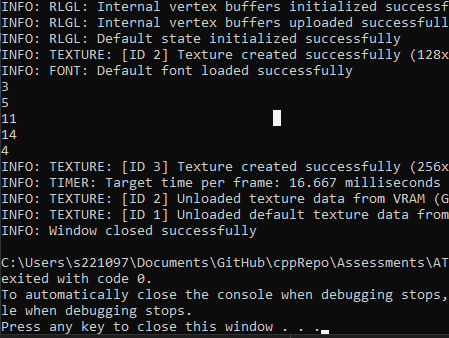
# Assessment Task 2: Debugging Problems

A summarised version of the other giant doc.

## Issue 1: Displaying Name

### TEST 1: check if nameSize being incorrectly set.

HOW: using 🡪 cout << nameSize << "\n";  
EXPECTED: 3, 5, 11, 14, 4  
ACTUAL: 3, 5, 11, 14, 4



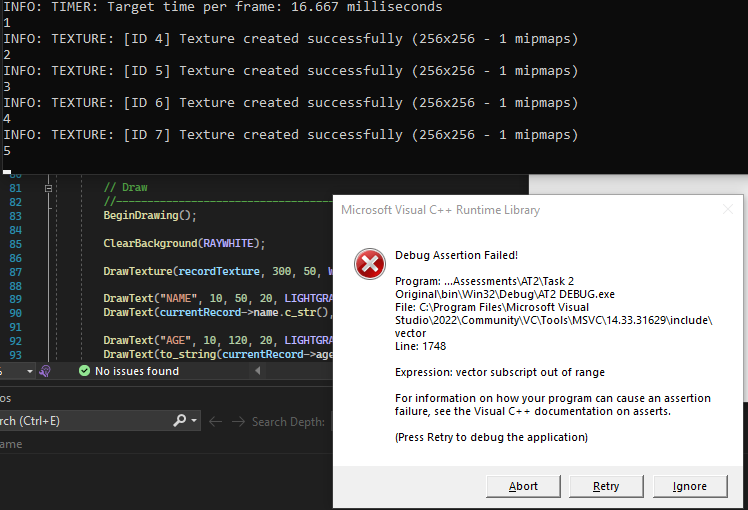
### Test 2: Check if the char\* length is being set correctly

HOW: using 🡪 cout << TextLength(name) << "\n";  
EXPECTED: 3, 5, 11, 14, 4  
ACTUAL: 12, 12, 15, 30, 12  
A screenshot of a computer program

Description automatically generated with medium confidence  
MODIFICATIONS MADE: Added 1 to name when char\* name is being declared, added a new line setting name[nameSize] = ‘\0’, denoting the end of the string.

## Issue 2: Crashing Program

### TEST 1: Check if bad value is being sent.

HOW: using 🡪 cout << currentRecordIdx << "\n";  
EXPECTED: 4  
ACTUAL: 5  
  
CHANGES MADE: subtracted 1 from the index in the if statement that’s trying to catch bad values.

### TEST 2: checking to see if it still gets set to 5.

HOW: ^^  
EXPECTED 4:  
ACTUAL 4:  
A screenshot of a computer program

Description automatically generated with medium confidence

## Issue 3: Random Access Pain

### TEST 1: Check file size is correct within program.

HOW: using 🡪 infile.seekg(0, ios::end);

cout << infile.tellg() << "\n";

EXPECTED: 1 310 861  
ACTUAL: 1 310 861

A screenshot of a computer

Description automatically generated with medium confidence  
MODIFICATIONS: none as a result of this

### TEST 2: check size of each Record

HOW: using 🡪 cout << recordSize << "\n";   
EXPECTED: 262151, 262153, 262159, 262162, 262152  
ACTUAL: 262151, 262153, 262159, 262162, 262152

A screenshot of a computer

Description automatically generated with medium confidence