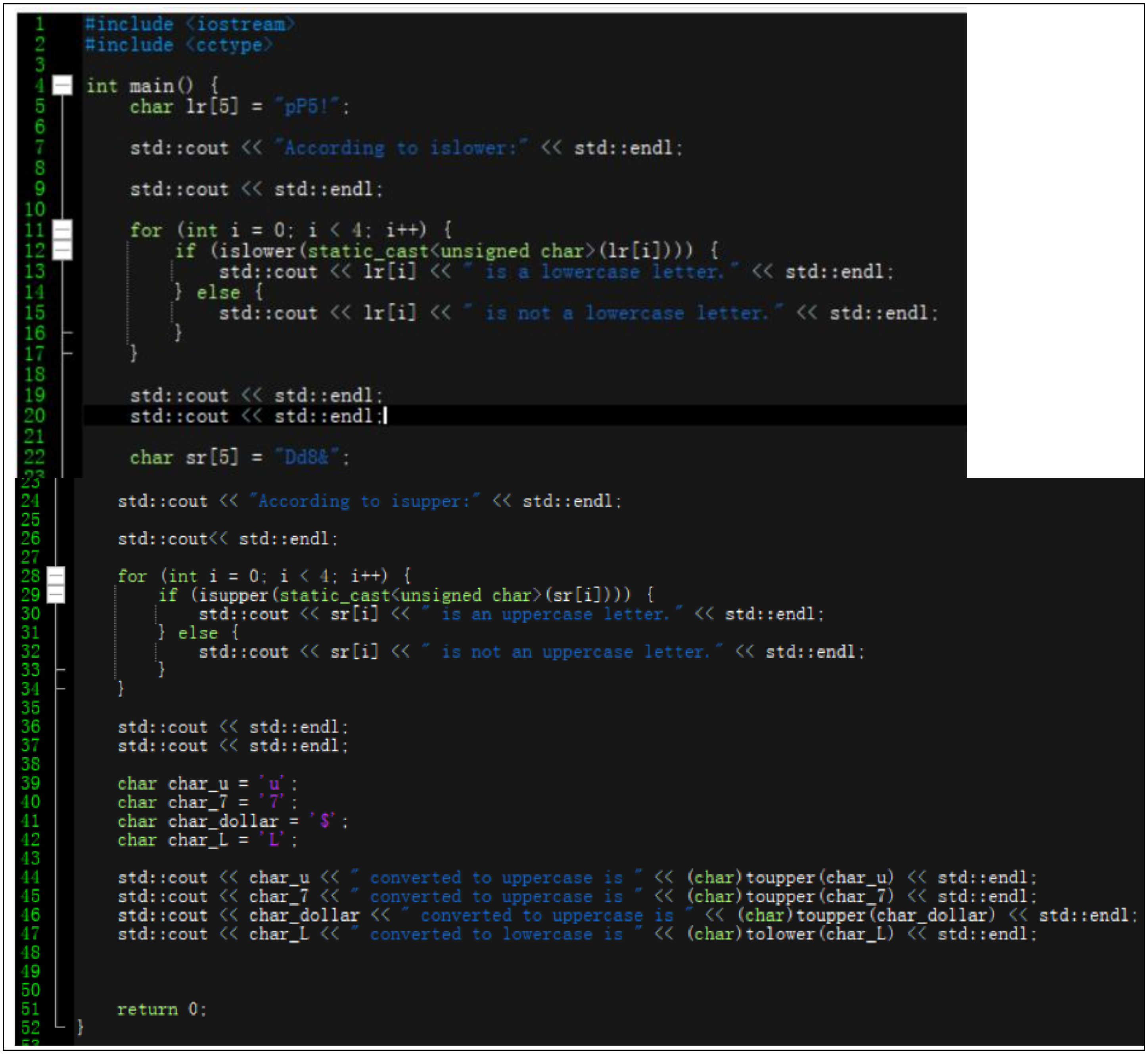
Activity No. 4.4	
Hands-on Activity 4.4: Characters and Strings	
Course Code: CPE007	**Program:** Computer Engineering
Course Title: Programming Logic and Design	**Date Performed: 9 – 24 – 25**
Section: CPE11S1	**Date Submitted: 9 – 25 – 25**
Name(s): Paula Esguerra	**Instructor: Engr. Jimlord Quejado**

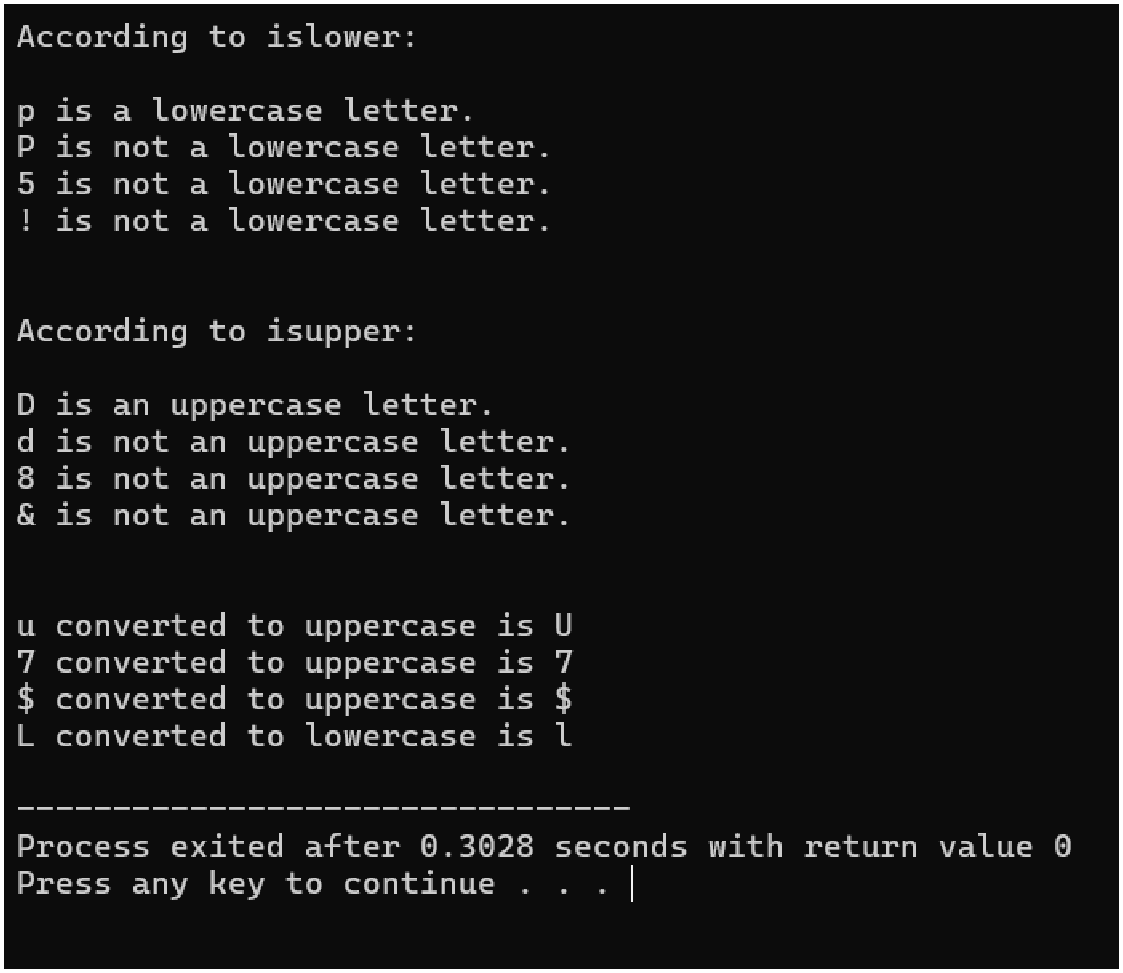
6. Output

1. Try to create a program that outputs the following?

```
According to islower:

p is a lowercase letter

P is not a lowercase letter

5 is not a lowercase letter

! is not a lowercase letter



According to isupper:

D is an uppercase letter

d is not an uppercase letter

8 is not an uppercase letter

& is not an uppercase letter



u converted to uppercase is U

7 converted to uppercase is 7

$ converted to uppercase is $

L converted to lowercase is l
```



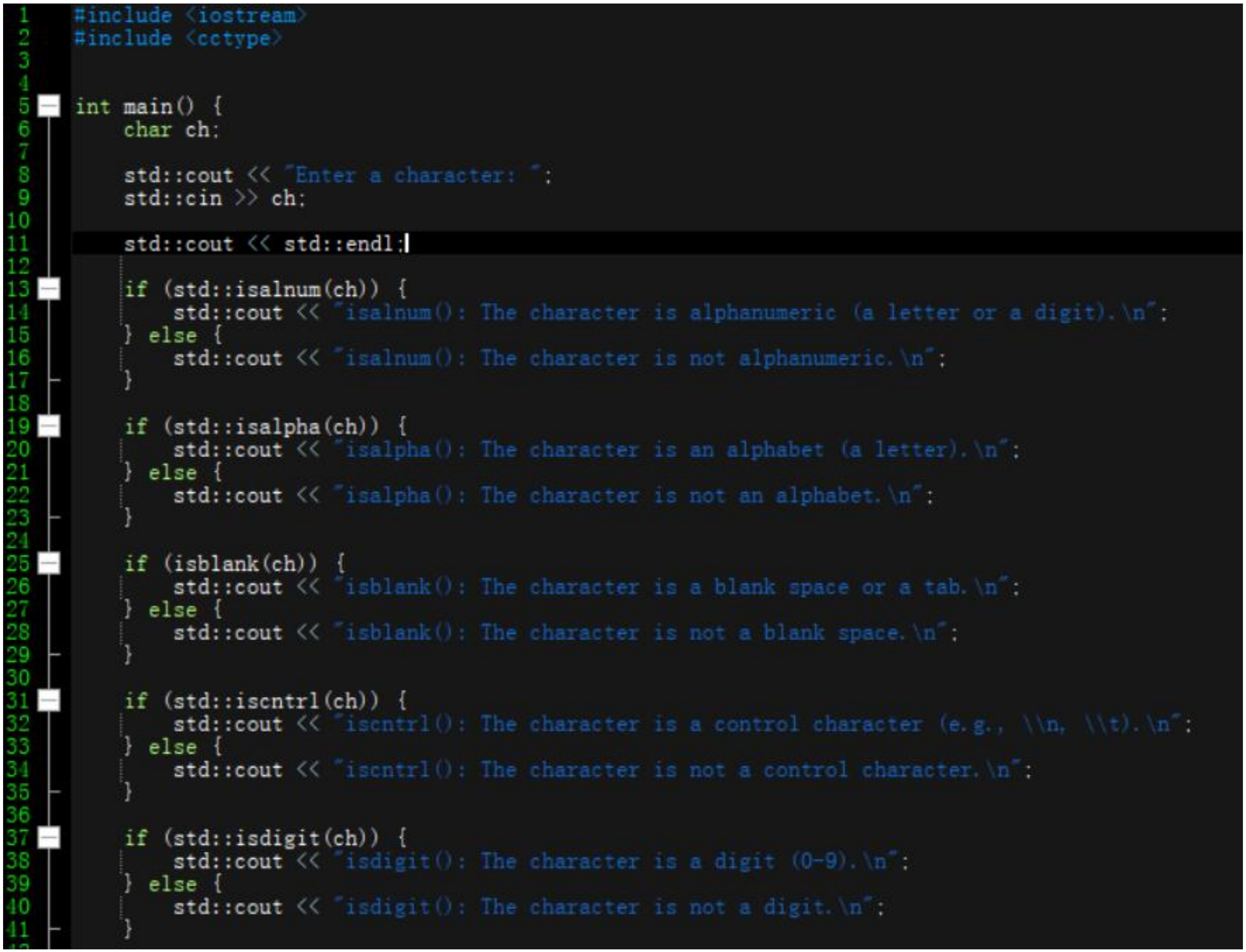
```cpp
#include <iostream>
#include <cctype>

int main() {
    char lr[5] = "pP5!";

    std::cout << "According to islower:" << std::endl;

    std::cout << std::endl;

    for (int i = 0; i < 4; i++) {
        if (islower(static_cast<unsigned char>(lr[i]))) {
            std::cout << lr[i] << " is a lowercase letter." << std::endl;
        } else {
            std::cout << lr[i] << " is not a lowercase letter." << std::endl;
        }
    }

    std::cout << std::endl;
    std::cout << std::endl;

    char sr[5] = "Dd8&";

    std::cout << "According to isupper:" << std::endl;

    std::cout<< std::endl;

    for (int i = 0; i < 4; i++) {
        if (isupper(static_cast<unsigned char>(sr[i]))) {
            std::cout << sr[i] << " is an uppercase letter." << std::endl;
        } else {
            std::cout << sr[i] << " is not an uppercase letter." << std::endl;
        }
    }

    std::cout << std::endl;
    std::cout << std::endl;

    char char_u = 'u';
    char char_7 = '7';
    char char_dollar = '$';
    char char_L = 'L';

    std::cout << char_u << " converted to uppercase is " << (char)toupper(char_u) << std::endl;
    std::cout << char_7 << " converted to uppercase is " << (char)toupper(char_7) << std::endl;
    std::cout << char_dollar << " converted to uppercase is " << (char)toupper(char_dollar) << std::endl;
    std::cout << char_L << " converted to lowercase is " << (char)tolower(char_L) << std::endl;



    return 0;
}
```



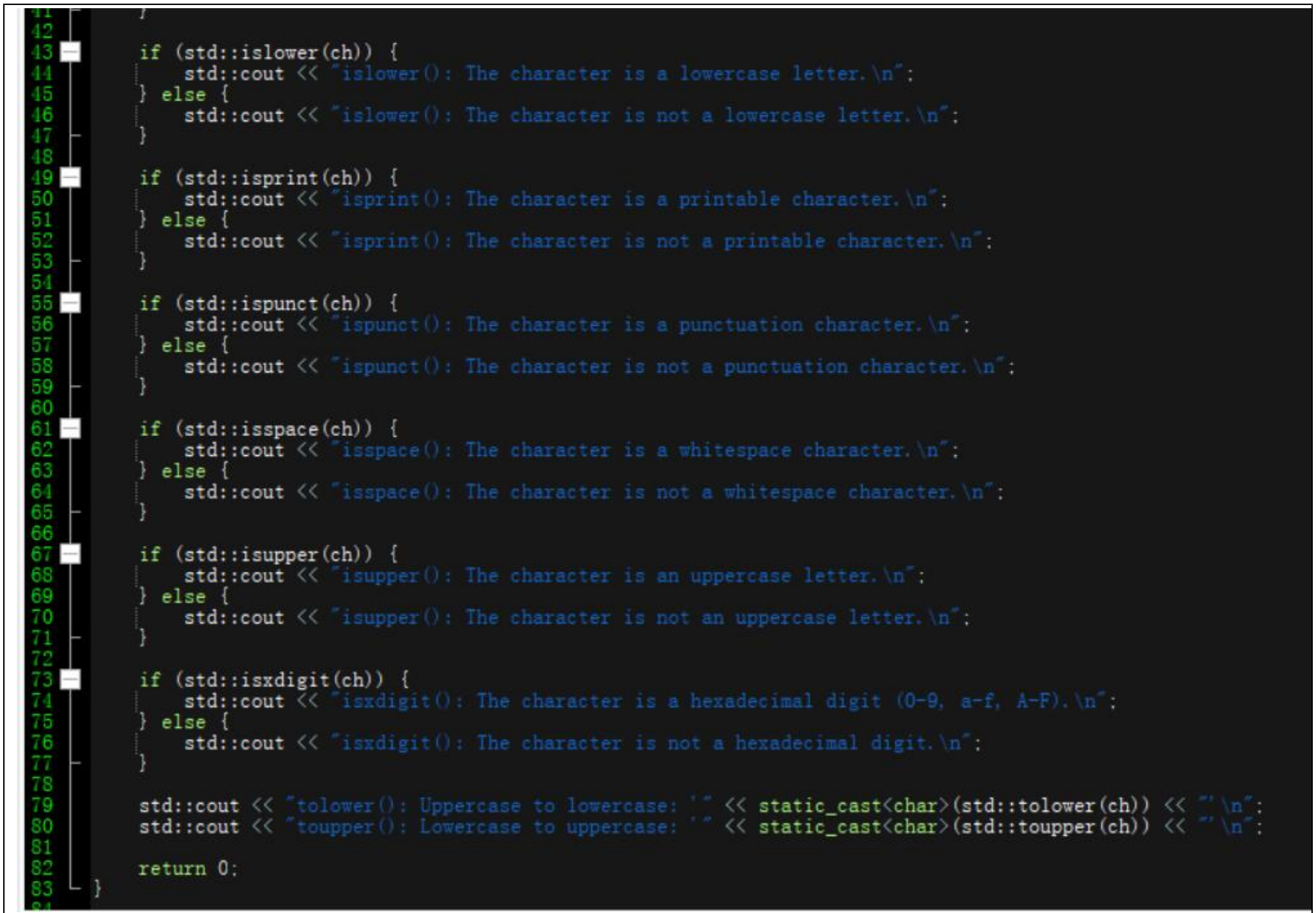
```
According to islower:

p is a lowercase letter.
P is not a lowercase letter.
5 is not a lowercase letter.
! is not a lowercase letter.


According to isupper:

D is an uppercase letter.
d is not an uppercase letter.
8 is not an uppercase letter.
& is not an uppercase letter.


u converted to uppercase is U
7 converted to uppercase is 7
$ converted to uppercase is $
L converted to lowercase is l

--------------------------------
Process exited after 0.3028 seconds with return value 0
Press any key to continue . . .
```

7. Supplementary Activity

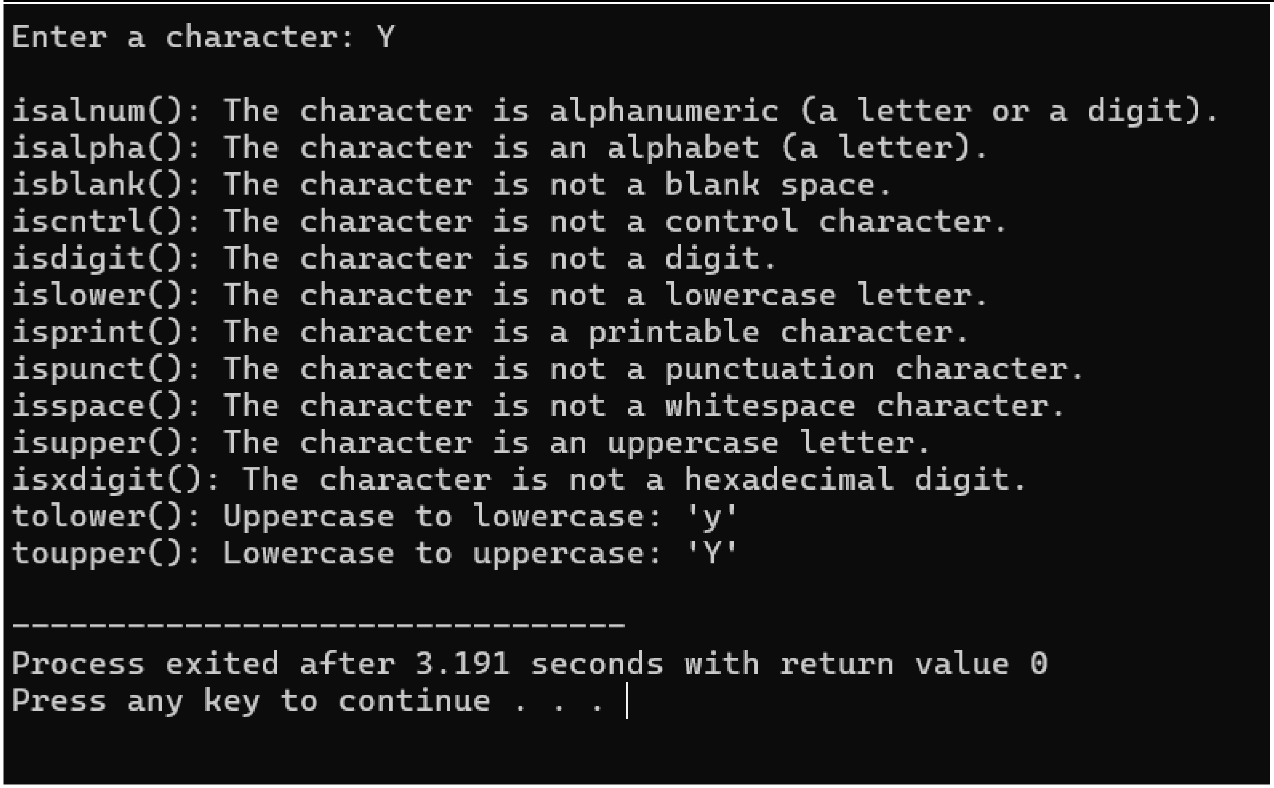
2. Write a program that inputs a character from the keyboard and tests the character with each of the functions in the character handling library. (Refer to the first Table above)



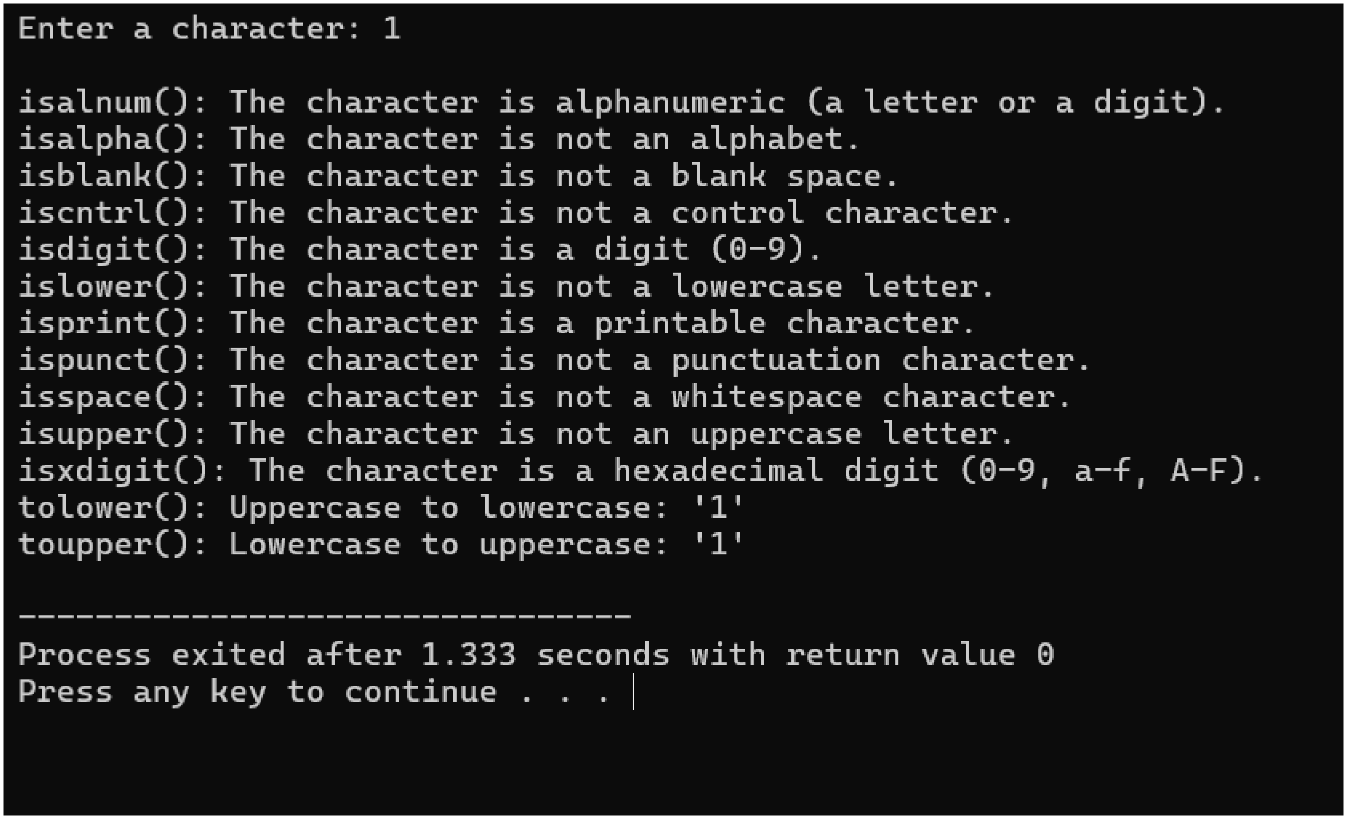
```cpp
#include <iostream>
#include <cctype>


int main() {
    char ch;

    std::cout << "Enter a character: ";
    std::cin >> ch;

    std::cout << std::endl;

    if (std::isalnum(ch)) {
        std::cout << "isalnum(): The character is alphanumeric (a letter or a digit).\n";
    } else {
        std::cout << "isalnum(): The character is not alphanumeric.\n";
    }

    if (std::isalpha(ch)) {
        std::cout << "isalpha(): The character is an alphabet (a letter).\n";
    } else {
        std::cout << "isalpha(): The character is not an alphabet.\n";
    }

    if (isblank(ch)) {
        std::cout << "isblank(): The character is a blank space or a tab.\n";
    } else {
        std::cout << "isblank(): The character is not a blank space.\n";
    }

    if (std::iscntrl(ch)) {
        std::cout << "iscntrl(): The character is a control character (e.g., \\n, \\t).\n";
    } else {
        std::cout << "iscntrl(): The character is not a control character.\n";
    }

    if (std::isdigit(ch)) {
        std::cout << "isdigit(): The character is a digit (0-9).\n";
    } else {
        std::cout << "isdigit(): The character is not a digit.\n";
    }
```



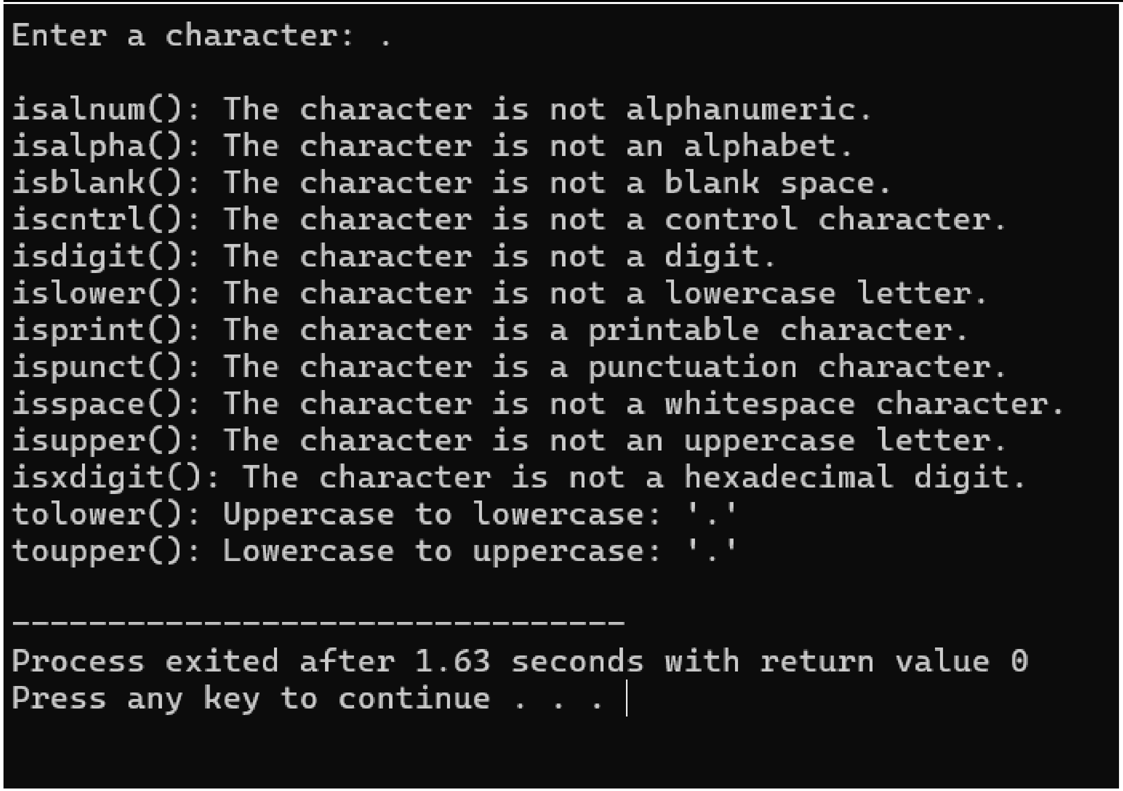
```cpp
    }

    if (std::islower(ch)) {
        std::cout << "islower(): The character is a lowercase letter.\n";
    } else {
        std::cout << "islower(): The character is not a lowercase letter.\n";
    }

    if (std::isprint(ch)) {
        std::cout << "isprint(): The character is a printable character.\n";
    } else {
        std::cout << "isprint(): The character is not a printable character.\n";
    }

    if (std::ispunct(ch)) {
        std::cout << "ispunct(): The character is a punctuation character.\n";
    } else {
        std::cout << "ispunct(): The character is not a punctuation character.\n";
    }

    if (std::isspace(ch)) {
        std::cout << "isspace(): The character is a whitespace character.\n";
    } else {
        std::cout << "isspace(): The character is not a whitespace character.\n";
    }

    if (std::isupper(ch)) {
        std::cout << "isupper(): The character is an uppercase letter.\n";
    } else {
        std::cout << "isupper(): The character is not an uppercase letter.\n";
    }

    if (std::isxdigit(ch)) {
        std::cout << "isxdigit(): The character is a hexadecimal digit (0-9, a-f, A-F).\n";
    } else {
        std::cout << "isxdigit(): The character is not a hexadecimal digit.\n";
    }

    std::cout << "tolower(): Uppercase to lowercase: '" << static_cast<char>(std::tolower(ch)) << "'\n";
    std::cout << "toupper(): Lowercase to uppercase: '" << static_cast<char>(std::toupper(ch)) << "'\n";

    return 0;
}
```



```
Enter a character: Y

isalnum(): The character is alphanumeric (a letter or a digit).
isalpha(): The character is an alphabet (a letter).
isblank(): The character is not a blank space.
iscntrl(): The character is not a control character.
isdigit(): The character is not a digit.
islower(): The character is not a lowercase letter.
isprint(): The character is a printable character.
ispunct(): The character is not a punctuation character.
isspace(): The character is not a whitespace character.
isupper(): The character is an uppercase letter.
isxdigit(): The character is not a hexadecimal digit.
tolower(): Uppercase to lowercase: 'y'
toupper(): Lowercase to uppercase: 'Y'

--------------------------------
Process exited after 3.191 seconds with return value 0
Press any key to continue . . .
```



```
Enter a character: 1

isalnum(): The character is alphanumeric (a letter or a digit).
isalpha(): The character is not an alphabet.
isblank(): The character is not a blank space.
iscntrl(): The character is not a control character.
isdigit(): The character is a digit (0-9).
islower(): The character is not a lowercase letter.
isprint(): The character is a printable character.
ispunct(): The character is not a punctuation character.
isspace(): The character is not a whitespace character.
isupper(): The character is not an uppercase letter.
isxdigit(): The character is a hexadecimal digit (0-9, a-f, A-F).
tolower(): Uppercase to lowercase: '1'
toupper(): Lowercase to uppercase: '1'

--------------------------------
Process exited after 1.333 seconds with return value 0
Press any key to continue . . .
```

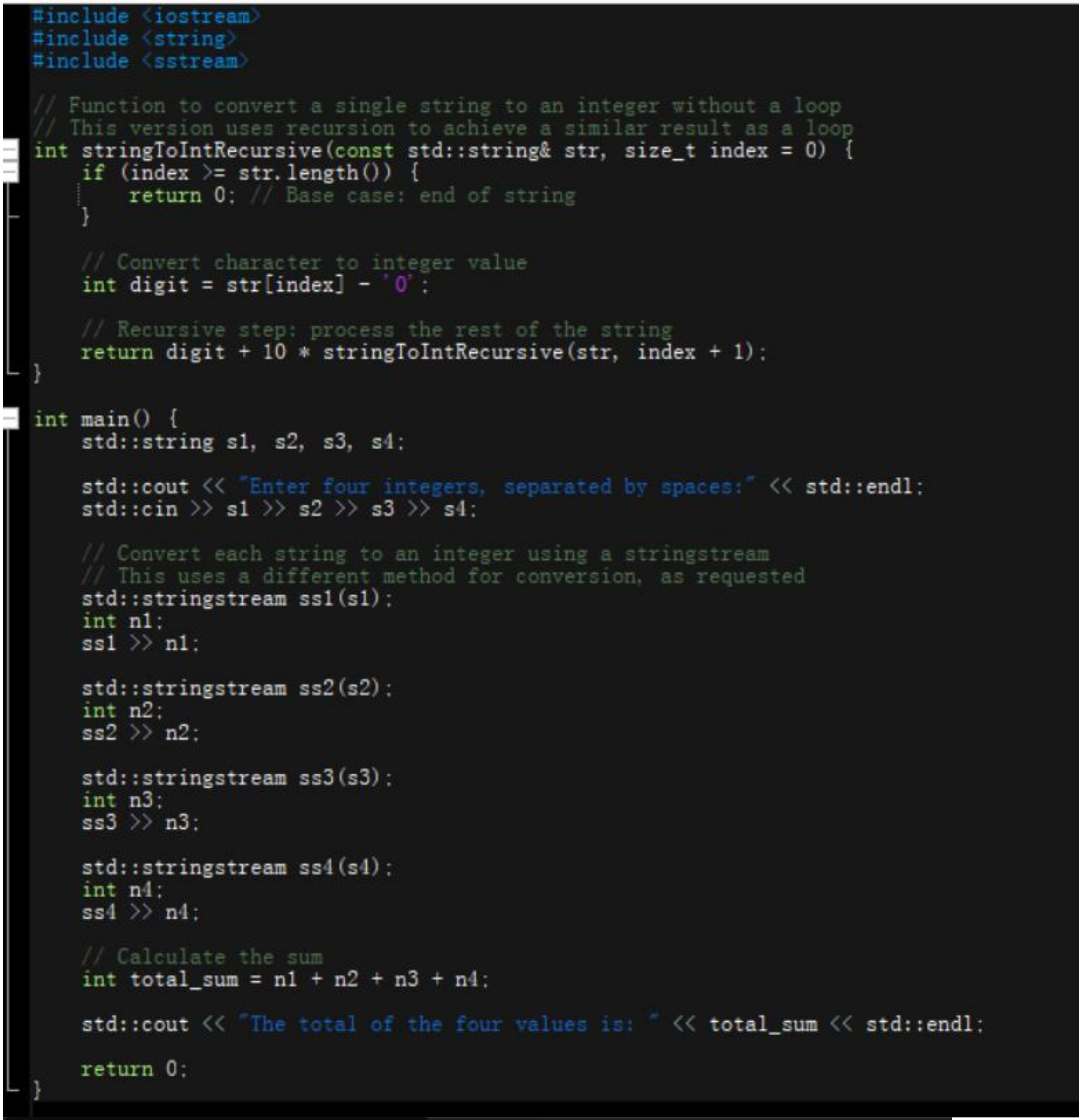


```
Enter a character: .

isalnum(): The character is not alphanumeric.
isalpha(): The character is not an alphabet.
isblank(): The character is not a blank space.
iscntrl(): The character is not a control character.
isdigit(): The character is not a digit.
islower(): The character is not a lowercase letter.
isprint(): The character is a printable character.
ispunct(): The character is a punctuation character.
isspace(): The character is not a whitespace character.
isupper(): The character is not an uppercase letter.
isxdigit(): The character is not a hexadecimal digit.
tolower(): Uppercase to lowercase: '.'
toupper(): Lowercase to uppercase: '.'

--------------------------------
Process exited after 1.63 seconds with return value 0
Press any key to continue . . .
```

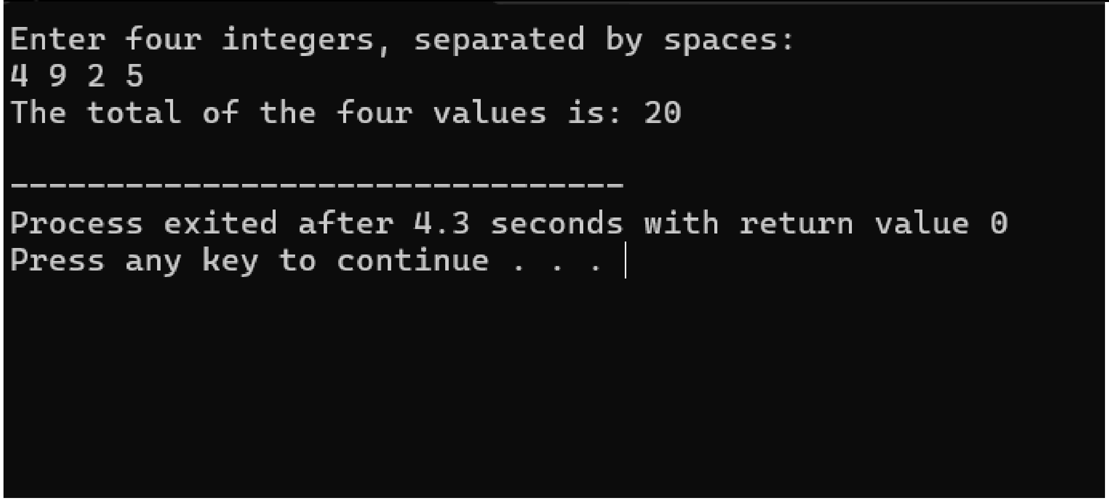
Analysis:

In this program, it considers what is inputted then one by one is analyzed if it is alphanumeric, alphabet, blank space, control character, digit, lowercase letter, printable character, punctuation character, whitespace character, uppercase letter, hexadecimal digit, and prints its upper and lower case at the end. If-else loop was used for it will be analyzed one by one at every step.

3. Write a program that inputs 4 strings that represent integers, converts the strings to integers, sums the values and prints the total of the 4 values.



```cpp
#include <iostream>
#include <string>
#include <sstream>

// Function to convert a single string to an integer without a loop
// This version uses recursion to achieve a similar result as a loop
int stringToIntRecursive(const std::string& str, size_t index = 0) {
    if (index >= str.length()) {
        return 0; // Base case: end of string
    }

    // Convert character to integer value
    int digit = str[index] - '0';

    // Recursive step: process the rest of the string
    return digit + 10 * stringToIntRecursive(str, index + 1);
}

int main() {
    std::string s1, s2, s3, s4;

    std::cout << "Enter four integers, separated by spaces:" << std::endl;
    std::cin >> s1 >> s2 >> s3 >> s4;

    // Convert each string to an integer using a stringstream
    // This uses a different method for conversion, as requested
    std::stringstream ss1(s1);
    int n1;
    ss1 >> n1;

    std::stringstream ss2(s2);
    int n2;
    ss2 >> n2;

    std::stringstream ss3(s3);
    int n3;
    ss3 >> n3;

    std::stringstream ss4(s4);
    int n4;
    ss4 >> n4;

    // Calculate the sum
    int total_sum = n1 + n2 + n3 + n4;

    std::cout << "The total of the four values is: " << total_sum << std::endl;

    return 0;
}
```



```
Enter four integers, separated by spaces:
4 9 2 5
The total of the four values is: 20

--------------------------------
Process exited after 4.3 seconds with return value 0
Press any key to continue . . .
```

Analysis:

In this code, I have observed from what i searched upon and wanted to try too, that it uses no loop, but it is similiar to where it uses a loop. Then it uses stringstream, where it uses a different method for conversion and calculated the sum.

8. Conclusion

From what I learned, for the first in forever, I got the code I wanted to in the output section, I somehow got the concept of the for loop, also the spaces. Then for the supplementary part, I enjoyed doing no. 2 it analyzes one input then defines them. Then for no. 3, I wanted to explore new codes, so I went a bit different from what we were taught, I am still learning how I did that and how I coded that, but still I learned something new, which is sstream, and the other character handling library.

9. Assessment Rubric