Algorithm for Traffic Light Simulation using Pygame

Step 1
Initialize the Pygame library and create a window for the simulation.
- Set the screen size to (800, 600).
- Set the window title to 'Traffic Light'.

Step 2
Define required colors:
- Black for the background.
- Grey for the traffic light box.
- Red, Orange, and Green for traffic lights, along with their dimmed versions.

Step 3
Initialize light properties:
- Store each light's position, radius, active color, and dimmed color in a dictionary.
- Create a list containing all light dictionaries.
- Define `current_light_index = 0` to track which light is currently active.
- Set `last_switch_time` using `pygame.time.get_ticks()`.
- Define `switch_interval = 2000 ms` (2 seconds) for changing lights.

Step 4
Start the main event loop:
- Check for user events:
  - If the **QUIT** event is detected, exit the loop.

Step 5
Update the traffic light state:
- Compare the current time with `last_switch_time`.
- If the time difference is greater than `switch_interval`:
  - Update `current_light_index` to switch to the next light in the list.
  - Reset `last_switch_time` to the current time.

Step 6
Render the graphics:
- Fill the background with **black**.
- Draw the **grey traffic light box** with rounded edges.
- Loop through all lights and:
  - Draw the **active** light in its bright color.
  - Draw **inactive** lights in their dimmed color.
- Refresh the display using `pygame.display.flip()`.

Step 7

Exit the simulation when the loop terminates.
- Quit Pygame.