

```
1: #!/usr/bin/perl
2: # $Id: haversine.perl,v 1.7 2018-02-28 16:40:07-08 - - $
3:
4: # Find distance between two airports using the haversine formula.
5: # http://andrew.hedges.name/experiments/haversine/
6: # Airport database is in prolog syntax.
7:
8: use strict;
9: use warnings;
10: $0 =~ s|.|/||;
11:
12: my $PI = 3.141592653589793238462643383279502884;
13: my $EARTH_RADIUS_MILES = 3961;
14:
15: my $database_name = ".score/database.pl";
16:
17: my %database;
18: open DATABASE, "<$database_name" or die "$0: $database_name: $!";
19: while (<DATABASE>) {
20:     next unless m/airport\\(\\s*(.*?),\\s*'(.*)',\\s*
21:                 degmin\\(\\s*(\\d+),\\s*(\\d+)\\s*\\),\\s*
22:                 degmin\\(\\s*(\\d+),\\s*(\\d+)\\s*\\)\\s*\\)/x;
23:     my ($airport, $name, $nlatdeg, $nlatmin, $wlondeg, $wlonmin)
24:         = ($1, $2, $3, $4, $5, $6);
25:     $airport = uc $airport;
26:     $database{$airport} = [$name, $nlatdeg, $nlatmin,
27:                             $wlondeg, $wlonmin];
28: }
29: close DATABASE;
30:
31: sub radians ($$) {
32:     # Convert degrees and minutes of arc to radians.
33:     my ($degrees, $minutes) = @_;
34:     return ($degrees + $minutes / 60) * $PI / 180;
35: }
36:
37: sub print_location(@) {
38:     my ($deg, $min, $dir) = @_;
39:     printf "%3d°%2d' %s(%6.2f°, %6.4f)",
40:           $deg, $min, $dir, $deg + $min / 60, radians ($deg, $min);
41: }
42:
43: sub print_airport($$) {
44:     my ($airport, $data) = @_;
45:     printf "%-3s (%-16s)", $airport, $$data[0];
46:     print_location @$data[1,2], "N";
47:     print_location @$data[3,4], "W";
48:     printf "\\n";
49: }
50:
51: for my $airport (sort keys %database) {
52:     print_airport $airport, $database{$airport};
53: }
54:
```

```
55:
56: my $circumference = 2 * $PI * $EARTH_RADIUS_MILES;
57: printf "\n";
58: printf "Earth radius:          %7.1f miles\n", $EARTH_RADIUS_MILES;
59: printf "Earth circumference: %7.1f miles\n", $circumference;
60: printf "Earth 1 degree arc:   %7.1f miles\n", $circumference / 360;
61: printf "Earth 1 minute arc:   %7.1f miles\n", $circumference / 360 / 60;
62: printf "Earth 1 radian arc:   %7.1f miles\n", $circumference / $PI / 2;
63:
64: sub haversine_distance ($$$$) {
65:     # Latitude1, longitude1 in radians.
66:     # Latitude2, longitude2 in radians.
67:     my ($lat1, $lon1, $lat2, $lon2) = @_;
68:     my $dlon = $lon2 - $lon1;
69:     my $dlat = $lat2 - $lat1;
70:     my $tmpa = (sin ($dlat / 2)) ** 2
71:         + cos ($lat1) * cos ($lat2) * (sin ($dlon / 2)) ** 2;
72:     my $unit_distance = 2 * atan2 (sqrt ($tmpa), sqrt (1 - $tmpa));
73:     my $distance_miles = $EARTH_RADIUS_MILES * $unit_distance;
74:     return $distance_miles;
75: }
76:
77: while (@ARGV >= 2) {
78:     my $airport1 = shift; $airport1 = uc $airport1;
79:     my $airport2 = shift; $airport2 = uc $airport2;
80:     my $data1 = $database{$airport1};
81:     my $data2 = $database{$airport2};
82:     warn "$0: $airport1, $airport2: invalid airport\n" and next
83:         unless $data1 && $data2;
84:     my $lat1 = radians ($data1->[1], $data1->[2]);
85:     my $lon1 = radians ($data1->[3], $data1->[4]);
86:     my $lat2 = radians ($data2->[1], $data2->[2]);
87:     my $lon2 = radians ($data2->[3], $data2->[4]);
88:     my $distance = haversine_distance ($lat1, $lon1, $lat2, $lon2);
89:     print "\nDistance:\n";
90:     print_airport $airport1, $data1;
91:     print_airport $airport2, $data2;
92:     printf "%.0f miles\n", $distance;
93: }
```

```
1: COMMAND: haversine.perl lax sfo sjc nyc sfo sea
2:
3: ATL (Atlanta           ) 33°39'N( 33.65°,0.5873) 84°25'W( 84.42°,1.4733)
4: BOS (Boston-Logan      ) 42°22'N( 42.37°,0.7394) 71° 2'W( 71.03°,1.2398)
5: CHI (Chicago           ) 42° 0'N( 42.00°,0.7330) 87°53'W( 87.88°,1.5339)
6: DEN (Denver-Stapleton) 39°45'N( 39.75°,0.6938) 104°52'W(104.87°,1.8303)
7: DFW (Dallas-Ft.Worth ) 32°54'N( 32.90°,0.5742) 97° 2'W( 97.03°,1.6936)
8: LAX (Los Angeles       ) 33°56'N( 33.93°,0.5922) 118°24'W(118.40°,2.0665)
9: MIA (Miami             ) 25°49'N( 25.82°,0.4506) 80°17'W( 80.28°,1.4012)
10: NYC (New York City     ) 40°46'N( 40.77°,0.7115) 73°59'W( 73.98°,1.2913)
11: SEA (Seattle-Tacoma    ) 47°27'N( 47.45°,0.8282) 122°18'W(122.30°,2.1345)
12: SFO (San Francisco     ) 37°37'N( 37.62°,0.6565) 122°23'W(122.38°,2.1360)
13: SJC (San Jose         ) 37°22'N( 37.37°,0.6522) 121°56'W(121.93°,2.1281)
14:
15: Earth radius:          3961.0 miles
16: Earth circumference: 24887.7 miles
17: Earth 1 degree arc:    69.1 miles
18: Earth 1 minute arc:    1.2 miles
19: Earth 1 radian arc:    3961.0 miles
20:
21: Distance:
22: LAX (Los Angeles       ) 33°56'N( 33.93°,0.5922) 118°24'W(118.40°,2.0665)
23: SFO (San Francisco     ) 37°37'N( 37.62°,0.6565) 122°23'W(122.38°,2.1360)
24: 339 miles
25:
26: Distance:
27: SJC (San Jose         ) 37°22'N( 37.37°,0.6522) 121°56'W(121.93°,2.1281)
28: NYC (New York City     ) 40°46'N( 40.77°,0.7115) 73°59'W( 73.98°,1.2913)
29: 2552 miles
30:
31: Distance:
32: SFO (San Francisco     ) 37°37'N( 37.62°,0.6565) 122°23'W(122.38°,2.1360)
33: SEA (Seattle-Tacoma    ) 47°27'N( 47.45°,0.8282) 122°18'W(122.30°,2.1345)
34: 680 miles
```