Creative Coding Part 2 – Report

Patrick Bondaruk – N00201161

One of the approaches that I made is using a map(); function and the map(); function in p5.js is used to normalize a number having range from min1 to max1 in a range of min2 to max2. So the function would look something like this,

let newValue = map(value, 0, 100, 0, 80);

So in this example it would scale the ‘value’ from the number between 0 to 100, to a range of 0 and 80.

I used this function for the vertical bar chart to scale the data, so the function would scale the number from the number between 0 to maxValue, to a range of 0 and chartHeight. So the function would look something like this,

ScaleData(num) { return map(num, 0, this.maxValue, 0, this.chartHeight); }

Similarly I used the same function for a scatter plot chart, and since I want to scale both the x-axis and the y-axis, I made separate functions for scaling data, one for the x-axis and one for the y-axis.

Another approach that I took which is probably one of the most common approaches that I took since it is so useful is the translate. The translate basically translates the x and y coordinates to the new origin position from the Canvas. For example if the sketch looked like this,

translate(30, 70);  
rect(0, 0, 50, 50);

the rectangle would start drawing from the new origin position. So instead of the starting position of 0, 0 it would actually start drawing from 30, 70.

The other function that would along with the translate is the push() and pop(). The push() and pop() function is the transformation, which the pop() that undoes the translations that are made following the previous push().

One of the other useful approaches that I took and also essential for the charts is the max() function. The max() takes the largest number from the list or the data, and is being assigned to a variable. In this case if I want to list the values in the data and I want to take the largest number, it would look something like this.

let listValues = this.data.map(function(x) { return x.values })  
this.maxValue = max(listValues);

It creates a variable ‘listValues’ and assigns it to the data to map and return all the values that are available and then uses the map() function to take the largest number from all the values.

The other useful approach that I took to certain data is the rotate with PI. The rotate basically rotates the values to a certain amount that is assigned, and the common element to use PI. The PI equals to 180 degrees, so if there is a certain value that I want to rotate by 90 degrees it would be rotate(PI / 2), and if there would be a certain value that I want to rotate by 45 degrees it would be rotate(PI / 4), and there would be a certain value that I want to rotate by 270 degrees it would be rotate(PI \* 1.5). So the PI is useful for rotating values or objects.