

EEN118 LAB NINE

This lab involves some scientific data processing. You will download a Geographical Database File which contains the coordinates of the boundaries of the 48 connected states and the five great lakes, and write a program that draws maps requested by the user.

Download the file “usamap.txt” from the class web site. The format of the file is very simple; it describes the outlines of the 48 older states plus the five great lakes. Here is the beginning of the file:

```
FL
81500 30666
80500 28500
80166 27000
80166 26000
80416 25166
81166 25166
81750 26000
82916 27833
82833 29000
84000 30166
85166 29666
86000 30333
87500 30416
87500 31000
85000 31000
84833 30666
82250 30500
82216 30300
82050 30300
81950 30716
81500 30666
-1 -1
AL
85000 31000
85000 32500
85666 35000
88166 35000...
```

and it continues like that for a long time. The first line “FL” indicates that this is the description of a state: FL is the postal abbreviation for Florida. The next 21 lines give the coordinates of a point along the border of the state. The coordinates are actually longitude and latitude measured in thousandths of a degree, but you can treat them simply as x and y values. The two -1’s after the list of numbers are simply to give you an easy way of telling that the list has finished (no real data in this file is ever negative). Then you see “AL” introducing the next state, Alabama’s, description.

The whole file is just like that. First a state’s abbreviation, then a list of coordinates, then -1 -1, all repeated 55 times. Notice that the coordinates of the last point are the same as the coordinates of the first point (81500, 30666). This is true of every state’s description; they all make nice closed figures. The coordinates 81500, 30666 represent the point $30\frac{2}{3}$ degrees North of the equator and $81\frac{1}{2}$ degrees West of the Greenwich meridian.

At the end of the file, which looks like this...

```
KS
102000 40000
102000 37000
 94666 37000
 94666 39166
 95333 40000
 95333 40000
102000 40000
-1 -1
END
```

... you'll find the word `END`. The last state, Kansas's description ends as usual with `-1 -1`, then the word `END` appears. No state has "END" as its postal abbreviation. There is no data following the "END".

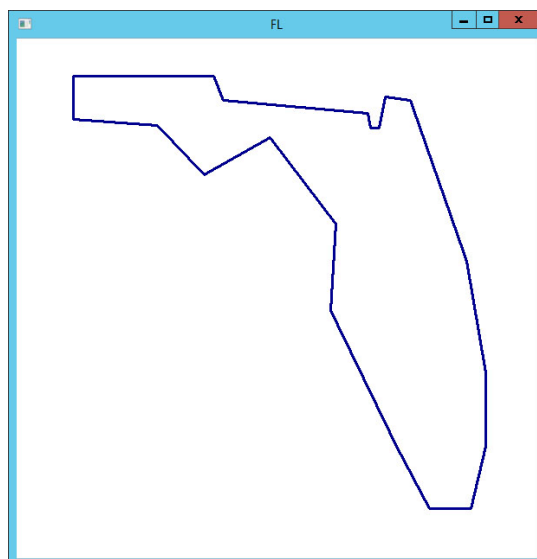
For your information, these are the extreme data values that appear in the file:

Minimum Longitude (x)	67000
Maximum Longitude (x)	124666
Minimum Latitude (y)	25166
Maximum Latitude (y)	49333

The states all have their standard two-letter postal abbreviations. The great lakes are given three-letter abbreviations: `LKE`, `LKH`, `LKM`, `LKO`, and `LKS`.

1. *Draw Florida*

Write a program that opens a reasonably large graphics window (you decide the size), and draws the outline of the first state that appears in the data file (i.e. Florida). You will have to scale and shift the coordinates before plotting, as one of the points in Florida is (81500, 30666) and there's no way you're going to get a window that big. Make sure your picture comes out the right way round:



2. Draw any state

Modify your program so that it asks the user to enter a state's (or lake's) abbreviation, and then draws that state. It should not draw anything else, just the outline of the selected state. Do not worry about sizing the window to fit that state properly. Make the window big enough to draw the whole country, and just draw the one state in its correct position.

3. Did you get it right?

Two of the states, Michigan (MI) and Virginia (VA) are not contiguous; they come in two sections separated from each other by water. The two sections of these states have their own descriptions in the data file: there are two sections beginning with VA and two beginning with MI. Make sure that if the user requests a non-contiguous state, all of its portions are drawn.

4. Label the States

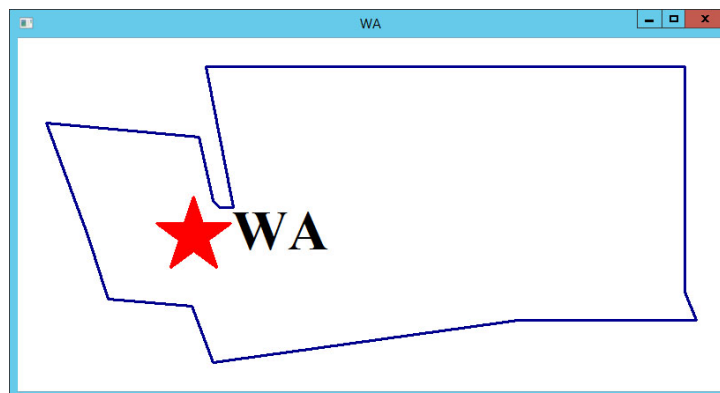
There is a second data file associated with this lab, called `capitals.txt`. Download it too. It has exactly 50 lines, and this is what the first four look like.

```
AL Montgomery 32.354 86.284
AK Juneau 58.388 134.133
AZ Phoenix 33.542 112.071
AR Little_Rock 34.722 92.354
```

The file has a line for each state, and each line contains exactly four pieces of information: the state's postal abbreviation, the name of its capital, and the latitude and longitude of its capital.

Notice that when a name consists of more than one word, an underline is used instead of a space, so that you can use the `>>` operator to read the file without trouble. Notice also that in this file latitude and longitude are measured in degrees as floating point numbers.

Improve your program so that every time it draws a state, it searches through this second file to find that state's capital. It should draw on the map, at the correct position to mark the capital, a star with the state's abbreviation beside it. This is Washington, Not-D.C.



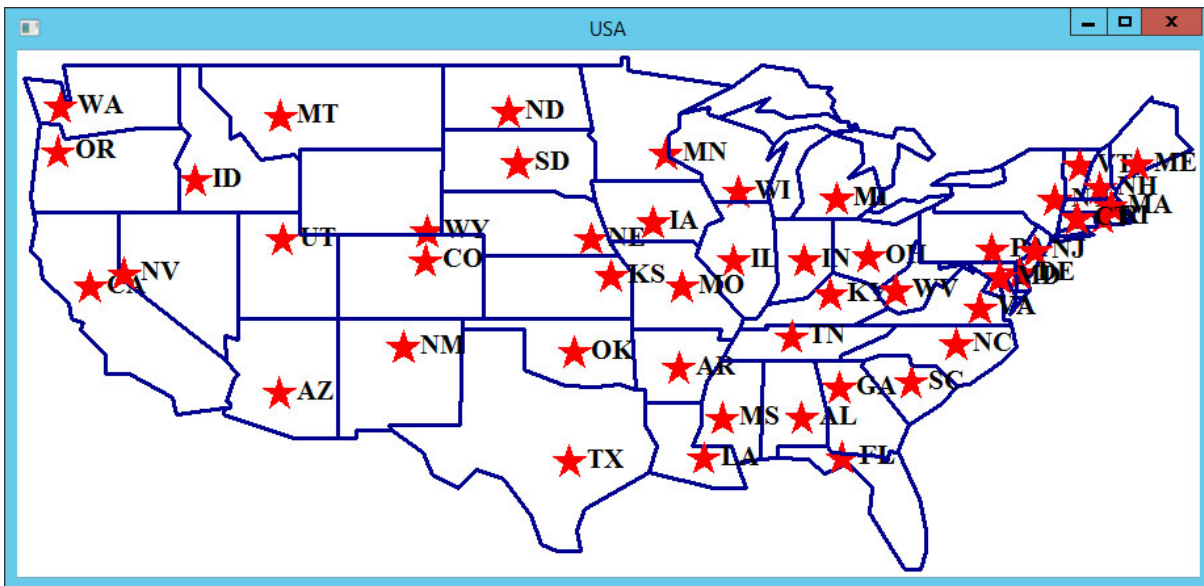
Remember that you'll need to reset the capitals file to its beginning after each search, or there'll be trouble.

5. *Make it incremental*

Add a loop to your program. After creating the window big enough to draw the whole country, it should enter the loop, repeatedly asking the user to enter a state abbreviation. For each abbreviation entered, it should add the outline of that state to what has already been drawn, so that the user can build up a map of many states if desired. Remember to open the file each time round the loop, and close it before the end of the loop; that way you'll be able to read it afresh from the beginning each time.

6. *Make it do everything.*

If the user enters ALL instead of a state abbreviation, your program should draw the outlines of all the states (in the same window), so that a map of the whole country appears. Of course the program should still draw individual states if the input is not "ALL". The capitals should also be marked, but you may want to select a smaller font to make it fit.



The map will look a bit over-crowded unless you create a really big window, but there isn't much that could reasonably be done about that.

7. EXTRA CREDIT.

I'm sure you remember that the library contains these three useful functions: `wait_for_mouse_click()`, `get_click_x()`, and `get_click_y()`. After drawing the whole country, make your program wait for the user to click the mouse somewhere, then report back the name of the state capital closest to that position.

Even better, there is another function, `get_pixel_color(x, y)`, which returns the colour of the pixel at position `x, y`, encoded as an int. If you colour all the states in with different colours when drawing their outlines, you can use this function to tell which state the user clicked on.

Make your program into a geography quiz. Don't draw the state capitals or names initially. Instead, choose a random state and tell the user to click on it. Don't let them go home until they have clicked in the right place.

Or something like that.