**Report for the Assignment:**

**Program Requirements:**

To fulfill the program requirements, the following data requirements, program control structures, and processing requirements need to be implemented in the program:

**Data Requirements:**

* Basic Price of the vehicle (numeric)
* Trade-in allowance (numeric)
* Accessories (String):
* Stereo System
* Leather Interior
* GPS
* Exterior Finish (string):
* Standard
* Modified
* Customized
* Price of additional accessories (numeric)

Sales Tax Rate (numeric)

**Program Control Structure:**

1. Input Devices (to obtain user input for data requirements)
2. Validation of user input (to ensure input data is valid)
3. Conditional Statements (to check if the user wants to add accessories)
4. Looping statements (to calculate the total price of additional accessories)
5. Event-Driven Programming Components:
6. Calculation button (to calculate the total amount due)
7. Clear button (to clear all input fields)
8. Exit button (to exit the program)
9. Reset button (to reset all fields to default values)

**Processing Requirements:**

1. Obtain user input for basic price, trade-in allowance, accessories, and exterior finish
2. Validate the user input to ensure that it is valid (e.g., numeric values are entered for price and allowance, Boolean values are selected for accessories, and a valid finish option is selected)
3. Calculate the total price of additional accessories based on the user's selection
4. Calculate the subtotal by adding the basic price and the total price of additional accessories
5. Calculate the sales tax on the subtotal
6. Calculate the amount due by subtracting the trade-in allowance from the subtotal
7. Display the subtotal, sales tax, and amount due on the output devices
8. Provide event-driven programming components for Calculation, Clear, Exit and Reset.

**PsuedoCode:**

BEGIN PROGRAM

SET trade in allowance = 0

GET basic price FROM USER

GET trade in FROM USER

IF trade in is not a number THEN

DISPLAY "Invalid trade-in allowance. Please enter a number."

GET trade in FROM USER

END IF

GET accessories FROM USER

SET total accessories price = 0

IF accessories includes Stereo System THEN

ADD stereo system price to total accessories price

END IF

IF accessories includes Leather Interior THEN

ADD leather interior price to total accessories price

END IF

IF accessories includes GPS THEN

ADD gps price to total accessories price

END IF

GET exterior finish FROM USER

SET exterior finish price = 0

IF exterior finish is Standard THEN

SET exterior finish price = standard price

ELSE IF exterior finish is Modified THEN

SET exterior finish price = modified price

ELSE IF exterior finish is Customized THEN

SET exterior finish price = customized price

END IF

SET subtotal = basic price + total accessories price + exterior finish price

SET sales tax = subtotal \* sales tax rate

SET amount due = subtotal + sales tax - trade in allowance

DISPLAY "Subtotal: " + subtotal

DISPLAY "Sales Tax: " + sales tax

DISPLAY "Amount Due: " + amount due

END PROGRAM

**Explanation of Each Function:**

* get basic price(): This function asks the user to input the basic price of the vehicle and returns that value as an integer. It also checks that the input is a valid number and keeps asking until the user inputs a valid number.
* get trade in allowance(): This function asks the user to input the trade-in allowance and returns that value as an integer. It also checks that the input is a valid number and keeps asking until the user inputs a valid number.
* get accessories(): This function asks the user to input the accessories they want and returns that value as a string. It also checks that the input is one of the three valid options (Stereo System, Leather Interior, or GPS) and keeps asking until the user inputs a valid option.
* get exterior finish(): This function asks the user to input the exterior finish they want and returns that value as a string. It also checks that the input is one of the three valid options (Standard, Modified, or Customized) and keeps asking until the user inputs a valid option.
* get total accessories price(accessories): This function takes the user's selection of accessories and returns the total price based on the prices written above.
* get exterior finish price(exterior finish): This function takes the user's selection of exterior finish and returns the price based on the prices written above.
* get subtotal(basic price, total accessories price, exterior finish price): This function takes the basic price, total accessories price, and exterior finish price, and returns the subtotal by adding them together.
* get sales tax(subtotal, sales tax rate): This function takes the subtotal and sales tax rate, and returns the sales tax by multiplying them together.
* get amount due(sales tax, subtotal, trade in allowance): This function takes the sales tax, subtotal, and trade-in allowance, and returns the amount due by subtracting the trade-in allowance from the subtotal and adding the sales tax.
* display results(subtotal, sales tax, amount due): This function takes the subtotal, sales tax, and amount due, and displays them in a user-friendly way.
* The reset() function calls the clear screen() function and then the main() function to reset the program to its initial state. The clear screen() function simply resets all the variables to zero or empty strings.
* The while loop starts after the amount due has been calculated and printed out by the display results() function. It prompts the user to input a key to either clear the screen, reset the program, exit the program, or continue with the program.
* If the user presses 'c', the clear screen() function is called to reset all the variables to their initial values. If the user presses 'r', the main() function is called to reset the program to its initial state. If the user presses 'x' or 'Escape', the program is terminated using the sys.exit() function. If the user inputs any other key, the loop is broken and the program continues to exit.

**Test Cases:**

Test Case 1

Input:

basic price = 1000

trade in allowance = 0

accessories = Stereo System

exterior finish = Standard

Expected Output:

Subtotal: 1030.5

Sales Tax: 82.44

Amount Due: 1112.94

Test Case 2

Input:

basic price = 1000

trade in allowance = 0

accessories = Leather Interior

exterior finish = Standard

Expected Output:

Subtotal: 1530.99

Sales Tax: 122.48

Amount Due: 1653.47

Test Case 3

Input:

basic price = 1000

trade in allowance = 0

accessories = GPS

exterior finish = Standard

Expected Output:

Subtotal: 1301.9

Sales Tax: 104.15

Amount Due: 1406.05

Test Case 4

Input:

basic price = 1000

trade in allowance = 0

accessories = Stereo System

exterior finish = Modified

Expected Output:

Subtotal: 1401

Sales Tax: 112.08

Amount Due: 1513.08

Test Case 5

Input:

basic price = 1000

trade in allowance = 0

accessories = Leather Interior

exterior finish = Modified

Expected Output:

Subtotal: 1901.49

Sales Tax: 152.12

Amount Due: 2053.61

Test Case 6

Input:

basic price = 1000

trade in allowance = 0

accessories = GPS

exterior finish = Modified

Expected Output:

Subtotal: 1672.4

Sales Tax: 133.79

Amount Due: 1806.19

Test Case 7

Input:

basic price = 1000

trade in allowance = 0

accessories = Stereo System

exterior finish = Customized

Expected Output:

Subtotal: 2288.49

Sales Tax: 183.0792

Amount Due: 2471.5692

Test Case 8

Input:

basic price = 1000

trade in allowance = 0

accessories = Leather Interior

exterior finish = Customized

Expected Output:

Subtotal: 2788.98

Sales Tax: 223.1184

Amount Due: 3012.0984

Test Case 9

Input:

basic price = 1000

trade in allowance = 0

accessories = GPS

exterior finish = Customized

Expected Output:

Subtotal: 2559.89

Sales Tax: 204.7912

Amount Due: 2764.6812

Test Case 10

Input:

basic price = 1000

trade in allowance = 500

accessories = Stereo System

exterior finish = Standard

Expected Output:

Subtotal: 1030.5

Sales Tax: 82.44

Amount Due: 612.94

**Summary:**

The program is well-organized and easy to understand. It is composed of several functions, each responsible for a specific task. The first four functions, get basic price(), get trade in allowance(), get accessories(), and get exterior finish(), prompt the user for input and validate it. For example, get basic price() asks the user to enter the basic price of the vehicle and ensures that the input is a numeric value. The other functions follow a similar pattern and validate the user's input accordingly.

The next three functions, get total accessories price(), get exterior finish price(), and get subtotal(), calculate the prices of the selected accessories and exterior finish, and then calculate the subtotal of the transaction. get total accessories price() receives the accessory selected by the user as input and returns the price of the accessory. get exterior finish price() receives the exterior finish selected by the user and returns the corresponding price. get subtotal() receives the basic price of the vehicle, the total price of the accessories, and the price of the exterior finish as inputs, and then calculates the subtotal of the transaction.

The get sales tax() and get amount due() functions calculate the sales tax and amount due. get sales tax() receives the subtotal and sales tax rate as inputs and calculates the sales tax. get amount due() receives the sales tax, subtotal, and trade-in allowance as inputs and calculates the final amount due by subtracting the trade-in allowance from the subtotal and adding the sales tax.

The display results() function simply displays the results of the transaction to the user.

Finally, the reset() and clear screen() functions allow the user to clear the screen and reset the inputs or exit the program. The main() function is the main function of the program, which orchestrates the other functions and takes input from the user.

In conclusion, the program performs all the calculations required for a car sales transaction and validates the user's inputs. It is also easy to use and allows the user to reset the inputs or exit the program. Overall, the program is well-organized, efficient, and accurate in its calculations.