

Assignment: GPS unit for hikers

GPS has been built in to many consumer devices. Here we consider the GPS unit typically used by hikers and hill-walkers. An important feature of such units is the ability to mark *waypoints* and collect them into a path that can be retrieved later. Another important feature is the distance and bearing to a waypoint - you have a target waypoint and want to know in what direction it is and how far away it is.

When queried, the GPS generates the current longitude and latitude - a pair of floating point numbers. The range of longitude is -180 degrees to +180 degrees with 0 being the meridian (line) which passes through Greenwich, England. The range of latitude is -90 degrees to +90 degrees with 0 being the Equator. This means that values for Ireland will always be negative for longitude and positive for latitude.

Write a method to create a reference to current location that returns a longitude/latitude pair. This method uses the `random` function. The ranges for Ireland are approximately $-11.0 < longitude < -5.0$ and $51.0 < latitude < 55.0$.

A *waypoint* will be a position (lon/lat pair) and a name.

A *route* will be an ordered collection of two or more waypoints.

Your program should implement this functionality:

1. Set and retrieve the current location. Save the current position as a named waypoint. Call it 'CurrentLocation'
2. Set and retrieve other user-defined waypoints".
3. Save and retrieve named paths consisting of a sequence of waypoints.
4. Calculate the distance to a given waypoint from the current location.
5. Calculate the *direction* as a compass bearing from the current location to a given waypoint. Range is $0 \leq bearing < 360$.

The formula for **bearing** is:

$$y = \sin(\lambda_2 - \lambda_1) * \cos(\phi_2)$$
$$x = \cos(\phi_1) * \sin(\phi_2) - \sin(\phi_1) * \cos(\phi_2) * \cos(\lambda_2 - \lambda_1)$$

$brng = \text{atan2}(y, x)$ expressed in degrees.

λ is longitude

ϕ is latitude

1 or 2 refers to the source (current) or target point (other)

So $\lambda_2 - \lambda_1$ is equivalent to `other.lon - self.lon`

Create a loop/menu that prompts the user with choices based on the above.
Test all methods.