PSEUDOCODE

```
CLASS HistogramApp
    FUNCTION __init__(traffic_data, date)
        SET self.traffic_data = traffic_data
        SET self.date = date
        INITIALIZE Tkinter window
        INITIALIZE vehicle count lists for two junctions

    FUNCTION setup_window()
        CREATE a canvas for drawing the histogram
        RETURN the canvas

    FUNCTION vehicle_perHr()
        TRY
            OPEN the CSV file with traffic data
            READ the data using a CSV reader
            FOR each row in the data
                EXTRACT hour from timeOfDay
                IF JunctionName is "Elm Avenue/Rabbit Road"
                    INCREMENT count for Elm junction
                ELSE IF JunctionName is "Hanley Highway/Westway"
                    INCREMENT count for Hanley junction
            END FOR
        EXCEPT handle errors

    FUNCTION draw_histogram()
```

```
        DEFINE bar width, padding, bottom margin, and max height
        EXTRACT hours for plotting
        CALCULATE scale for bar heights based on max traffic
        FOR each hour from 0 to 23
            CALCULATE bar positions
            DRAW bars for both junctions
            ADD hour labels and values on top of the bars
        END FOR
        ADD axis labels

    FUNCTION add_legend()
        DRAW legend for the histogram

    FUNCTION run()
        CALL setup_window()
        CALL vehicle_perHr()
        CALL draw_histogram()
        CALL add_legend()
        START Tkinter main loop

CLASS MultiCSVProcessor
    FUNCTION __init__()
        INITIALIZE current data

    FUNCTION validate_date_input()
        PROMPT user for date, month, and year
        VALIDATE inputs and return formatted date
```

```
FUNCTION load_csv_file(valid)
    SET database variable based on valid date
    RETURN the corresponding CSV file name

FUNCTION clear_previous_data()
    CLEAR console output

FUNCTION handle_user_interaction()
    PROMPT user to select another date or exit
    RETURN user's choice

FUNCTION process_files(database)
    TRY
        OPEN the selected CSV file
        INITIALIZE counters for various vehicle types
        FOR each row in the data
            UPDATE counters based on conditions
        END FOR
        STORE results in an output list
        RETURN the output list
    EXCEPT handle file errors

FUNCTION display_outcomes(output)
    PRINT the results from the output list

FUNCTION save_results_to_file(output)
```

```
        OPEN results file in append mode
        WRITE the results to the file

FUNCTION main()
    CREATE instance of MultiCSVProcessor
    WHILE True
        TRY
            CALL validate_date_input() and store result
            CALL load_csv_file(valid) and store result
            CALL process_files(database) and store result
            CALL display_outcomes(output)
            CALL save_results_to_file(output)
            CREATE instance of HistogramApp with database and valid date
            CALL run() on HistogramApp instance
            IF handle_user_interaction() is True
                BREAK the loop
        EXCEPT handle errors

CALL main()
```