CS 205 **Lecture Worksheet**
April 12, 2016

Using Google Maps

Last class, we created a timetable of all of the buses that are about to arrive at the Illini Union. Today, we will take it a step further and add the buses to a Google Map.

Google Maps provides a JavaScript API that allows programmers to interact with their Maps software:
https://developers.google.com/maps/documentation/javascript/

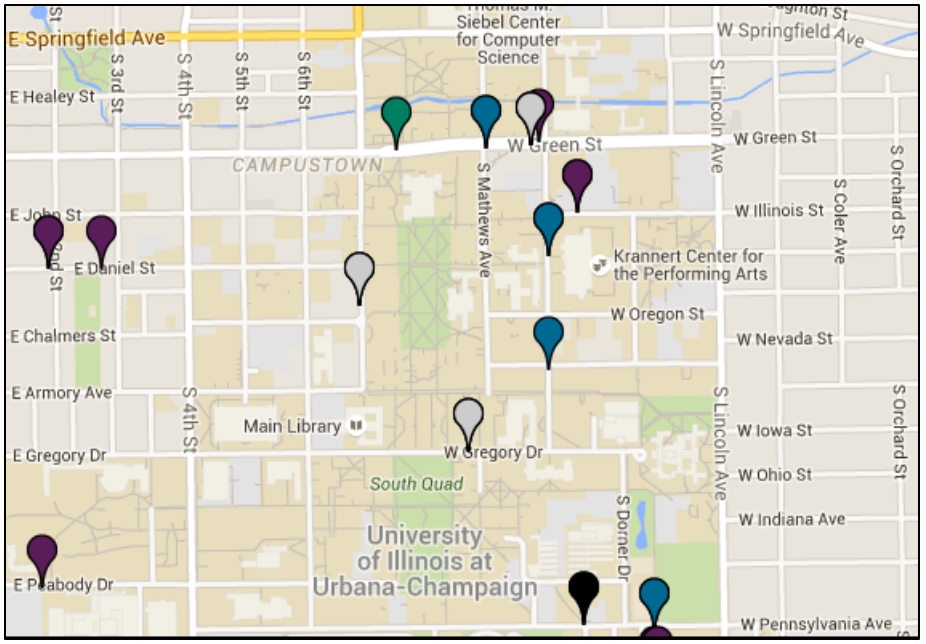
From the "Markers" example on the page linked above, the JavaScript code to add a map only requires the following code (modified for C-U location:

```
1  // Create the map object
2  var mapOptions = {
3    center: { lat: 40.108564, lng: -88.227134},
4    zoom: 15
5  };
6
7  var map = new google.maps.Map(
8    document.getElementById("gmaps"),
9    mapOptions
10 );
11
12
13 // Create a marker and set its position
14 var loc = { lat: 40.108564, lng: -88.227134};
15
16 var marker = new google.maps.Marker({
17   position: loc,
18   map: map,
19   title: "Hello, world!"
20 });
```

In order to place a marker where the buses are currently located, we need to save the current latitude and longitude of the buses from the CUMTD API. *…while we're at it, save a color that is appropriate for the route so we can make colored markers later.*

Puzzle #1:

Modify the Python code in **exp_cumtd_gmaps** to save a "lat", "lon" and "route_color" for every bus that was returned from the CUMTD API. You can look at the format of the data from CUMTD in the res/cumtd_raw.json file.



Adding Google Maps

Using the code from the first section of this worksheet, get Google Maps to display within your Workbook:

1. Add Lines 1-10 (from the right), this creates a Google Map centered on The University of Illinois with no markers. **Run it, test it!**
2. Add Lines 13-20, this adds a single marker to the center of The University of Illinois. **Run it, test it!**
3. Finally, add a for-loop around the code to add the marker to add one marker for each route. **Run it, test it!**

To add color to each marker, add the following dictionary entry to the options you program is passing to the Marker class (Lines 16-20):

```
icon: "http://www.googlemapsmarkers.com/v1/" +
      d.route_color + "/"
```

…where ***d.route_color*** *is the color for a specific route (without #).*

Puzzle #2:

Modify the JavaScript code to create the map printed at the top of this page. This will require looping over your data gathered from the Python.

Drawing Shapes for Routes

In addition to Markers, Google allows for Polylines to be drawn on a map – a feature we can use to draw on top of roads to indicate the route of a bus.

Looking at the CUMTD API (https://developer.cumtd.com/), what API calls can we make to get the shape of the route?

What additional information do we need to get from **`GetDeparturesByStop`** in order to make a call to this new API call?

Puzzle #3:

In Python, use a new CUMTD API call to get the route information for each of the buses in our dataset. Your new JSON should contain all of the information about the route **plus** the shape of the route.

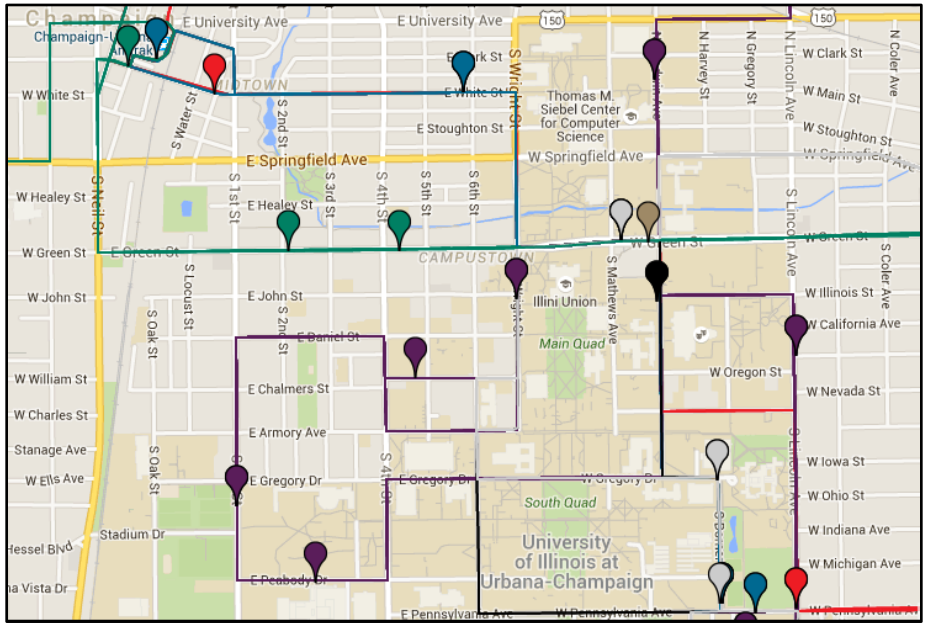
Using Google Maps Documentation to Add a Polyline

When using APIs, you will rely on the developer's documentation and examples in order to understand their APIs.

Google has provided some great examples of using a Polyline on a Google Map, which can be viewed here:
https://developers.google.com/maps/documentation/javascript/shapes

Puzzle #4:

In JavaScript, add a Polyline for the route of each bus in our dataset to the Google Map.



Final visualization, with bus markers and bus paths on a Google Map.