# Lab 5

## Exception Handling

Skills Required

* Exception Handling
* Read and write Files, Work with arrays and vectors, Create Functions, Include Headers and other files, Loops (while, for), Conditional (if, switch), Datatypes, assignment, etc.

**Assignment**

1. Calculating fuel economy. This program will use exceptions and stream errors to make a robust application that gets the number of miles and gallons each time the user fuels their car. It will put those values into vectors. Once the user wants to quit enter values, it will calculate the fuel economy.
   1. Create GetMiles() function that returns double. It should prompt the user via cout, and read the result from cin.
      1. If there is a stream error, then throw a runtime\_error with the message Invalid input received, you must enter a decimal number. Don’t forget to clear the error and ignore all characters until the end of the stream.
      2. If the value is less then or equal to zero, then throw a runtime\_error with the message. Miles cannot be less than 0.
      3. Otherwise, return the miles the user entered
   2. Create GetGallons() function that returns double; again, the prompt is printed to cout, and the user input entered from cin.
      1. If there is a stream error, then throw a runtime\_error with the message Invalid input received, you must enter a decimal number. Don’t forget to clear the error and ignore all characters until the end of the stream.
      2. If the value is less than or equal to zero, then throw a runtime\_error with the message. Gallons cannot be less than 0.
      3. Otherwise, return the miles the user entered
   3. Create GetMPG(vector<double miles, vector<double> gallons) function that returns a double.
      1. If the size of the vectors is 0, then throw a runtime\_error with the message No values recorded MPG is nan.
         1. (‘nan’ stands for not-a-number; you can use the full phrase if you’d rather.)
      2. Otherwise, total the miles and gallons and return the miles per gallon.
   4. The main program should loop and get the Gallons and Miles catching any exceptions that were thrown. Then ask if they want to enter another tank. If they enter gallons and miles correctly, put the values into the vectors given. When the user is done the program should calculate the MPG by calling GetMPG, catching the exception if the user did not enter any values. Then it should show the result.

**Stream Errors**

cout << "Enter a number: " << endl;

cin >> number;

if (cin.fail()) {

// Clear error state

cin.clear();

// Ignore characters in stream until newline

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

cout << "There was an error: " << endl;

}

**Throwing Errors**

**throw** runtime\_error("Invalid value.");

**Catching Errors**

**try** {

// Code to try

}

**catch** (runtime\_error &excpt) {

// Prints the error message passed by throw statement

cout << excpt.what() << endl;

}

**Make sure you include stdexcept and vector as well as the other standard modules.**

**Submit your assignment**

Update your files on GitHub (Remember to save the files in the IDE before adding them), commit all changes, and push the changes to GitHub. Submit the GitHub link on Canvas.