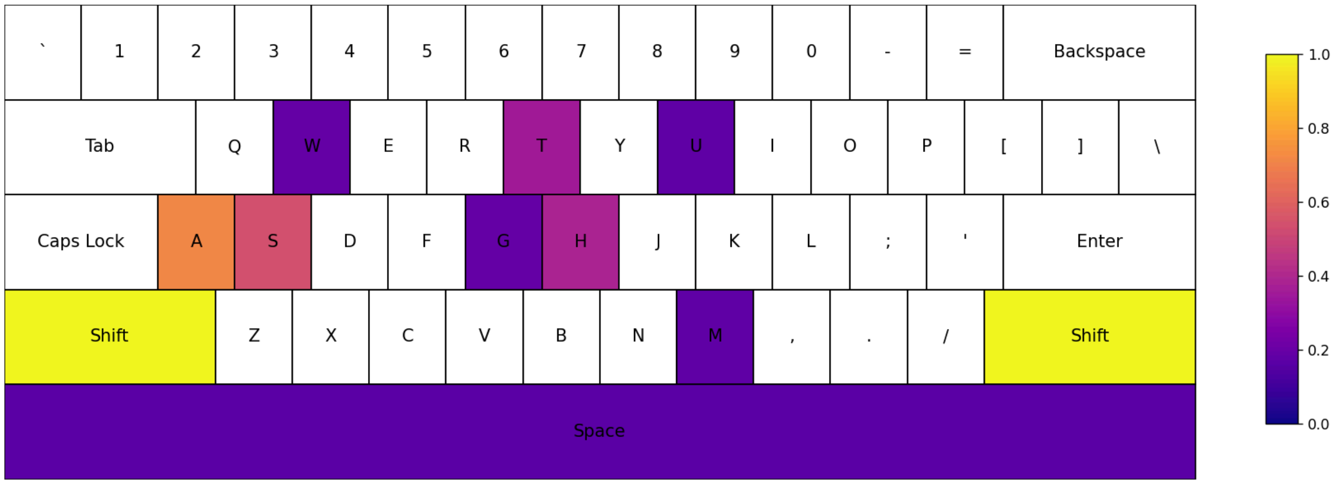
ASSIGNMENT 4



SHASHWAT GAUTAM

EE23B150

MAIN ASSUMPTION AND ADDED FUNCTIONALITY:

- My assignment is based on the First Problem Statement as I was unable to do the Programming quiz on keyboard since I was hospitalized.
- I have assumed that if any Uppercase character is followed by lower case character the user has used the shift key
- I have assumed travel distance for SHIFT_R and SHIFT_L is same
- I have assumed if any uppercase character is followed by another uppercase character then the user has used the caps lock key
- I have assumed that to write any uppercase character let's say “S” the user press “Shift” then “S” while writing “SS” the user press “Capslock on” then “S” then “Capslock off” then “Capslock” on then S then “Capslock off”
- I have divided the key_counts by max_count so that my color bar represent between 0 and 1
- I have added the functionality:
    1. Animation
    2. Custom Layout : The user can input the custom layout via .txt but it should be in a specific format I have attached the file in zip for reference
    3. The color of the first character of input string will differ a bit from rest of the character color even if all have same frequency this is to make the first character stand out in the string.

LOGIC SUMMARY

- ## <u>DEFAULT KEYBOARD LAYOUT:</u>

  I have used the QWERTY keyboard as my default keyboard and used try and error to get size of special keys like Caps lock, shift , space etc

- ## <u>STORING KEY COORDINATES:</u>

  I m storing the key coordinates in the dictionary named **“key_cords”** here the layout whether default or custom is a list of lists therefore I used enumerate, initialized x-coordinate and loop through each key in the row.

For special keys I defined a separate dictionary **“key_sizes”** which contains the width of these keys if these key are pressed in input_string we use them else the default size is 1 unit at the end I just update the x coordinate for the next key

- ## CALCULATING EUCLIDEAN DISTANCE BETWEEN THE KEYS:

Used the euclidean formula to calculate the distance and then multiplied by two because to type let say “Sh” finger goes to Shift , other finger goes to S and third goes to h and final all return back to home row

- ## COUNTING KEYPRESSES AND CALCULATING DISTANCE:

This code aims to count key presses and calculate the total Euclidean distance traveled by fingers on a keyboard while typing an **`input_string`**

This function handles normal key press and various possible cases of special keys like Shift Caps Lock , Space given as comment in code

The function returns key_counts and total_distance

- ## DEFINING MAIN FUNCTION:

I m asking the user whether he want to give custom layout or continue with default QWERTY layout inside main , I have taken input string in main itself and define the two important functions namely **“draw_keyboard_with_heatmap()”** and **“animate”** while other function are called and imported library are used here

- ## PLOTTING KEYBOARD AND HEATMAP:

I have defined the **draw_keyboard_with_heatmap()** function to draw a keyboard layout using matplotlib. It handles the placement of keys in rows and sets their respective sizes, positions, and labels. For key placement the inner for loop iterate over each each in the current row and the function manages both the position (using x_start and y_pos) and size (using width and key_height) of each key.Then I have created Rectangle object for each key using matplotlib.patches.Rectangle. The rectangle's position is determined by the (x_start, y_pos) coordinates, and its size is set by width and key_height. Then i used The ax.text() method to convert the rectangles into key i.e placing the char on top of each rectangle.After drawing each rectangle,I have appended the rectangles list along with its corresponding key label (rectangles.append((rect, key))) , this serves the purpose of later applying heatmap colors and other visual effects to specific keys based on frequency or intensity.

- ## ADDING ANIMATION:

I have defined the function **animate(i)** to animate the intensity of key usage over time on keyboard layout and have used matplotlib to update the color of keys based on how many times they are pressed, with specific logic for the Shift and caps lock in addition the logic starts with key press detection followed by shift key and caps lock logic which has been given as comment in code itself then there is the section that updates the heatmap for key presses The function animation.FuncAnimation() is used to create the animation. It calls the animate(i) function for each frame, where i represents the index of the current character being processed in input_string.

RUNNING THE CODE:

1. Unzip the file it contains three files namely:
    a. README.pdf
    b. EE23B150.ipynb
    c. custom_layout.py
2. Open the Jupter and upload EE23B150.ipynb and custom_layout.py
3. Run theEE23B150.ipynb like a normal python note as mentioned above the custom_layout.py is simply an example of layout which can be given as custom_layout.