

Documentation: Critical Thinking 2

Alejandro Ricciardi

Colorado State University Global

CSC450: Programming III

Professor: Reginald Haseltine

October 20, 2024

Documentation: Critical Thinking 2

This documentation is part of the Critical Thinking 2 Assignment from CSC450: Programming III at Colorado State University Global, the program name is Two String Input Concatenated. This documentation is an overview of a program's functionality and testing scenarios including console output screenshots. The program is coded in C++ 23.

The Assignment Direction:

Demonstrate an understanding of basic C++ programming concepts by completing the following:

- Program: Create a simple C++ console application that will write a program that will take two string inputs from a user. Your program should concatenate the two strings and then print the resulting output to the screen. Take the two string inputs from the user 3 times for varying string lengths.

Compile and submit your pseudocode, source code, and screenshots of the application executing the application, the results and your GIT repository in a single document.

⚠ My notes:

- The simple C++ console application is in file **CTA-2-Strings.cpp**

Program Description:

The program takes two strings inputted by the user, concatenates them, and prints the resulting concatenated string.

The input strings and their concatenation are safely handled using C++'s ‘`std::string`’ class, which automatically manages memory, preventing buffer overflows.

The program also uses ‘`std::getline`’ for string inputs, ensuring strings with spaces are fully captured.

The program repeats the process three times, accepting two string inputs and concatenating them to test if varying string lengths are handled securely.

Git Repository

I use [GitHub](#) as my Distributed Version Control System (DVCS), the following is a link to my GitHub, [Omegapy](#).

My GitHub repository that is used to store this assignment is named [My-Academics-Portfolio](#).

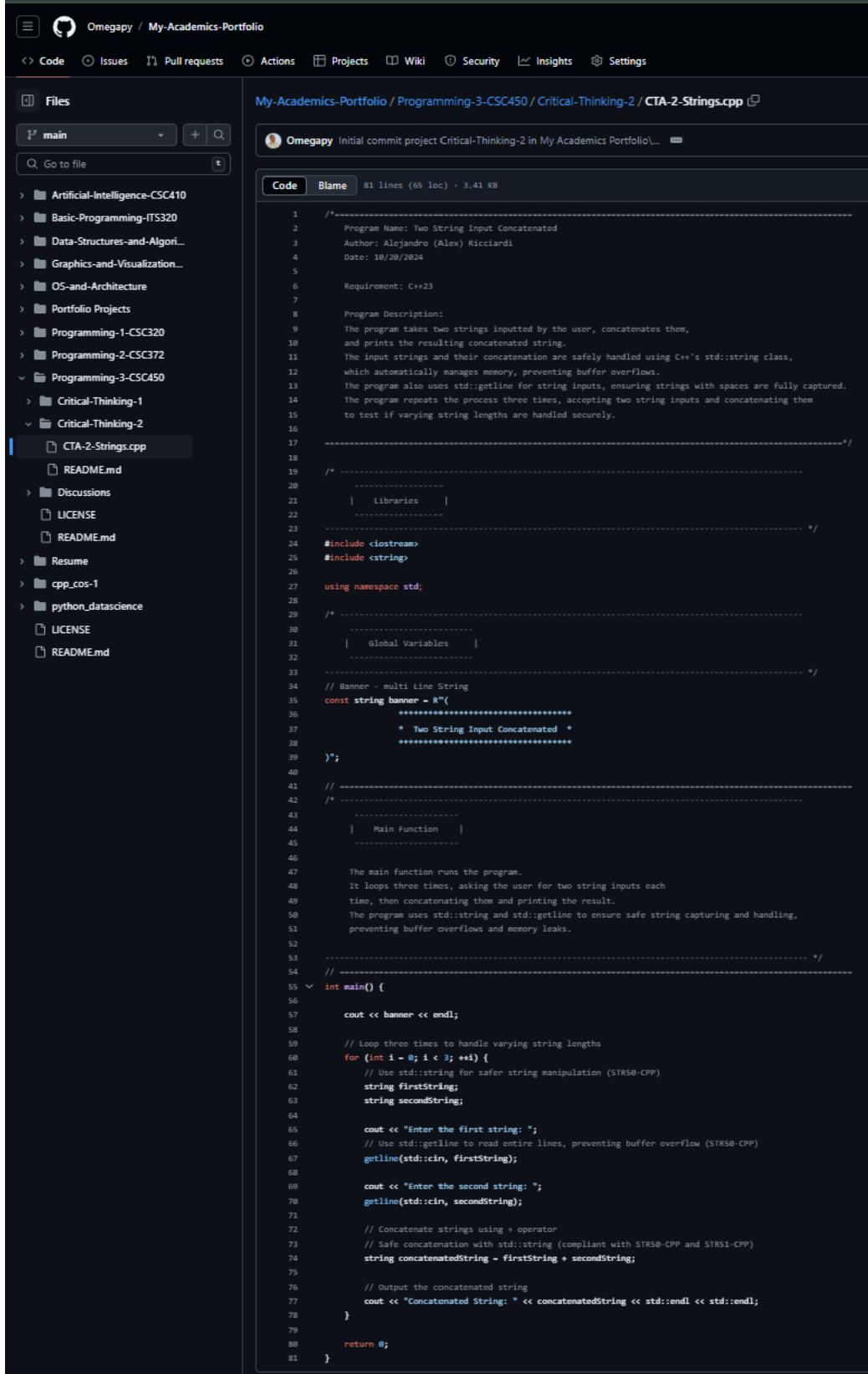
The link to this specific assignment is:

<https://github.com/Omegapy/My-Academics-Portfolio/tree/main/Programming-3-CSC450/Critical-Thinking-2>



Image of the source code in the GitHub: see next page

Figure 1
Source Code in GitHub



The screenshot shows a GitHub repository interface. The top navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The repository name is "My-Academics-Portfolio". The current file being viewed is "CTA-2-Strings.cpp" located in the "Programming-3-CSC450/Critical-Thinking-2" directory. The code editor displays the content of the C++ program, which concatenates two strings input by the user three times. The code is well-documented with comments explaining its purpose, requirements, and safe string handling.

```

/*
-----+
2 Program Name: Two String Input Concatenated
3 Author: Alejandro (Alex) Ricciardi
4 Date: 10/28/2024
5
6 Requirement: C++23
7
8 Program Description:
9 The program takes two strings inputted by the user, concatenates them,
10 and prints the resulting concatenated string.
11 The input strings and their concatenation are safely handled using C++'s std::string class,
12 which automatically manages memory, preventing buffer overflows.
13 The program also uses std::getline for string inputs, ensuring strings with spaces are fully captured.
14 The program repeats the process three times, accepting two string inputs and concatenating them
15 to test if varying string lengths are handled securely.
16
17 -----
18
19 /*
20 | Libraries |
21 |-----|
22 |----- */
23
24 #include <iostream>
25 #include <string>
26
27 using namespace std;
28
29 /*
30 | Global Variables |
31 |-----|
32 |----- */
33
34 // Banner - multi Line String
35 const string banner = R"(
36     *****
37     * Two String Input Concatenated *
38     *****
39 )";
40
41 // -----
42 /*
43 | Main Function |
44 |-----|
45
46 The main function runs the program.
47 It loops three times, asking the user for two string inputs each
48 time, then concatenating them and printing the result.
49 The program uses std::string and std::getline to ensure safe string capturing and handling,
50 preventing buffer overflows and memory leaks.
51
52
53 // -----
54
55 int main() {
56
57     cout << banner << endl;
58
59     // Loop three times to handle varying string lengths
60     for (int i = 0; i < 3; ++i) {
61         // Use std::string for safer string manipulation (STR50-CPP)
62         string firstString;
63         string secondString;
64
65         cout << "Enter the first string: ";
66         // Use std::getline to read entire lines, preventing buffer overflow (STR50-CPP)
67         getline(std::cin, firstString);
68
69         cout << "Enter the second string: ";
70         getline(std::cin, secondString);
71
72         // Concatenate strings using + operator
73         // Safe concatenation with std::string (compliant with STR50-CPP and STR51-CPP)
74         string concatenatedString = firstString + secondString;
75
76         // Output the concatenated string
77         cout << "Concatenated String: " << concatenatedString << std::endl << std::endl;
78     }
79
80     return 0;
81 }

```

Figure 2
Source Code in IDE

CTA-2-Strings.cpp X

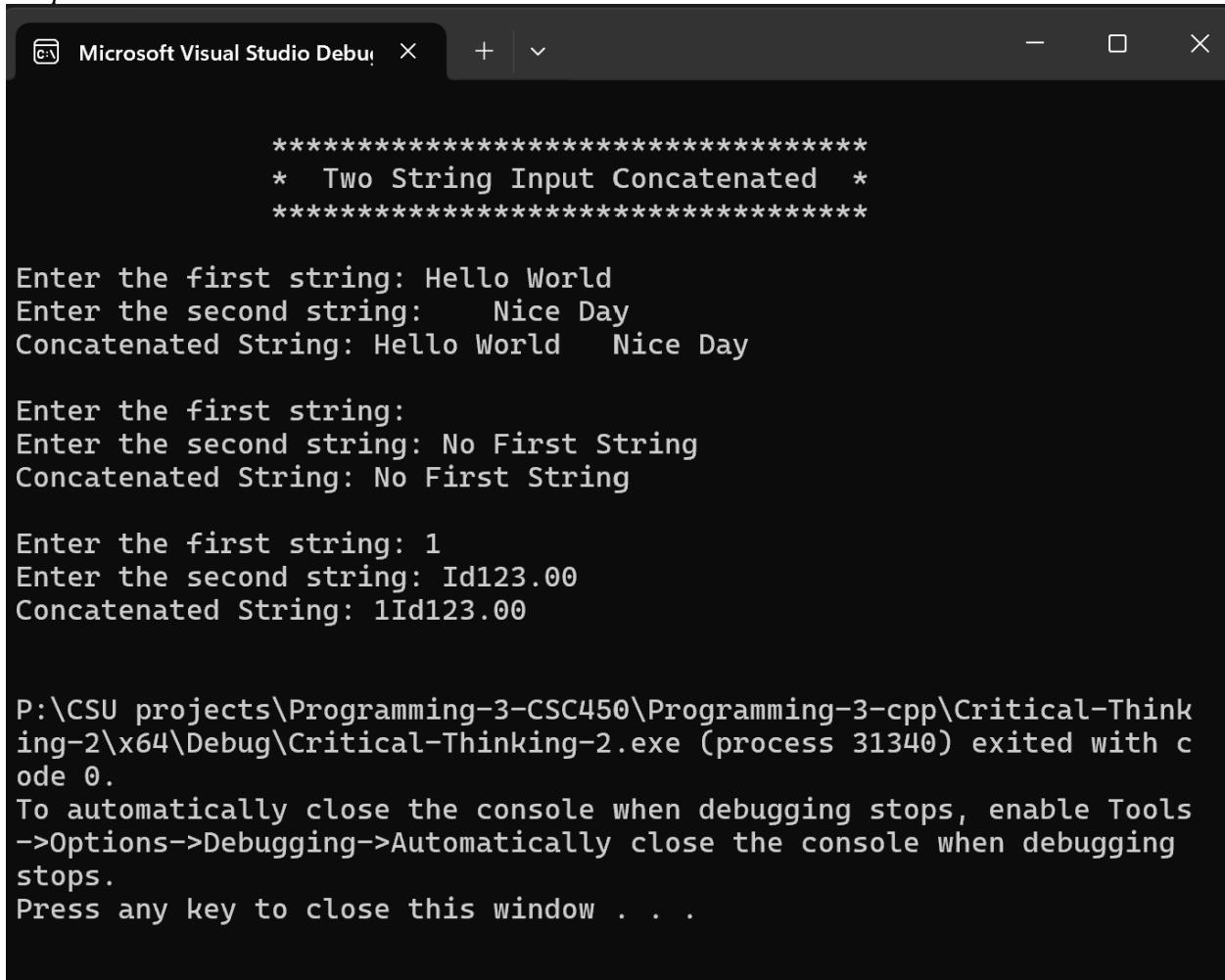
++ Critical-Thinking-2

```

1 //=====
2 // Program Name: Two String Input Concatenated
3 // Author: Alejandro (Alex) Ricciardi
4 // Date: 10/20/2024
5
6 Requirement: C++23
7
8 Program Description:
9 The program takes two strings inputted by the user, concatenates them,
10 and prints the resulting concatenated string.
11 The input strings and their concatenation are safely handled using C++'s std::string class,
12 which automatically manages memory, preventing buffer overflows.
13 The program also uses std::getline for string inputs, ensuring strings with spaces are fully captured.
14 The program repeats the process three times, accepting two string inputs and concatenating them
15 to test if varying string lengths are handled securely.
16 =====
17
18
19 /* -----
20 | Libraries |
21 -----
22 */
23 #include <iostream>
24 #include <string>
25
26 using namespace std;
27
28 /*
29 | -----
30 | Global Variables |
31 | -----
32 */
33 // Banner - multi Line String
34 const string banner = R"(
35 *****
36 * Two String Input Concatenated *
37 *****
38 )";
39
40 // -----
41 // -----
42 /*
43 | Main Function |
44 | -----
45 */
46
47 The main function runs the program.
48 It loops three times, asking the user for two string inputs each
49 time, then concatenating them and printing the result.
50 The program uses std::string and std::getline to ensure safe string capturing and handling,
51 preventing buffer overflows and memory leaks.
52
53 // =====
54 int main() {
55
56     cout << banner << endl;
57
58     // Loop three times to handle varying string lengths
59     for (int i = 0; i < 3; ++i) {
60         // Use std::string for safer string manipulation (STR50-CPP)
61         string firstString;
62         string secondString;
63
64         cout << "Enter the first string: ";
65         // Use std::getline to read entire lines, preventing buffer overflow (STR50-CPP)
66         getline(std::cin, firstString);
67
68         cout << "Enter the second string: ";
69         getline(std::cin, secondString);
70
71         // Concatenate strings using + operator
72         // Safe concatenation with std::string (compliant with STR50-CPP and STR51-CPP)
73         string concatenatedString = firstString + secondString;
74
75         // Output the concatenated string
76         cout << "Concatenated String: " << concatenatedString << std::endl << std::endl;
77     }
78
79     return 0;
80 }
81
82

```

Figure 3
Output Code



The screenshot shows a Microsoft Visual Studio Debug console window. The title bar reads "Microsoft Visual Studio Debug". The console displays three separate runs of a program. Each run starts with a header: "*****", "* Two String Input Concatenated *", and "*****". The first run prompts for two strings ("Enter the first string:" and "Enter the second string:"), concatenates them ("Concatenated String:"), and then repeats the process. The second run also repeats the same steps. The third run concatenates the strings "1" and "Id123.00" to produce "1Id123.00". Finally, the console shows the path "P:\CSU projects\Programming-3-CSC450\Programming-3-cpp\Critical-Thinking-2\x64\Debug\Critical-Thinking-2.exe (process 31340) exited with code 0.", a message about automatically closing the console, and a prompt to press any key.

```
*****
* Two String Input Concatenated *
*****

Enter the first string: Hello World
Enter the second string: Nice Day
Concatenated String: Hello World Nice Day

Enter the first string:
Enter the second string: No First String
Concatenated String: No First String

Enter the first string: 1
Enter the second string: Id123.00
Concatenated String: 1Id123.00

P:\CSU projects\Programming-3-CSC450\Programming-3-cpp\Critical-Thinking-2\x64\Debug\Critical-Thinking-2.exe (process 31340) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Debugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```

As shown in Figures 1 through 3 the programs run without any issues displaying the correct outputs as expected.

