Coursework - Software Development I (4COSC006C - Programming)

Name : D. W. Yeheni Kaveesha Dodanwela
Student No : 20241486 (IIT) | w2121364 (UoW)
Date : 23/12/2024

Design (Pseudocode) for Part A,B,C

```
Begin program part_ABC

Function is_leap_year (year):
    If year is divisible by 4:
        If year is divisible by 100:
            If year is divisible by 400:
                Return True
            Else:
                Return False
        Else:
            Return True
    Else:
        Return False

Function validate_date_input():
    Repeat
        Print "Enter the day (DD):"
        day = Get user input
        If day is not a number or day is out of range (1-31):
            Print "Invalid day. Please enter a valid day."
        Endif
    Until day is valid

    Repeat
        Print "Enter the month (MM):"
        month = Get user input
        If month is not a number or month is out of range (1-12):
            Print "Invalid month. Please enter a valid month."
        Endif
  Until month is valid
```

```
    If month is February:
        If is_leap_year (year):
            If day > 29:
                Print "Invalid day for February in a leap year."
            Endif
        Else:
            If day > 28:
                Print "Invalid day for February."
            Endif
    Else if month has 30 days:
        If day > 30:
            Print "Invalid day for this month."
        End if
    Else if month has 31 days:
        If day > 31:
            Print "Invalid day for this month."
        End if
    End if

    Repeat
        Print "Enter the year (YYYY):"
        year = Get user input
        If year is not within valid range (2000 - 2024):
            Print "Invalid year. Please enter a year between 2000 and 2024."
        Endif
  Until year is valid

    Return formatted date string (day + month + year)

Function validate_continue_input():
    Print "Do you want to process another file? (Y/N):"
    response = Get user input
    IF response is "Y" or "y":
        Return True
    Elseif response is "N" or "n":
        Print "End of run"
        Return False
    Else:
        Print "Invalid input. Please enter 'Y' or 'N'."
        Return validate_continue_input()
```

```
Function process_csv_data(file_path):
    Open file at file_path
    Read header of CSV file
    Find column indices for relevant data (eg: VehicleType, elctricHybrid, etc.)

    Initialize counters for total vehicles, electric vehicles, trucks, etc.
    Set all counters to 0

    For each row in CSV file:
        Extract relevant data (eg: vehicle type, speed, weather condition)
        Increment total vehicle count
        Update various counters based on vehicle types
        Update junction-specific counts
        Track rain hours
        Track peak traffic hours
     End for

    Calculate truck percentage
    Calculate scooter percentage
    Find peak traffic hours

    Create list with all calculated statistics
    Return results and filename

Catch FileNotFoundError
     Display “File not found”
     Return nothing

Function save_results_to_file(outcomes):
  If outcomes exist then
    Open file "results.txt" in append mode
    Write header to file (eg: dataset processed)
    Write each outcome from outcomes to file
    Write separator line after outcomes
    Print "Results saved to file."
  End if
```

```
Main program
    Set continue_program to True
    While continue_program is True
        Get valid date from user
        Create filename using date
        Process csv data
        If data was processed successfully Then
            Display results
            Save results to file
        End if
        Ask user to continue
        Update continue_program based on user input
    End while
End program
```

Design (Pseudocode) for Part D,E

```
Begin program part_DE

Class HistogramApp
  Function initialize (traffic_data, date)
    Store traffic data and date
    Create main window
    Setup window display
    Draw histogram
    Add legend

  Function setup_window()
    Set window title to "Histogram"
    Set window size to 1500x500
    Create white canvas
    Add canvas to window


  Function draw_histogram()
    Set margins and spacing
    Calculate maximum value from data
```

```
Draw x-axis
        Add title
        Add x-axis label

        For each hour in traffic data
            Calculate bar positions
            Draw blue bar for first junction
            Add count label
            Draw green bar for second junction
            Add count label
            Add hour label
        End for

    Function add_legend()
        Set legend position to top - right

        Draw blue box and label for first junction
        Draw green box and label for second junction

    Function run()
        Start main window loop
End Class

Class MultiCSVProcessor
    Function initialize()
        Set current_data to nothing
        Set selected_date to nothing

    Function load_csv_file(filepath)
        Create data structure for junction counts
        Open and read CSV file
        For each row in file
            Get hour from time
            Count vehicles for each junction
        End for
        Store processed data
        return data and date

    Function clear_previous_data()
        Reset current data and date
```

```
    Function handle_user_interaction()
        Get valid date from user
        Create filename
        Process csv data
        If processing successful then
            Display results
            Save results
            Create and show histogram
            Wait for histogram to close
        End if
        Return user's continue choice

    Function process_files()
        Repeat
            Process one file
            Clear data
        Until user chooses to stop
End class

Main program
    Create MultiCSVProcessor
    Start processing files
End program
```