Michael Smith

Step One:

1. The user inputs consist of all the expenses and info they need to provide the company about their business trip. These inputs would vary from single yes or no answers and also exact prices.
2. The outputs would be various questions about the trip and then using those inputs to show the final data points as an output.
3. My assumptions are that the user inputs valid numbers and answers for the various questions. Majority of the inputs should be integers and there should be a few or none strings as inputs.
4. The overall task of the program is to calculate the expenses and reimbursements of the business trip. The subtasks would consist of various functions that are all used differently to calculate the final data sets using the inputs.

Step Two:

1. You should store the data using pointers and addresses so you're able to update them as the user enters more information.
2. I would store the expense in one variable that is being updated as the user inputs values. This means it would be updated in every function that would incorporate anything that influences the expense value.
3. The reimbursements would be calculated and updated within each function. The reimbursement variable would start at zero and then be added with every function that incorporates that value.
4. The expenses should be all the inputs of the user involving the cash he spent. You would then subtract the reimbursement value from the expense value to find the final expense of the trip that the employee should repay.
5. The input should be read every time they enter an integer and value that pertains to the final output. Also, any input that would change the course of the functions that would be ran.
6. A function would be repeated for everyday that the employee was on the business trip. These function could vary considering the other inputs the user enters.
7. You should be able to keep track of the reimbursement total through pointers and addresses. Both the reimbursement value and the expense value would be updated through the program as the user inputs value and data of the business trip.

Step Three:

1. Money: bad- not an integer, good- integer, min = 0
2. Time: bad- not an integer, good- integer within time limit, min = 0, max = 24hr
3. Car: bad- integer, good- string
4. Miles: bad- not an integer, good- integer, min = 0

Assignment #5 Design