Jason Qiu

2

1. Basic user interface is completed.

   The locator-pane with the background of the map as well as the ability to place pairs of locators is implemented.

   The user will also be able to enter the addresses of their starting point and destinations directly.

   DEADLINE: April 7 (COMPLETED)

2. Mathematica will be able to find and navigate roads on a map and store it in some way.

   DEADLINE: April 17

3. The ability to calculate the most efficient path between two points on the map while following roads is developed.

   DEADLINE: April 19

4. The program will be able to estimate arrival times, while taking speed limits of roads into account.

   DEADLINE: April 22

5. The function to check whether or not all of the routes given their respective start times are possible for a given number of drivers is developed.

   DEADLINE: April 28

6. Each route will be able to have given number of passengers, and drivers' cars will be able to have given amount of passenger space, while keeping the previous milestone's function intact.

   DEADLINE: If we have time.

3.

We learned that the built-in image processing functions in Mathematica could be the key for our project.

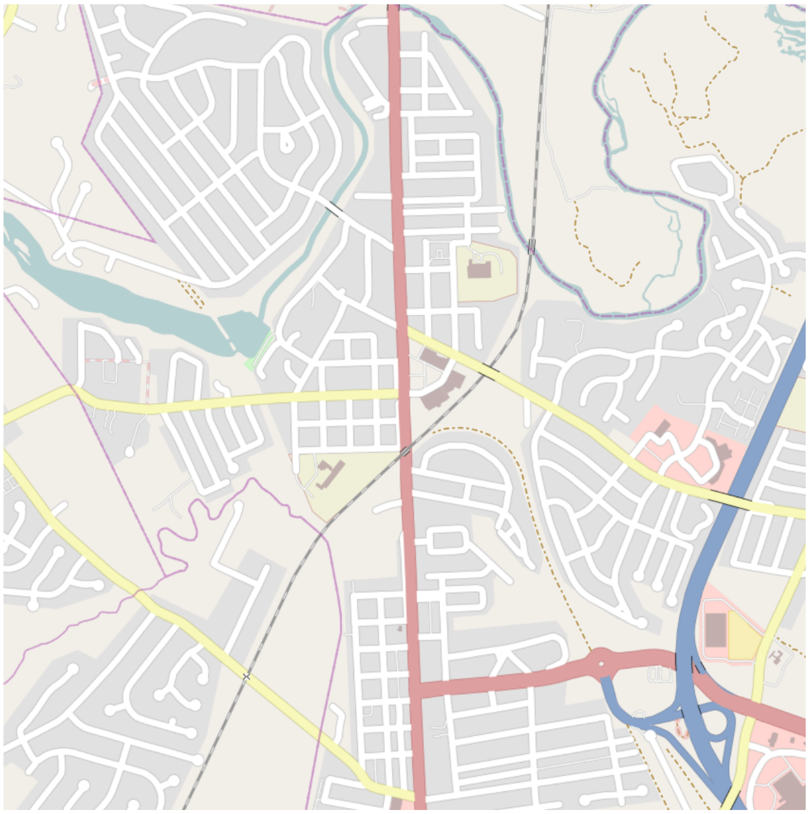
We have managed to create a draggable, zoomable, using in-built Mathematica functions. We can also easily find locations on that map using street addresses as input.

```
In[28]:= (* finds the points on the map that correspond to the given addresses *)
         pt1 = Interpreter["StreetAddress"]["1509 College Heights Blvd"];
               解释器
         pt2 = Interpreter["StreetAddress"]["1201 Morgantown Rd, Bowling Green, Kentucky"];
               解释器                                                    绿色
         pts = {pt1, pt2};

         (* creates a dynamic map that is draggable
          and zoomable with given points marked on the map *)
         DynamicGeoGraphics[GeoMarker[pts], GeoRange → Quantity[1, "Miles"], GeoZoomLevel → 15]
         动态地理图形          地理标记          地理范围    数量                  地理放大级别

Out[31]=
```

We also figured out how to change the map's style so that it contains no labels, as well as how to export and import it.

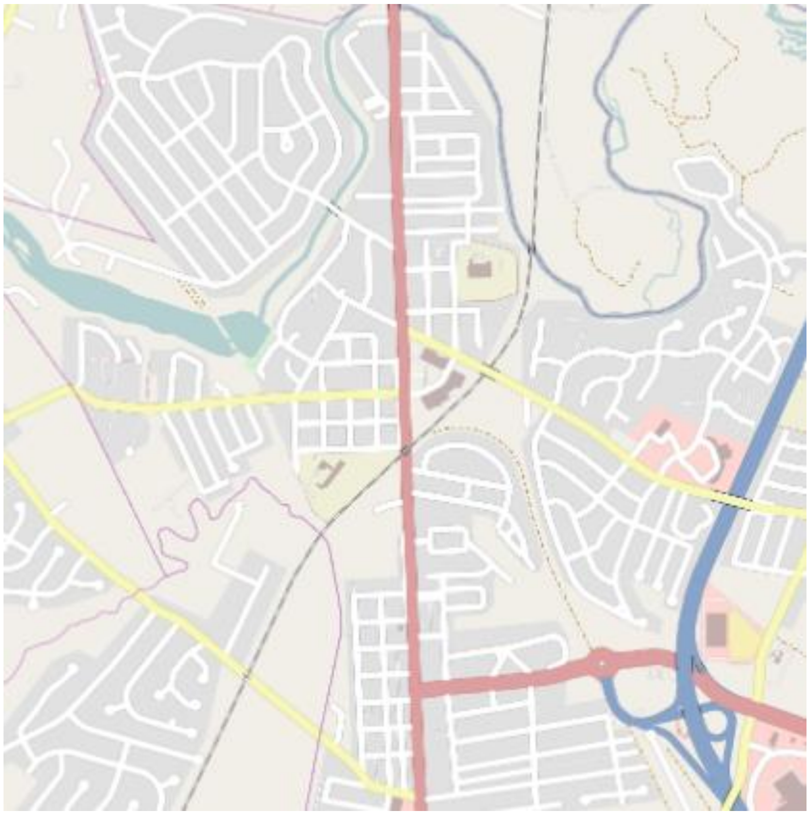
```
In[5]:= (* creates a map centered at a point using a different tile server,
        so that it contains no labels of any kind *)
        map = GeoGraphics[GeoCenter → pt1, GeoRange → Quantity[1, "Miles"], GeoZoomLevel → 15,
              地理图形    地理中心           地理范围    数量                  地理放大级别
          GeoServer → {"https://tiles.wmflabs.org/osm-no-labels/``/``/``.png"}]
          地理服务器

        (* sets the current directory, so that Mathematica knows where to export and import *)
        SetDirectory[NotebookDirectory[]]
        设置目录        当前笔记本的目录
        Export["map.jpg", map]
        导出
        Import["map.jpg"]
        导入
```

Out[5]=



```
Out[6]= F:\School\Gatton Academy\2019 Spring Term\MATH-CS 371\FinalProject

Out[7]= map.jpg
```

Out[8]=

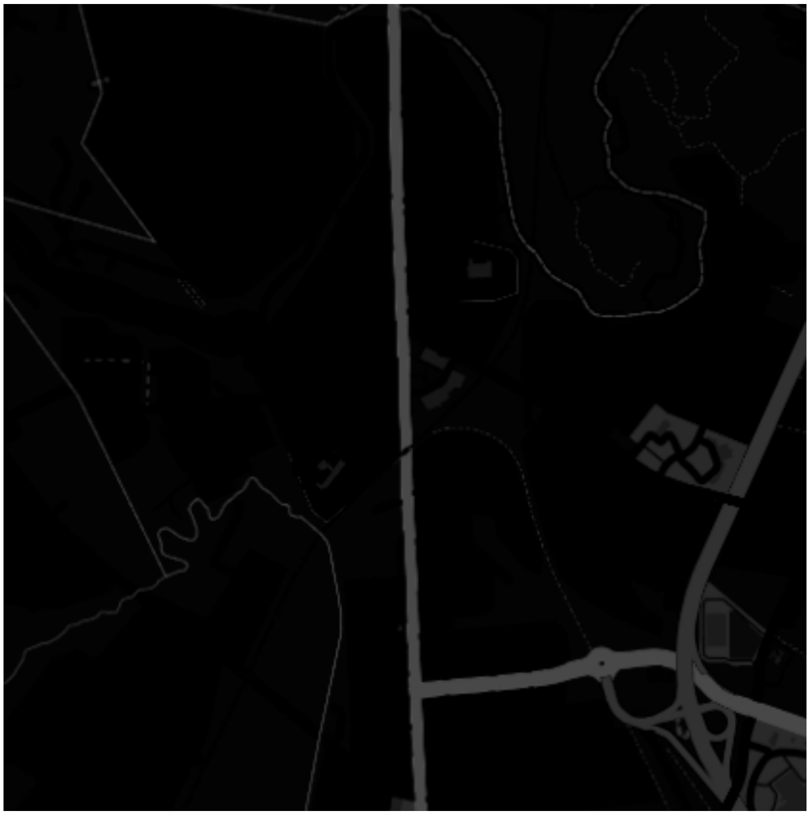


We have managed to extract some types of roads out of the map using a combination of in-built Mathematica functions, so that they can be more easily processed later on.

In[9]:=

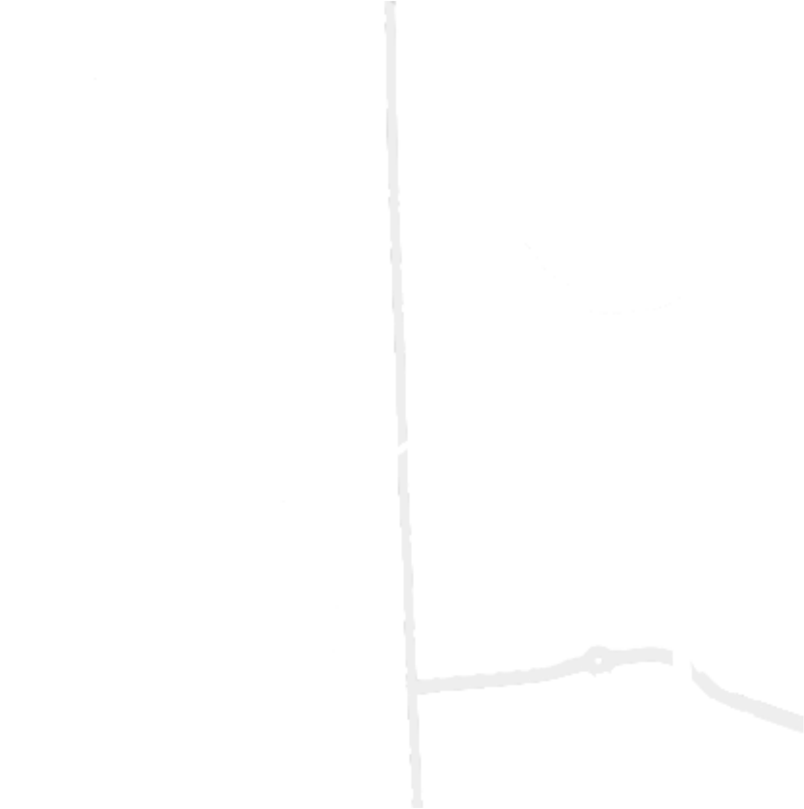
```
(* create a b/w image of the map,
with lighter corresponding to colors closer to magenta *)
m = ColorSeparate[map, "M"]
     分离颜色


(* negate the color in the image and then multiply the image on itself
 by a small factor to remove extraneous details and improve visibility *)
Print["Magenta roads isolated: "];
打印     品红色
ImageMultiply[ColorNegate[m], 1.3]
图像相乘          图像|彩色负片
```

Out[9]=



Magenta roads isolated:

Out[11]=



First of all, we still need to figure out how to extract the other types of roads. After that, we have to solve the problem of downsizing the image so that it can be more quickly processed. Next, we must determine how to store the path’s data as well as finding the paths themselves (while accounting for small gaps). Following that, we have to figure out how to find the most efficient path from one point to another using the given paths. Finally, we need to take that path and display it on a user-friendly map so that it can be easily seen.

4.

The largest issue we have is finding out how to navigate and store the roads on the maps. We will look further into Graphs and see if we can come up with anything.