Survey Mapping Challenge

GEO 4910

20-point challenge

Task

General Description

Create a python function to read survey data from a standardized CSV file and construct a polygon representing the surveyed land parcel.

Input

- *A path (string) referencing a standardized CSV file with survey data for a single land parcel, containing the following columns:*

  (a) NS : bearing base direction (either "N" or "S")

  (b) deg : bearing offset from the base direction, in degrees

  (c) EW : direction of bearing offset (either "E" or "W")
  Together the above three fields indicate a "bearing", i.e. the direction of the survey line.

  (d) poles : distance of the survey line (1 pole = 16.5 feet)

  (e) description : a description of the survey corner.
  The description field will not be used for this challenge.

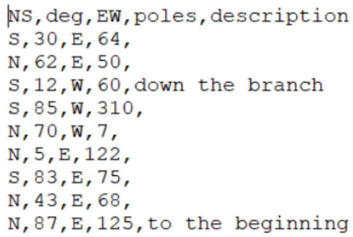
Output

- *A polygon, represented as list of tuples indicating points, and displayed on matplotlib.*

Logical Specification

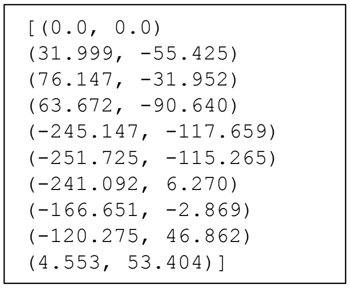
The output polygon will be a list of points starting from an arbitrary location, e.g. (0.0,0.0). Each subsequent point will be located at the specified distance and bearing from the previous point. The polygon might not close exactly (keep in mind that the data provided comes from 18th century surveys!) but it should nearly close, i.e. the last point should usually be within ~50 poles of the first point.

Example



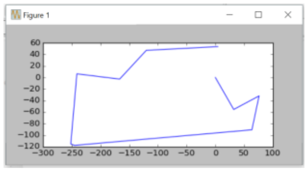
```
NS,deg,EW,poles,description
S,30,E,64,
N,62,E,50,
S,12,W,60,down the branch
S,85,W,310,
N,70,W,7,
N,5,E,122,
S,83,E,75,
N,43,E,68,
N,87,E,125,to the beginning
```

input CSV File ("...\broadwater434.csv")



```
[(0.0, 0.0)
(31.999, -55.425)
(76.147, -31.952)
(63.672, -90.640)
(-245.147, -117.659)
(-251.725, -115.265)
(-241.092, 6.270)
(-166.651, -2.869)
(-120.275, 46.862)
(4.553, 53.404)]
```

output data



output data plotted to matplotlib

Submission

Submit your working code, fully documented, as one or more python files. Be sure to include your name in comments at the top of each script.

Additional Notes

Survey Data

Data for this lab is provided in the included zip folder. Unzip this and you will see two subfolders.

The *grant_data* subfolder contains seven csv files. Your script should work on any of these files, by modifying only the input data reference. You should include a reference to this file at the top of your script, so that it may be easily modified.

The *grant_images* subfolder contains images of each of the original land grants. You do not need to use these images to complete the challenge, they are provided for reference only.

Re-use your Code

Don't reinvent the wheel! You should be able to re-use at least one function from the *gis_utils* assignment to complete this challenge.

Code Structure

For full points, your main code should be packaged as a function, with the input parameters and output (return) value as specified above. Plot your data to matplotlib – how well do the polygons close?