David Lane

004638553

1. The first obstacle I overcame was not knowing where to start. I didn’t know how to move through a string running checks on individual characters. Eventually I read the textbook and found the best solution was to run a for loop with an index variable, so I used that with nested if statements to run the checks.

The biggest obstacle I encountered was with my first version of code, if the last digits of the commands string were a plus sign followed by a 1, the ontime percentage function returned nothing. This only happened when the last character was a 1, no other integer. So I went in to my ontime/late checking and found that the only problem I could see was the possibility of running out of bounds with some of my if statements. I found the solution was to add an if statement (commented on in my code) to ensure, that I was not running out of bounds.

1. My checker function runs mainly by using for loops to move through the given string, and several nested if statements to check each feature of the string. Example

For index = 0 to sizeof commands string;

{If (index character is a letter)

{Check to make sure the next two letters form a valid code;

Index = index + 2;

If (index character is a number)…

Call number checker function

If(index character is + or -)…

And so on in accordance with the format given in the assignment parameters, if it finds a problem anywhere along the string it returns false otherwise it returns true.

The ontime percentage checker is very similar. It first ensures the airlinecode and commands strings are valid, then it looks for instances of the airlinecode letters using a for loop moving the index variable. I declare three double variables:

double ontimeRatio;

double codeInstances(0);

double lateInstances(0);

…

if ( instance of airline code is found in commands)

{codeInstances++; //and we check if that flight is ontime by proceeding //through the string to the next instance of a + sign (if it were a – we //know its on time)

If (lots of logic to make sure the next the characters value more than 14)

lateInstances++;

//for loop continues

//finally we have

ontimeRatio = 1 – (lateInstances/codeInstances);

return ontimeRatio; // and if all is valid the function concludes by returning the ontime //percentage

1. First a list of ‘airlinecode’ possibilities
2. “aa” and “ua” make sure it returns true for valid lowercase strings
3. “AA” makes sure that the lowercase conversion works fine
4. “a” and “UAd” make sure the size check is working
5. “bc” makes sure that the checker returns false for invalid two letter combinations

Next for possible ‘commands’ strings

1. “ua” to make sure a bad string returns -1
2. “ua12-1” makes sure a basic valid string returns true
3. “ua12-1aa9-24” to make sure the loop runs seamlessly
4. “ua1221-899”
5. “ua13+1221” these two check if my flightnumbercheck and ontimecheck functions are working, should return false
6. “ua12+1” with airlinecode “AA” to ensure that if no instances are found in a valid commands string the ontime percentage checker returns 0
7. “ua12-1aa9-24aa12+15” with “AA” to make sure the function can check multiple instances of the airline code
8. The rest would be several more similar long strings like “ua12-1aa9+011aa12+1” to check the ontime/late logic and to make sure I don’t have out of bounds errors (commented on in my code)