Name – **ARPIT TYAGI**

Roll no- **202401100400050**

Branch – **CSE (AI & ML)**

Project –Traffic light control system

1. Introduction

Traffic lights are an essential part of modern road infrastructure, helping manage traffic flow and enhance safety. A Traffic Light Control System automates the switching of lights (Red, Yellow, Green) to regulate vehicle and pedestrian movement at intersections. This project simulates a basic traffic light control system using Python. The system follows predefined timing rules to cycle through the lights, mimicking real-world traffic signals.

2. Methodology

Our approach involves:

1. Defining Light States: Red, Yellow, and Green with appropriate durations.
2. Simulation using Python: Using loops and delays to cycle through the lights.
3. Graphical Visualization: Using matplotlib or Tkinter to display the traffic lights dynamically.

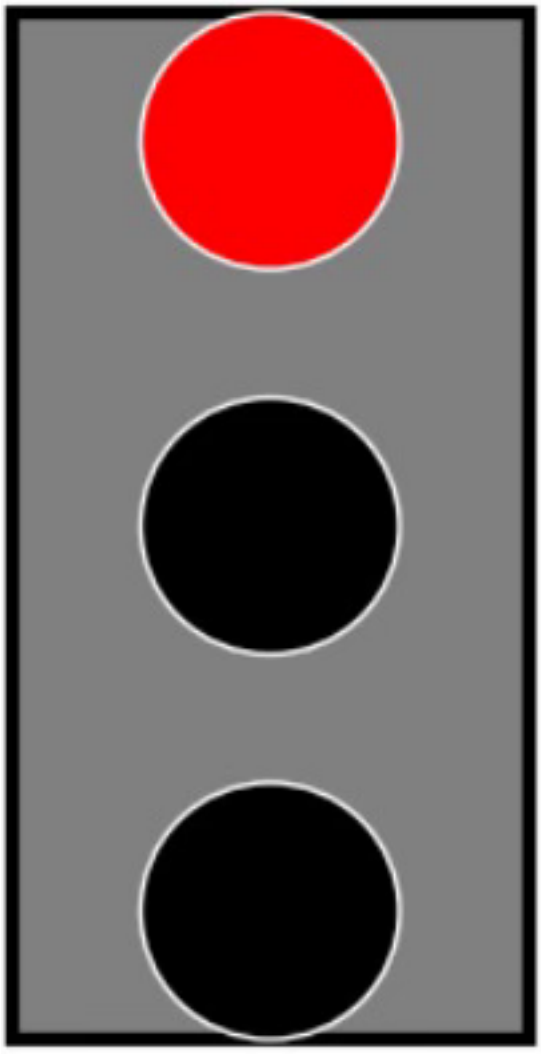
3. Code

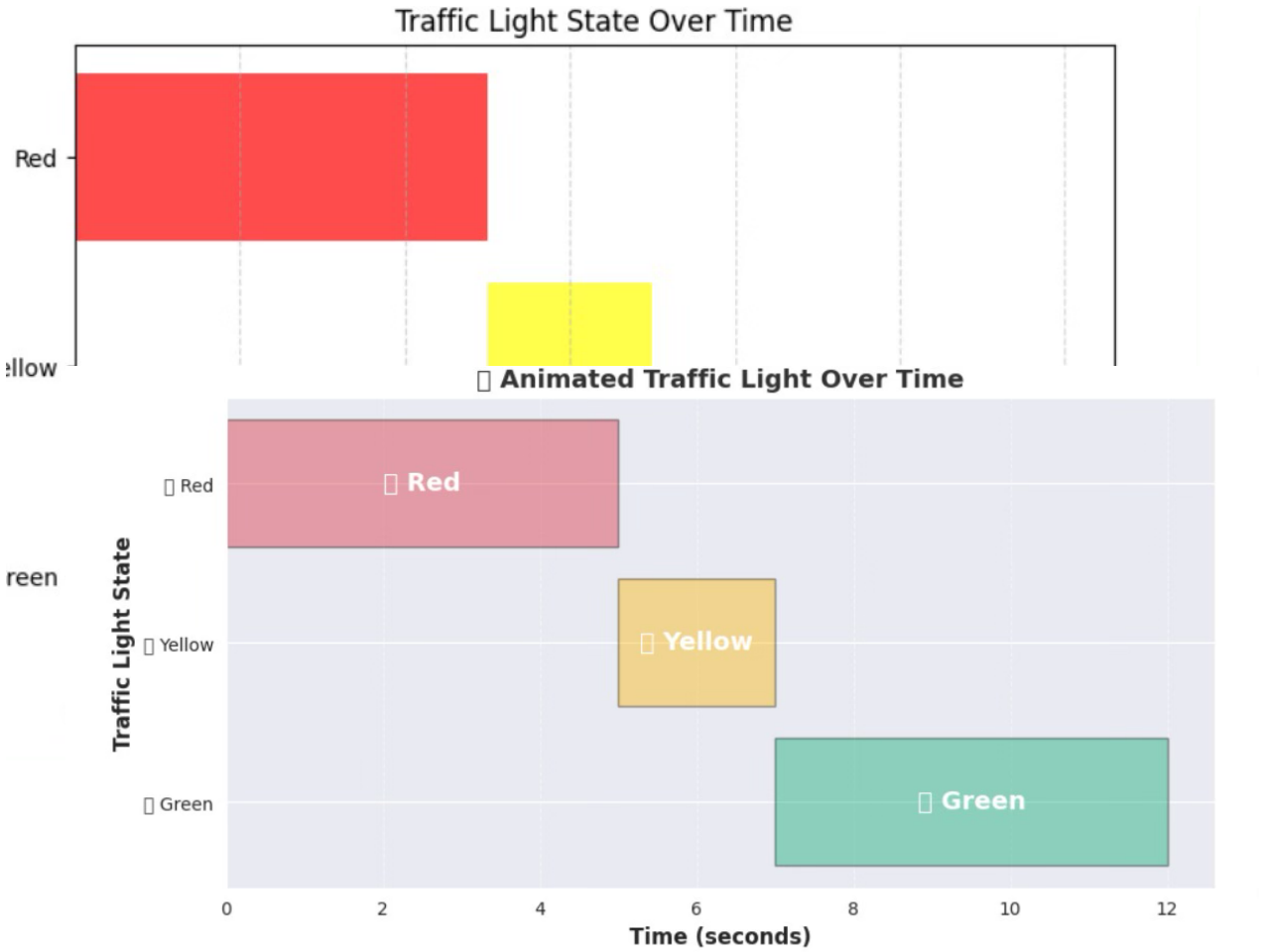
```
import time
import matplotlib.pyplot as plt
import matplotlib.patches as patches
from IPython.display import display, clear_output
```

```
# Function to draw the traffic light
def draw_traffic_light(color):
    """
    Draws a traffic light with the given active color.

    Parameters:
    color (str): The color of the active light (red, yellow, or green).
    """

    # Create a figure and axis for the plot
    fig, ax = plt.subplots(figsize=(3, 6))
    ax.set_xlim(0, 3)
    ax.set_ylim(0, 6)
    ax.set_xticks([])
    ax.set_yticks([])
    ax.set_frame_on(False)  # Remove frame for a cleaner look

    # Draw the traffic light box (outer rectangle)
    light_box = patches.Rectangle((0.5, 1), 2, 4, edgecolor='black', facecolor='gray', linewidth=3)
    ax.add_patch(light_box)

    # Define positions for the three lights (Red, Yellow, Green)
    positions = {'red': 4.5, 'yellow': 3, 'green': 1.5}

    # Draw all three lights, highlighting the active one
    for light, y_pos in positions.items():
        facecolor = color if light == color else 'black'  # Only the active light is colored
        ax.add_patch(patches.Circle((1.5, y_pos), 0.5, edgecolor='white', facecolor=facecolor))

    # Clear previous output in Google Colab and display the new figure
    clear_output(wait=True)
    display(fig)
```

```
    plt.close(fig)  # Close the figure to avoid memory issues


# Infinite loop to simulate traffic light sequence
while True:
    # Traffic light sequence: Red → Yellow → Green
    for color, duration in [('red', 5), ('yellow', 2), ('green', 5)]:
        draw_traffic_light(color)  # Display the active light
        time.sleep(duration)  # Pause for the duration of each light
```

OUTPUT:





Special credit to Bikki sir