Assignment 4

Approach followed

- I first create a keyboard layout from the qwerty_layout in the file as in the keyboard heatmap programming quiz.
- The keyboard layout is generated using matplotlib rectangles in thhe function `genKeyboardLayout`
- My heatmap is a pixel grid of 58 * 16 pixels in x, y coordinates.
- Takes in an input string from the user.
- I then generate two arrays x, y each containing x, y coordinate of each key pressed for example:
  - if i press a which has coordinates (1.75, 2) 1.75 gets appended to x, and 2 gets appended to y
  - This keeps track of the frequencies with which each coordinate in the keyboard has been pressed.
- I then call the `calculate_key_travel` function which calculates the total distance travelled for pressing the entire text. it follows the following logic -
  - for each character c in the `input_string` it checks each key to be pressed, and adds the distance to each of those keys onto sum
  - it does this byy going to the `characters` dictionary, which contains a tuple consisting of each key to be pressed for achieving a particular character.
  - For eg, say we want to type A, the `characters` dictionary will have a key `A` with the element `('Shift_R', 'a')`, so it adds the distance for going to both shift_R and a.
  - returns sum
- if -a flag is passed using the command line, generates animation, else saves heatmap.png
- -a flags creates animation, although it takes time for longer text samples.
- -a calls the plot function (for animation) and otherwise calls the plot1 function (without animatoin)
- The plot function works as follows
  1. Consists of heatmap array, containing an element for each pixel.
  2. Contains X, Y arrays which are 2d arrays constructed using `np.meshgrip`
  3. For each letter, i update corresponding values in the heatmap array, (inside a circle of radius 0.6 units) multiplied with a function that has

decreasing values as it goes further away from the center of the key, to create a gradient effect.

4. I define a custom colormap for the heatmap from blue to red and use `plt.imshow` function to ccreate the heatmap and save it to `heatmap.png`
5. If i use the `plot` function with the -a flag for animation, i create a function update, which gradually increments the heatmap array, updates the artist , used as an arguement to `FuncAnimation` function from numpy, and saves it to `animation.gif`

The QWERTY LAYOUT FORMAT

- I follow the same layout format given in the programming quiz, the layout has two dictionaries :
  1. `keys` which contains a key for each key in the dictionary, and the value is a dictionary of the format `{'pos': (x,y), 'start':'home_row_key'}` where the pos is the coordinate of the given key and `start` is the home row key to be used while typing that key
  2. `characters` which contains a character as key in the dictionary , and each character corresponds to a tuple of individual keys that have to be pressed for typing that key for example. to obtain A, we have to type both `Shift_R` and `a`

Results

Sample Text 1 :

```
The environment is a vital part of our planet, providing the resources we need to live and t
```

Distance travelled for qwerty layout = 483.68 units

Heatmap

Distance travelled for dvorak layout = 295.28 units

Heatmap

Distance travelled for colemak layout = 208.830 units

Heatmap

Sample text 2 :

```
Programming is the art of crafting solutions through code, transforming ideas into functiona
```

Distance travelled for qwerty layout = 506.98 units

Heatmap

Distance travelled for dvorak layout = 373.016 units

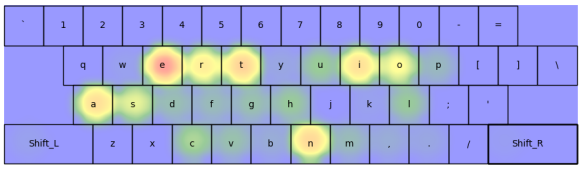

Figure 1: Heatmap

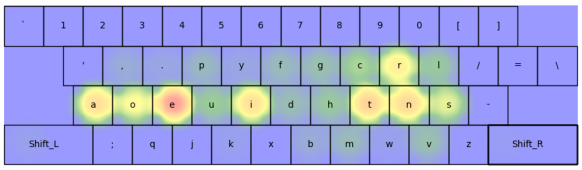

Figure 2: Heatmap

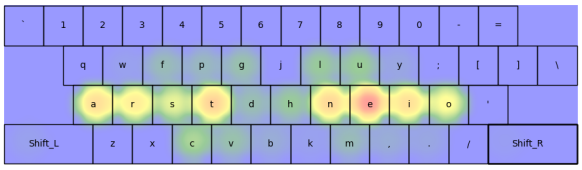

Figure 3: Heatmap

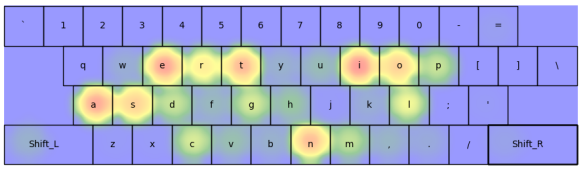

Figure 4: Heatmap

Heatmap

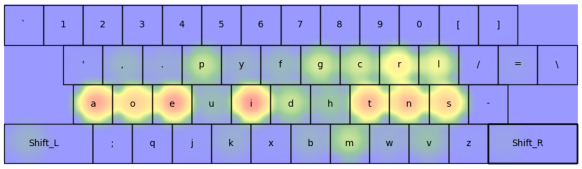

Figure 5: Heatmap

Distance travelled for colemak layout = 269.971 units

Heatmap

Steps to test code with different layouts

All the layout files have to be **imported** eg `import qwerty_layout` or `import dvorak_layout`

all the layouts have been uploaded in the zip file.

in the `main` block, update the layout used in the line `layout = qwerty_layout`

Notes

- All the `layout.py` has to be in the same directory as the script.py, if default analysis keyboard needs to be changed, change the `layout` to the required `layout`

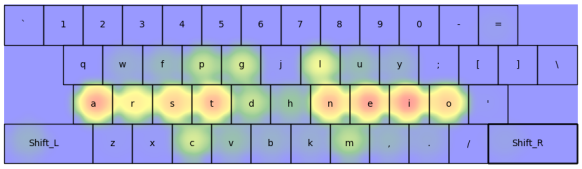

Figure 6: Heatmap