103010 1301302133022
584508 4136506
```

Otherway:

```
$ negadecimal -4 -d 584508 4136506
12113103010 1301302133022
```

**SEE ALSO**

<http://math2.org/math/general/numnotation.htm>

**NAME**

**nono2cross+a** - Convert .nono to monochrome cross+a format

**SYNOPSIS**

`nono2cross+a` [*options*]

**DESCRIPTION**

Convert .nono to monochrome cross+a format.

**OPTIONS**

**-D lvl** Debug level

**EXAMPLE**

Convert GC5MYCE.nono to cross+a format:

```
$ nono2cross+a < ~/proj/caches/GC5MYCE.nono > xxx.jcp
```

**SEE ALSO**

geo-nonogram, nono2jsolver, nono2teal

<http://www.cross-plus-a.com/>

<http://webpbn.com/export.cgi>

**NAME**

**nono2jsolver** - Convert .nono to jsolver format

**SYNOPSIS**

**nono2jsolver** [*options*]

**DESCRIPTION**

Convert .nono to monochrome cross+a format.

**OPTIONS**

**-D lvl** Debug level

**EXAMPLE**

Convert GC5MYCE.nono to jsolver format:

```
$ nono2jsolver < ~/proj/caches/GC5MYCE.nono > xxx.jsolver
```

**SEE ALSO**

geo-nonogram, nono2cross+a, nono2teal

<https://sourceforge.net/projects/jsolver/>

<http://webpbn.com/export.cgi>

**NAME**

**nono2teal** - Convert .nono to teal format

**SYNOPSIS**

**nono2teal** [*options*]

**DESCRIPTION**

Convert .nono to teal format.

**OPTIONS**

**-D lvl** Debug level

**EXAMPLE**

Convert GC7CJ4W-w.nono to a teal format:

```
$ nono2teal < ~/proj/caches/GC7CJ4W-w.nono
{"ver":[[[2,2,1,1,3,3,1,2,2,1,1,1],[1,1,1,1,1,1,1,1,1,1,1,1],
[2,1,1,1,1,1,1,1,1,1,1,1],[1,1,1,1,1,1,1,1,1,1,1,5],
[2,2,3,1,1,3,1,1,3,1,1],[4,3,3,3,2,3],[1,1,1,1,1,1,1,1,1,1],
[1,2,1,1,3,3,2,2],[1,1,1,1,1,1,1,1,1,1,1],
[2,3,2,2,2,2,3,3,1,1,4,2,3],
[1,1,1,1,1,1,1,1,1,1,2,1,1,1,1,1],
[2,3,2,2,1,1,2,1,1,4,1,2,2,2],
[1,1,1,1,1,1,1,1,1,1,2,1,1,1,1,1],
[2,1,2,2,2,2,1,3,1,1,4,2,1,1]],
"hor":[[[3,1,3,1],[1,3,1,3],[5],[6,6],[1,2,1,2,1],[1,6],[5],
[11],[6,2,1,1],[5],[1,5],[11,1,1],[1,1,1,1],[10],[1,1,1],
[10,3],[1,1,1,1],[8,1],[5,1,3],[6],[2,1,2],[5],[11],
[1,1,2,1],[2,3,1],[5,5],[1,1,1,1], [0],[5,5],[1,2],[1,2],
[5],[4],[2,5],[2,1,1],[2,1,1,1],[4,1,3],[0],[5],
[1,1,1],[0],[5],[1],[2,2]]]}
```

**SEE ALSO**

geo-nonogram, nono2cross+a, nono2jsolver

<http://a.teall.info/nonogram/>

<http://webpbn.com/export.cgi>

**NAME**

**oc-nearest** - Fetch a list of nearest geocaches from opencaching.com

**SYNOPSIS**

**oc-nearest** [*options*]

**oc-nearest** [*options*] *lat lon*

**DESCRIPTION**

Fetch a list of nearest geocaches from opencaching.com.

Requires:

**curl** <http://curl.haxx.se/>

Options:

**-c** Report (include) cross-listed caches.

**-f** Do not report any found or unavailable caches.

**-F** Report caches found by the login 'username' as unfound.

**-T datespec**

Return caches placed or modified since 'datespec', which can be any date accepted by the **date(1)** command.

**-n num** Return "num" caches [20]

**-s** Output short names for the caches (gpsbabel option)

**-I term** Include only caches with 'term' [\*]

**-X term**

Exclude caches with 'term' [**-unavail**]. Terms: unfound, ifound, unavail, regular, multi, virtual, webcam, event, hybrid, moving

**-r radius**

Display only caches with radius (e.g. **-r 25M**)

**-M mystery**

Use file 'mystery' for unknown/mystery/puzzle caches [/home/rick/.oc-mystery]. Awk Format: *gcid lat lon comment* i.e: OCXCBVB n44.45.123 w93.00.321 Final

**OCXCC1Z 44.123456 -93.564123**

Cache

**-o format**

Output format, **-o?** for possibilities [gpsdrive]. Plus "gpsdrive.sql" for direct insertion into MySQL DB. Plus "map[,geo-map-opts]" to display a geo-map.

**-O filename**

Output file, if not stdout

**-S** Alias for **-o gpsdrive.sql**

**-d** For **-S**, just delete selected records

**-P** For **-S**, purge all records of type **-t Geocache\***

**-t type** For **-ogpsdrive.sql**, the waypoint type [Geocache]

**-H htmdir**

Also fetch the printable HTML pages (slowly)

**-L logdir**

Also fetch the plain text log entries (slowly)

- D lvl** Debug level [0]
- U** Retrieve latest version of this script

Defaults can also be set with variables in file \$HOME/.georc:

```
LAT=latitude;          LON=logitude;
NUM=num;               OUTFMT=format;       BABELFLAGS=-s;
SQLUSER=gast;         SQLPASS=gast;         SQLDB=geoinfo;
```

## EXAMPLES

Add nearest 50 caches to a GpsDrive SQL database

```
oc-nearest -n50 -f -s -S
```

Purge the existing SQL database of all geocaches, and fetch 200 fresh ones...

```
oc-nearest -S -P -s -n200
```

Include cross-listed (i.e. gc.com) caches

```
oc-nearest -c -s
```

## SEE ALSO

geo-newest, geo-nearest, geo-found, geo-placed, geo-code, geo-map, geo-waypoint, oc-newest,  
<http://geo.rkkda.com/>

**NAME**

**oc-newest** - Fetch a list of newest geocaches from opencaching.com

**SYNOPSIS**

**oc-newest** [*options*] [*country*] [*state*]

**oc-newest** [*options*] [*state*]

**oc-newest** [*options*] [*state*] *lat lon*

**DESCRIPTION**

Fetch a list of newest geocaches from opencaching.com.

Requires:

**curl** <http://curl.haxx.se/>

Options:

- c** Report (include) cross-listed caches.
- f** Do not report any found or unavailable caches.
- F** Report caches found by the login 'username' as unfound.
- T datespec**  
Return caches placed or modified since 'datespec', which can be any date accepted by the **date(1)** command.
- n num** Return "num" caches [20]
- s** Output short names for the caches (gpsbabel option)
- I term** Include only caches with 'term' [\*]
- X term**  
Exclude caches with 'term' [**-unavail**]. Terms: unfound, ifound, unavail, regular, multi, virtual, webcam, event, hybrid, moving
- r radius**  
Display only caches with radius (e.g. **-r 25M**)
- M mystery**  
Use file 'mystery' for unknown/mystery/puzzle caches [/home/rick/.oc-mystery]. Awk Format: *gcid lat lon comment* i.e: OCXCBVB n44.45.123 w93.00.321 Final  
**OCXCC1Z 44.123456 -93.564123**  
Cache
- o format**  
Output format, **-o?** for possibilities [gpsdrive]. Plus "gpsdrive.sql" for direct insertion into MySQL DB. Plus "map[.geo-map-opts]" to display a geo-map.
- O filename**  
Output file, if not stdout
- S** Alias for **-o gpsdrive.sql**
- d** For **-S**, just delete selected records
- P** For **-S**, purge all records of type **-t Geocache\***
- t type** For **-ogpsdrive.sql**, the waypoint type [Geocache]
- H htmdir**  
Also fetch the printable HTML pages (slowly)

- L logdir** Also fetch the plain text log entries (slowly)
- D lvl** Debug level [0]
- U** Retrieve latest version of this script

Defaults can also be set with variables in file \$HOME/.georc:

```
LAT=latitude;      LON=logitude;
NUM=num;           UTFMT=format;      BABELFLAGS=-s;
SQLUSER=gast;     SQLPASS=gast;      SQLDB=geoinfo;
```

## EXAMPLES

Add newest 50 caches to a GpsDrive SQL database

```
oc-newest -n50 -f -s -S
```

Purge the existing SQL database of all geocaches, and fetch 200 fresh ones...

```
oc-newest -S -P -s -n200
```

Include cross-listed (i.e. gc.com) caches

```
oc-newest -c -s
```

## SEE ALSO

geo-newest, geo-nearest, geo-found, geo-placed, geo-code, geo-map, geo-waypoint, <http://geo.rkkda.com/>

**NAME**

**ok-nearest** - Fetch a list of nearest geocaches from opencaching.us

**SYNOPSIS**

**ok-nearest** [*options*]

**ok-nearest** [*options*] *lat lon*

**DESCRIPTION**

Fetch a list of nearest geocaches from opencaching.us.

Requires:

**curl** <http://curl.haxx.se/>

Options:

**-c** Report (include) cross-listed caches.

**-f** Do not report any found or unavailable caches.

**-F** Report caches found by the login 'username' as unfound.

**-T datespec**

Return caches placed or modified since 'datespec', which can be any date accepted by the **date(1)** command.

**-n num** Return "num" caches [20]

**-s** Output short names for the caches (gpsbabel option)

**-I term** Include only caches with 'term' [\*]

**-X term**

Exclude caches with 'term' [**-unavail**]. Terms: unfound, ifound, unavail, regular, multi, virtual, webcam, event, hybrid, moving

**-r radius**

Display only caches with radius (e.g. **-r 25M**)

**-M mystery**

Use file 'mystery' for unknown/mystery/puzzle caches [/dev/null]. Awk Format: *gcid lat lon comment* i.e: OU058C n44.45.123 w93.00.321 Final

**OU058D 44.123456 -93.564123**

Cache

**-o format**

Output format, **-o?** for possibilities [gpsdrive]. Plus "gpsdrive.sql" for direct insertion into MySQL DB. Plus "map[,geo-map-opts]" to display a geo-map.

**-O filename**

Output file, if not stdout

**-S** Alias for **-o gpsdrive.sql**

**-d** For **-S**, just delete selected records

**-P** For **-S**, purge all records of type **-t Geocache\***

**-t type** For **-ogpsdrive.sql**, the waypoint type [Geocache]

**-H htmldir**

Also fetch the printable HTML pages (slowly)

**-L logdir**

Also fetch the plain text log entries (slowly)

- E var=val** Set environment "var" to "val" i.e. DATEFMT=0|1
- D lvl** Debug level [0]
- U** Retrieve latest version of this script

Defaults can also be set with variables in file \$HOME/.georc:

```
LAT=latitude;          LON=logitude;
NUM=num;              OUTFMT=format;          BABELFLAGS=-s;
SQLUSER=gast;        SQLPASS=gast;          SQLDB=geoinfo;
OKBASE="http://www.opencaching.us";      OKCC=;
```

## EXAMPLES

Add nearest 50 caches to a GpsDrive SQL database

```
ok-nearest -n50 -f -s -S
```

Purge the existing SQL database of all geocaches, and fetch 200 fresh ones...

```
ok-nearest -S -P -s -n200
```

Nearest in UK:

```
ok-nearest -s -E OKBASE=http://www.opencaching.org.uk n53.5 w1.5
```

## SEE ALSO

geo-newest, geo-nearest, geo-found, geo-placed, geo-code, geo-map, geo-waypoint, ok-newest, <http://geo.rkkda.com/>

**NAME**

**ok-newest** - Fetch a list of newest geocaches from opencaching.us

**SYNOPSIS**

**ok-newest** [*options*] [*state*]

**ok-newest** [*options*] *state lat lon*

**ok-newest** [*options*] *country lat lon*

**DESCRIPTION**

Fetch a list of newest geocaches from opencaching.us.

Requires:

**curl** <http://curl.haxx.se/>

Options:

- c** Report (include) cross-listed caches.
- f** Do not report any found or unavailable caches.
- F** Report caches found by the login 'username' as unfound.
- T datespec**  
Return caches placed or modified since 'datespec', which can be any date accepted by the **date(1)** command.
- n num** Return "num" caches [20]
- s** Output short names for the caches (gpsbabel option)
- I term** Include only caches with 'term' [\*]
- X term**  
Exclude caches with 'term' [**-unavail**]. Terms: unfound, ifound, unavail, regular, multi, virtual, webcam, event, hybrid, moving
- r radius**  
Display only caches with radius (e.g. **-r 25M**)
- M mystery**  
Use file 'mystery' for unknown/mystery/puzzle caches [/dev/null]. Awk Format: *gcid lat lon comment* i.e: **OU058C n44.45.123 w93.00.321 Final**  
**OU058D 44.123456 -93.564123**  
Cache
- o format**  
Output format, **-o?** for possibilities [gpsdrive]. Plus "gpsdrive.sql" for direct insertion into MySQL DB. Plus "map[,geo-map-opts]" to display a geo-map.
- O filename**  
Output file, if not stdout
- S** Alias for **-o gpsdrive.sql**
- d** For **-S**, just delete selected records
- P** For **-S**, purge all records of type **-t Geocache\***
- t type** For **-ogpsdrive.sql**, the waypoint type [Geocache]
- H htmdir**  
Also fetch the printable HTML pages (slowly)

- L logdir**  
Also fetch the plain text log entries (slowly)
- E var=val**  
Set environment "var" to "val" i.e. DATEFMT=0|1
- D lvl** Debug level [0]
- U** Retrieve latest version of this script

Defaults can also be set with variables in file \$HOME/.georc:

```
LAT=latitude;          LON=logitude;
NUM=num;              OUTFMT=format;          BABELFLAGS=-s;
SQLUSER=gast;         SQLPASS=gast;          SQLDB=geoinfo;
OKBASE="http://www.opencaching.us";      OKCC=;
```

## EXAMPLES

Add newest 50 caches to a GpsDrive SQL database

```
ok-newest -n50 -f -s -S
```

Purge the existing SQL database of all geocaches, and fetch 200 fresh ones...

```
ok-newest -S -P -s -n200
```

Include cross-listed (i.e. gc.com) caches

```
ok-newest -c -s
```

Newest in UK:

```
ok-newest -s -E OKBASE=http://www.opencaching.org.uk uk n53.3 w1.5
```

Newest in Germany:

```
ok-newest -s -E OKBASE=http://www.opencaching.de germany n50 e7
```

## SEE ALSO

geo-newest, geo-nearest, geo-found, geo-placed, geo-code, geo-map, geo-waypoint, <http://geo.rkkda.com/>

**NAME**

**osg2ll** - British National Grid to Lat/Lon

**SYNOPSIS**

**osg2ll** [*options*] [*letters*] *easting northing*

**DESCRIPTION**

British National Grid, a.k.a. Ordnance Survey Grid to Lat/Lon.

**EXAMPLE**

Convert to DegDec:

```
$ osg2ll 338937 443358
Muggleton:      53.882610 -2.929045
Hannah:        53.882801 -2.930417
```

```
$ osg2ll SD 38937 443358
Muggleton:      53.882610 -2.929045
Hannah:        53.882801 -2.930417
```

Convert to RickDec:

```
$ osg2ll -orickdec 338937 443358
Muggleton:      n53.52.957 w02.55.743
Hannah:        n53.52.968 w02.55.825
```

**OPTIONS**

- l**     Print latitude only
- L**     Print longitude only
- odegdec**  
      Print lat/lon in DegDec format (default)
- omindec**  
      Print lat/lon in MinDec format
- orickdec**  
      Print lat/lon in dotted MinDec format
- odms**  Print lat/lon in DMS format
- D lvl**  Debug level

**SEE ALSO**

[http://www.dorcus.co.uk/carabus/ngr\\_ll.html](http://www.dorcus.co.uk/carabus/ngr_ll.html)

<http://www.hannahfry.co.uk/blog/2012/02/01/converting-british-national-grid-to-latitude-and-longitude-ii>

**NAME**

**pbnsolve-wrapper** - Wrapper for .non format nonograms

**SYNOPSIS**

**pbnsolve-wrapper** [*options*] skel.nono ...

**DESCRIPTION**

Nonogram (paint-by-number) solver using a program by Jan Wolter in Steven Simpson's .non format.

**OPTIONS**

- v** Clear screen and print the grid while it is being solved in case of Warning: puzzle has imbalance
- p png-file**  
Convert and image it into a PNG png-file
- P pdf-file**  
Convert and image it into a PDF pdf-file
- D lvl** Debug level

**EXAMPLE**

Solve GC6EQAP.nono:

```
$ pbnsolve-wrapper GC6EQAP.nono
GC6EQAP
```

UNIQUE SOLUTION:

```

ââ   ââ   ââââââââ   ââââââââ   ââ
ââ   ââ   ââ   ââ   ââ   ââ
ââââââââ   ââââââââ   ââââââââ   ââââââââââââ
      ââ       ââ       ââ   ââ   ââ
      ââ       ââ   ââââââââ   ââââââââââââ

ââ       ââââââââ   ââââââââ   ââââââââââ
ââ       ââ   ââ   ââ   ââ   ââ   ââ
ââââââââââ   ââââââââââ   ââ   ââ   ââ   ââ
ââ   ââ       ââ   ââ   ââ   ââ   ââ
ââââââââââ       ââ   ââââââââ   ââââââââââ

ââââââââââ   ââââââââ   ââââââââââ   ââââââââââ
ââ   ââ       ââ   ââ   ââ       ââ   ââ
ââââââââââ   ââââââââââ   ââââââââââ       ââ
ââ   ââ       ââ   ââ   ââ   ââ   ââ   ââ
ââââââââââ   ââââââââââ   ââââââââââ       ââââââââ
```

**SEE ALSO**

<http://webpbn.com/pbnsolve.html>  
 geo-nonogram

**NAME**

**pgpdb2txt** - Convert a Mapopolis Place Guide .pdb file to text

**SYNOPSIS**

**pgpdb2txt** [*options*] [*file*] ...

**DESCRIPTION**

Convert a Mapopolis Platinum Place Guide .pdb *file* to text. This is useful for creating a waypoint database for GpsDrive.

The **-F0** (default) output text format is:

```
Category|Name|StreetAddress|CityStateZip|Phone|Lat|Lon|
```

The **-F1** or **-F2** output format is:

```
Category|Name|StreetAddress|CityStateZip|Phone|Lat|Lon|Index|
```

The **-F3** (GpsDrive way.txt) output format is:

```
ShortName Lat Lon Category
```

The **-F4** (GpsDrive SQL) output format is:

```
ShortName Lat Lon Category Comment
```

**OPTIONS****-c category**

Select category [\*] category may be an RE, e.g. **-cRest.\***

**-n name**

Select name [\*] name may be an RE, e.g. **-n.\*McDonald.\***

**-l** Just list the categories in this *file*.

**-o dec** Output lat/lat in 'degdec' (44.456789) or 'mindec' (44.12.123) format.

**-r** Just print the lat/lon coverage rectangle of this *file*.

**-t type** The waypoint type to output [<category-in-pdb-file>]

**-u** Do not convert text to mixed case

**-F1** Append record number as Index

**-F2** Append filename and record number as Index

**-F3** Produce output compatible with GpsDrive v1.32 way.txt

**-F4** Produce output compatible with GpsDrive v1.32 SQL

**-d** For **-F4**, just delete selected records

**-D lvl** Set Debug level [0]

**NAME**

**rd2ll** - RD (Dutch) to Lat/lon

**SYNOPSIS**

**rd2ll** [*options*]

**DESCRIPTION**

RD (Dutch) to Lat/lon.

<http://www.dekoepel.nl/pdf/Transformatieformules.pdf>

**EXAMPLE**

Convert to DegDec:

```
$ rd2ll 86160 448438
52.02 4.38429
```

Convert to RickDec:

```
$ rd2ll -orickdec 86160 448438
n52.01.201 e4.23.057
```

**OPTIONS**

- l** Print latitude only
- L** Print longitude only
- odegdec**  
Print lat/lon in DegDec format (default)
- omindec**  
Print lat/lon in MinDec format
- orickdec**  
Print lat/lon in dotted MinDec format
- odms** Print lat/lon in DMS format
- D lvl** Debug level

**NAME**

**rect2geomap** - Calculate the scale, image width/height and lat/lon

**SYNOPSIS**

**rect2geomap** [*options*] *scale latUL lonUL lat LR lonLR*

**DESCRIPTION**

Calculate the *scale*, image width/height and *lat/lon* center point command line arguments for geo-map that will enclose a *lat/lon* rectangle at the specified *scale* factor.

**OPTIONS****-P pixelfact**

Override the default pixel factor [2817.947378]

**-D lvl** Debug level**EXAMPLE**

Calculate the *scale*...

```
$ rect2geomap 50000 45.25 -93.375 44.75 -92.675  
-s50000 -W3113 -H3131 45 -93.025
```

**NAME**

**reverse-montage** - reverse (split up) montage image

**SYNOPSIS**

**reverse-montage** [*options*] *rows columns file ...*

**DESCRIPTION**

Reverse (split up) a montage image. Files are written to *file-row-col.jpg*.

**OPTIONS****-x offset**

Left offset (0)

**-X offset**

Right offset (0)

**-y offset**

Top offset (0)

**-Y offset**

Bottom offset (0)

**-D lvl** Debug level

**EXAMPLES**

Reverse montage <http://coord.info/GC491T7>:

```
$ reverse-montage 3 5 ladies.png  
$ eog ladies-*.jpg
```

Reverse montage <http://coord.info/GC48JGN>:

```
$ reverse-montage 3 3 surv1.jpg  
$ reverse-montage 3 3 surv2.jpg  
$ eog surv1-*.jpg surv2-*.jpg
```

**NAME**

**reverse-wherego** - Reverse Wherego Decoder

**SYNOPSIS**

**reverse-wherego** [*options*] [*code-A*] [*code-B*] [*code-C*]

**DESCRIPTION**

Reverse Wherego decoder. Code's A, B, and C are off the geocache description, i.e. <http://coord.info/GC6GJX1>, <https://coord.info/GC672VM>, <https://coord.info/GC71ADB>, <https://coord.info/GC65FXK>, or <https://coord.info/GC7B2T7>.

**reverse-wherego** can use command line arguments or it can read from stdin.

**OPTIONS**

**-c** Ignore the checkdigit for 7 letter codes.

**-D lvl** Debug level

**EXAMPLE**

Decode GC6GJX1 ("The Reverse Cache - beta):

```
$ reverse-wherego 809382 480302 325138
42.931380 -88.032280 n42.55.883 w88.01.937
```

Decode GC672VM ("Generic Cartridge"):

```
$ reverse-wherego 01733414227001748839 61733424947042719489
start: 44.778300 -93.170420 n44.46.698 w93.10.225
final: 44.774860 -93.174190 n44.46.492 w93.10.451
```

Decode GC71ADB ("Wonder Wherego"):

```
$ reverse-wherego 590325 488604 587219 090325 484605 525279
start: 45.052150 -92.964880 n45.03.129 w92.57.893
final: 45.052700 -92.965280 n45.03.162 w92.57.917
```

Decode GC65FXK ("Wonder Wherego V1.5")

```
$ reverse-wherego 6043354 7740432 9909731 6043756 7730403 9289630 \
5043254 7740465 9319531
start: 44.796750 -93.073930 n44.47.805 w93.04.436
middle: 44.796650 -93.077200 n44.47.799 w93.04.632
final: 44.795550 -93.072360 n44.47.733 w93.04.342
```

Decode GC7B2T7 ("Wonder Wherego Time Trials V1.4")

```
$ reverse-wherego 1953246 4350030 1003794 1853247 4008006 2340694 \
8853241 4098002 1059794 \
3445695
start: 44.513710 -92.903030 n44.30.823 w92.54.182
middle: 44.520610 -92.880300 n44.31.237 w92.52.818
final: 44.519780 -92.880000 n44.31.187 w92.52.800
threshold: 5.87 ft/sec or 4.002273 MPH
```

Decode GC7C0CA and GC7C0CC and GC7DKTA from stdin:

```
$ reverse-wherego
Enter code(s):
```

309334141510603650793  
44.930730 -93.016610 n44.55.844 w93.00.997  
404385079294629323120  
44.934150 -92.978360 n44.56.049 w92.58.702  
304313209504079854437809334340500374145891599324240590414854595  
start: 44.943430 -93.001800 n44.56.606 w93.00.108  
middle: 44.945880 -93.003100 n44.56.753 w93.00.186  
final: 44.944550 -92.994800 n44.56.673 w92.59.688

**SEE ALSO**

<https://www.geocaching.com/bookmarks/view.aspx?guid=eedd6021-ec17-4095-a14f-36f2b6a609f3>

**NAME**

**segment2text** - N-Segment Display to Text

**SYNOPSIS**

**segment2text** [*options*] [*display*] ...

**DESCRIPTION**

N-segment *display* to text from stdin or command line args. The default is **-14**.

**OPTIONS**

- 7**     7 segment *display*
- 9**     9 segment *display*
- 14**    14 segment *display*
- 16**    16 segment *display*, but the displays **MUST** be sorted
- D lvl** Debug level

**EXAMPLES**

7 segment *display* from <http://coord.info/GC1AGGD>:

```
$ segment2text -7 DGFAC CB BCDEFA AGDBCEF FDECAB BCGF FAGCD
5108045
```

14 segment *display* from <http://coord.info/GC5TFEZ>:

```
$ segment2text -14
BCEFHK ABCDEF ABCDEF ABEFG1G2K ABCDIL
noord
```

**SEE ALSO**

<http://www.geocachingtoolbox.com/index.php?lang=en&page=segmentDisplay>

**NAME**

**smilies2cryptogram** - Geocaching 'smilies' to cryptogram

**SYNOPSIS**

**smilies2cryptogram** [*options*]

**DESCRIPTION**

Geocaching 'smilies' to cryptogram. Uses the source of the page.

**EXAMPLE**

Use curl to download:

```
$ curl -L -s http://coord.info/GC4DAK7 | smilies2cryptogram
ABCDEFGHIJKBLKJMNEDFGHIJDECJJMNEDOBDPJCBAMAJKMQJFGHIJRJFDFGHIJ
BAJDRBDRBFGHIJBAJJMNEDOBDDRBRBDRBKMQJ
```

Then, go online to <http://www.quipqiup.com/> . This one uses clues: d=t e=h c=r j=e a=n k=f

**OPTIONS**

**-D lvl** Debug level

**NAME**

**spiritdvd2text** - Spirit DVD Code to/from text

**SYNOPSIS**

**spiritdvd2text** [*options*]

**DESCRIPTION**

Spirit DVD Code to or from text. Decoding uses '1', 'l', '|', or 'I' as aliases for '1'.

**OPTIONS**

**-e** Encode

**-D lvl** Debug level

**EXAMPLES**

Decode:

```
$ echo "11--1--111--11----11--11" | spiritdvd2text  
NORTH
```

Encode:

```
$ echo NORTH | spiritdvd2text -e  
11--1--111--11----11--11
```

**SEE ALSO**

<http://www.planetary.org/explore/projects/redrover/mars-dvd-code-clues.html>

<http://rumkin.com/tools/cipher/substitution.php>

**NAME**

**tap-code** - Tap Code or Polybius Square decoder

**SYNOPSIS**

**tap-code** [*options*] [*codes*]

**DESCRIPTION**

The tap code, sometimes called the knock code, is a way to encode messages, letter by letter, in a very simple way and transmit it using a series of tap sounds, hence its name. It has been commonly used by prisoners to communicate with each other. The method of communicating is usually by "tapping" either the metal bars, pipes or the walls inside the cell.

**tap-code** can read from stdin or on the command line.

With **-c** (default):

row/col	1	2	3	4	5
1	A	B	C/K	D	E
2	F	G	H	I	J
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

Or with **-i**:

row/col	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	I/J	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

Or with **-u**:

row/col	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	I	J
3	K	L	M	N	O
4	P	Q	R	S	T
5	U/V	W	X	Y	Z

Or with **-6**:

row/col	1	2	3	4	5	6
1	A	B	C	D	E	F
2	G	H	I	J	K	L
3	M	N	O	P	Q	R
4	S	T	U	V	W	X
5	Y	Z	0	1	2	3
6	4	5	6	7	8	9

It can use a period (.) or an asterisk (\*). A slash (/) is translated to a space.

A polybius square is like a tap code except you use two numbers instead of tapping.

**OPTIONS**

**-n** Numbers instead of '.' or '\*'

- r** Col/Row instead of Row/Col
- c** Combine c/k (default)
- i** Combine i/j instead of c/k
- u** Combine u/v instead of c/k
- 6** Matrix is 6x6, A..Z and 0..9 can be done.
- D lvl** Debug level

## EXAMPLES

Decode two words by tapping:

```
$ echo ". . / . . . . . . . . . . . . . . ." | tap-code
a tap
```

Decode 5 words by polybius square:

```
$ echo "44232443 2443 11 431513421544 32154343112215" | tap-code -i -n
this is a secret message
```

Decode from the command line:

```
$ tap-code 44-23-15 / 42-24-13-23 / 52-11-33-44 / 24-44
the rich want it
```

Decode with 6x6 matrix:

```
$ tap-code -6 31 15 15 42 / 42 22 43 / 55 56 53 53 22 36
meet thu 2300hr
```

## SEE ALSO

[http://en.wikipedia.org/wiki/Tap\\_code](http://en.wikipedia.org/wiki/Tap_code)

[http://en.wikipedia.org/wiki/Polybius\\_square](http://en.wikipedia.org/wiki/Polybius_square)

**NAME**

**update-caches** - Update caches

**SYNOPSIS**

**update-caches** [*options*] [*place*]

**DESCRIPTION**

Update caches. Super script for rick.

**OPTIONS**

**-0** Update, but no geo-nearest/geo-newest  
**-a** All. Do geo-nearest AND geo-newest.  
**-f** Find  
**-F** Force  
**-i** Incremental (200)  
**-I** Incremental (2000)  
**-n** Use geo-newest instead of geo-nearest  
**-p** Puzzles  
**-q** Quick (60)  
**-D lvl** Debug level

**EXAMPLES**

Incremental:

```
$ update-caches -i
```

Newest incremental:

```
$ update-caches -i -n
```

Newest incremental in Shakopee:

```
$ update-caches -i -n shakopee
```

Newest incremental puzzles:

```
$ update-caches -i -n -p
```

Nearest incremental puzzles:

```
$ update-caches -i -p
```

**PLACES**

[0-9]\*|n[0-9]\* shak\* bloom\* stpaul andy buff\* norwood elm\* blaine blainea\* rich\* cache2000 zoo brook\* fair winsted wac\* lake\* white\* falls vict\* ne\* maple\* home|"" n40 w40 s40 e40 hudson

**NAME**

**urwigo-decode** - urwigo and earwigo decoder

**SYNOPSIS**

**urwigo-decode** [*options*]

**DESCRIPTION**

urwigo and earwigo decoder.

**OPTIONS**

**-v** Verbose. Print the undecoding string as well

**-D lvl** Debug level

**EXAMPLES**

Decode a wherigo:

```
$ wherigo2lua how_the_grinch_sto.gwc >xxx
```

```
$ urwigo-decode <xxx | tail -1
```

```
The Grinch steals your Stable Key! Then he dissappears off toward the Northeast.
```

**NAME**

**usng2ll** - US National Grid to Lat/long

**SYNOPSIS**

**usng2ll** [*options*]

**DESCRIPTION**

US National Grid to Lat/lon.

Also known as "Military Grid Reference System" (MGRS).

<http://dhost.info/usngweb/>

**EXAMPLE**

Convert:

```
$ usng2ll 17R LN 64066 80742
wp = 29.649226 -82.404421    n29.38.954 w82.24.265
```

```
$ usng2ll 15T VJ 22779 63998
wp = 43.925049 -93.961954    n43.55.503 w93.57.717
```

```
$ usng2ll 46T EP 77220 63998
wp = 43.925049 93.961942    n43.55.503 e93.57.717
```

**OPTIONS**

**-D lvl** Debug level

**NAME**

**ll2utm** - Convert lat/lon to Universal Transverse Mercator (UTM) format

**SYNOPSIS**

ll2utm [options] lat lon

Convert WGS-84 latitude/longitude to UTM. See examples for the input formats which are acceptable.

**OPTIONS**

- e** Print easting only. May be combined with **-n**.
- n** Print northing only. May be combined with **-e**.
- m** Print multi-line results (one field per line)
- D lvl** Set Debug level [0]

**EXAMPLES**

All of the below are equivalent and output this result... 15 T 459594 4928460

# DegDec format

```
ll2utm 44.508333 -93.508333
ll2utm N44.508333 W93.508333
```

# Rick style MinDec format

```
ll2utm N 44.30.50 W 93.30.50
ll2utm N44.30.50 W93.30.50
ll2utm 44.30.50 -93.30.50
ll2utm "N 44.30.50" "W 93.30.50"
ll2utm "N44.30.50" "W93.30.50"
ll2utm "44.30.50" "-93.30.50"
```

# Space separated MinDec format

```
ll2utm N 44 30.50 W 93 30.50
ll2utm N44 30.50 W93 30.50
ll2utm 44 30.50 -93 30.50
ll2utm "N 44 30.50" "W 93 30.50"
ll2utm "N44 30.50" "W93 30.50"
ll2utm "44 30.50" "-93 30.50"
```

# gc.com web page cut and paste format

```
ll2utm N 44° 30.50 W 093° 30.50
ll2utm "N 44° 30.50" "W 093° 30.50"
```

# DMS format

```
ll2utm 44 30 30 W93 30 30
ll2utm 44 30 30 -93 30 30
ll2utm "44 30 30" "W93 30 30"
ll2utm "44 30 30" "-93 30 30"
ll2utm 44 30 29.99999999 W93 30 29.99999999
ll2utm 44 30 29.99999999 -93 30 29.99999999
```

**NAME**

**utm21l** - Convert Universal Transverse Mercator (UTM) to lat/lon

**SYNOPSIS**

**utm21l** [*options*] *zone* [*nz*] *easting* *northing*

**utm21l** [*options*] *zone\_nz* *E* *easting* *N* *northing*

**Convert** *UTM* to *WGS-84 DegDec* latitude/longitude.

**OPTIONS**

**-l** Print latitude only

**-L** Print longitude only

**-odegdec**

Print lat/lon in *DegDec* format (default)

**-omindec**

Print lat/lon in *MinDec* format

**-orickdec**

Print lat/lon in Rick's *MinDec* format

**-odms** Print lat/lon in *DMS* format

**-D lvl** Set Debug level [0]

**EXAMPLES**

**Convert:**

```
$ utm21l 15 T 460601 4972618
44.905897 -93.499074
```

```
$ utm21l 13T E 511168 N 4553536
41.133054 -104.866940
```

**NAME**

**wherigo2jpg** - Pull jpg images out of a wherigo file

**SYNOPSIS**

**wherigo2jpg** [*options*] [wherigo.gwc]

**DESCRIPTION**

Pull jpg images out of a wherigo file and write them to "wherigoNNN.jpg".

Needs gawk.

**OPTIONS**

**-d dir** Directory to write images [.]

**-D lvl** Debug level

**EXAMPLE**

Survivor:

```
$ wherigo2jpg -d tmp survivor.gwc
```

```
$ eog tmp/*.jpg
```

**NAME**

**wherigo2lua** - De-compile wherigo

**SYNOPSIS**

**wherigo2lua** [*options*] *file*

**DESCRIPTION**

De-compile a wherigo to stdout. gwcd output is redirected to stderr.

Uses python for gwcd, and java for unluac.

Just a wrapper script which contains:

```
python /home/rick/Wherigo/gwcd/gwcd.py --all $1 >&2
java -jar /home/rick/Wherigo/unluac_2015_06_13.jar cartridge.luac
```

**OPTIONS**

**-d dir** Directory of wherigo files (/home/rick/Wherigo)

**-D lvl** Debug level

**EXAMPLES**

Various examples:

```
$ wherigo2lua antlers_amusement_.gwc > xxx
```

```
$ wherigo2lua you_cant_take_it_w.gwc > xxx
$ urwigo-decode <xxx
```

```
$ wherigo2lua historic_mackinaw.gwc >xxx
$ geo-map -s0 -c $(zonepoint2map <xxx)
```

```
$ reverse-wherigo 809382 480302 325138
42.931380 -88.032280 n42.55.883 w88.01.937
```

**SEE ALSO**

reverse-wherigo, urwigo-decode, zonepoint2map

<https://github.com/driquet/gwcd>

/home/rick/Wherigo/gwcd/gwcy.py

<https://sourceforge.net/projects/unluac/>

/home/rick/Wherigo/unluac\_2015\_06\_13.jar

**NAME**

**zonepoint2map** - Convert Wherigo ZonePoint's to geo-map coords

**SYNOPSIS**

**zonepoint2map** [*options*] [*file*]

**DESCRIPTION**

Convert Wherigo ZonePoint's to geo-map coords. Reads from stdin or *file*.

**OPTIONS**

**-D lvl** Debug level

**EXAMPLE**

Convert:

```
$ wherigo2lua historic_mackinaw.gwc >xxx
```

```
$ geo-map -s0 -c $(zonepoint2map xxx)
```

**SEE ALSO**

reverse-wherigo, urwigo-decode, wherigo2lua

zonepoint2map(1)

zonepoint2map(1)