

Hands-On Activity 4.4	
Characters and Strings	
Course Code: CPE007	Program: Computer Engineering
Course Title: Programming Logic and Design	Date Performed: 25/09/2025
Section: CPE11S1	Date Submitted: 25/09/2025
Name(s): James Daniel M. Verano	Instructor: Engr. Jimlord M. Quejado

6. Output

Code Snippet:

- (Islower) function

```

1  #include <iostream>
2  using namespace std;
3
4  int main() {
5
6      char ch;
7
8      ch = 'p';
9      if (islower(ch)) {
10         cout << ch << " Is A Lowercase Letter.\n";
11     } else {
12         cout << ch << " is Not A Lowercase Letter.\n";
13     }
14
15     ch = 'P';
16     if (islower(ch)) {
17         cout << ch << " is Not A Lowercase Letter.\n";
18     } else {
19         cout << ch << " is Not A Lowercase Letter.\n";
20     }
21
22     ch = '5';
23     if (islower(ch)) {
24         cout << ch << " Is Not A Lowercase Letter.\n";
25     } else {
26         cout << ch << " is Not A Lowercase Letter.\n";
27     }
28
29     ch = '!';
30     if (islower(ch)) {
31         cout << ch << " Is Not A Lowercase Letter.\n";
32     } else {
33         cout << ch << " is Not A Lowercase Letter.\n";
34     }
35

```

Output:

```
p Is A Lowercase Letter.  
P is Not A Lowercase Letter.  
5 is Not A Lowercase Letter.  
! is Not A Lowercase Letter.  
  
-----  
Process exited after 0.1706 seconds with return value 0  
Press any key to continue . . .
```

Analysis:

- *In the provided code, we are using the `islower()` function to determine if a character is a lowercase letter. We are using the if else statement thus, The function `islower()` returns a value depending if the character is a lowercase letter, and if it is not. Since we are using an if else statement, the variable code assigns a lowercase letter 'p' to `ch`, then the '`islower(ch)`' function is initiated to check if the letter is lowercase. If it is, the program prints that functioning cout code depending on what if-else statement condition was met.*

- (Isupper) function

Code Snippet:

```
38     ch = 'D';
39     if (isupper(ch)) {
40         cout << ch << " Is A Uppercase Letter.\n";
41     } else {
42         cout << ch << " is Not A Uppercase Letter.\n";
43     }
44
45     ch = 'd';
46     if (isupper(ch)) {
47         cout << ch << " is Not A Uppercase Letter.\n";
48     } else {
49         cout << ch << " is Not A Uppercase Letter.\n";
50     }
51
52     ch = '8';
53     if (isupper(ch)) {
54         cout << ch << " Is Not A Uppercase Letter.\n";
55     } else {
56         cout << ch << " is Not A Uppercase Letter.\n";
57     }
58
59     ch = '&';
60     if (isupper(ch)) {
61         cout << ch << " Is Not A Uppercase Letter.\n";
62     } else {
63         cout << ch << " is Not A Uppercase Letter.\n";
64     }
```

Output:

```
D Is A Uppercase Letter.
d is Not A Uppercase Letter.
8 is Not A Uppercase Letter.
& is Not A Uppercase Letter.

-----
Process exited after 0.1481 seconds with return value 0
Press any key to continue . . .
```

Analysis:

- In this part it has the same functionality on the previous function, which was the islower() function. The isupper() function is the opposite of what the islower() function does in terms of converting characters to uppercase. We are using the same if-else statement thus, The function analyzes the same procedure from the previous program, in which the function is initiated to check if the letter is uppercase or not. If it is, the program prints that functioning cout code depending on what if-else statement condition was met.

- (Toupper) & (Tolower) Function

Code Snippet:

```
1  #include <iostream>
2  #include <cctype>
3  using namespace std;
4
5  int main() {
6
7      char ch;
8
9      ch = 'u';
10     cout << ch << " Converted to uppercase is: " << (char)toupper(ch) << "\n";
11     ch = '7';
12     cout << ch << " Converted to uppercase is: " << (char)toupper(ch) << "\n";
13     ch = '$';
14     cout << ch << " Converted to uppercase is: " << (char)toupper(ch) << "\n";
15     ch = 'L';
16     cout << ch << " Converted to lowercase is: " << (char)tolower(ch) << "\n";
17 }
```

Output:

```
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 0.1471 seconds with return value 0
Press any key to continue . . .
```

Analysis:

- This program is a combination of both first and second examples, but in this function, it is within the cout program. We added "(char)toupper(ch)" to convert the value of the "char ch" that is assigned in the ch value. Thus, the program will convert according to what type of converting function.

7. Supplementary Activity

Supplementary 1:

Code Snippet:

```

1  #include <iostream>
2  #include <cctype>
3  using namespace std;
4
5  int main() {
6      char ch;
7      cout << "Enter a character: ";
8      cin >> ch;
9      cout << "-----\n";
10
11     if (isalnum(ch)) {
12         cout << ch << " is alphanumeric\n";
13         cout << "-----\n";
14     }
15     else {
16         cout << ch << " is not alphanumeric\n";
17         cout << "-----\n";
18     }
19
20     if (isalpha(ch)) {
21         cout << ch << " is an alphabet\n";
22         cout << "-----\n";
23     }
24     else {
25         cout << ch << " is not an alphabet\n";
26         cout << "-----\n";
27     }

```

```

28
29     if (isblank(ch)) {
30         cout << "It is a blank space\n";
31         cout << "-----\n";
32     }
33     else {
34         cout << "It is not a blank space\n";
35         cout << "-----\n";
36     }
37
38     if (iscntrl(ch)) {
39         cout << "It is a control character\n";
40         cout << "-----\n";
41     }
42     else {
43         cout << "It is not a control character\n";
44         cout << "-----\n";
45     }
46
47     if (isdigit(ch)) {
48         cout << ch << " is a digit\n";
49         cout << "-----\n";
50     }
51     else {
52         cout << ch << " is not a digit\n";
53         cout << "-----\n";
54     }

```

```

55
56     if (islower(ch)) {
57         cout << ch << " is a lowercase letter\n";
58         cout << "-----\n";
59     }
60     else {
61         cout << ch << " is not a lowercase letter\n";
62         cout << "-----\n";
63     }
64
65     if (isprint(ch)) {
66         cout << ch << " is a printable character\n";
67         cout << "-----\n";
68     }
69     else {
70         cout << ch << " is not a printable character\n";
71         cout << "-----\n";
72     }
73
74     if (ispunct(ch)) {
75         cout << ch << " is a punctuation character\n";
76         cout << "-----\n";
77     }
78     else {
79         cout << ch << " is not a punctuation character\n";
80         cout << "-----\n";
81     }
82

```

```

83     if (isspace(ch)) {
84         cout << "It is a whitespace character\n";
85         cout << "-----\n";
86     }
87     else {
88         cout << "It is not a whitespace character\n";
89         cout << "-----\n";
90     }
91
92     if (isupper(ch)) {
93         cout << ch << " is an uppercase letter\n";
94         cout << "-----\n";
95     }
96     else {
97         cout << ch << " is not an uppercase letter\n";
98         cout << "-----\n";
99     }
100
101     if (isxdigit(ch)) {
102         cout << ch << " is a hexadecimal digit\n";
103         cout << "-----\n";
104     }
105     else {
106         cout << ch << " is not a hexadecimal digit\n";
107         cout << "-----\n";
108     }

```

```

109
110     cout << "Lowercase form of " << ch << " is " << (char)tolower(ch) << "\n";
111     cout << "-----\n";
112     cout << "Uppercase form of " << ch << " is " << (char)toupper(ch) << "\n";
113
114     return 0;
115 }

```

Output/s:

```
Enter a character: q
-----
q is alphanumeric
-----
q is an alphabet
-----
It is not a blank space
-----
It is not a control character
-----
q is not a digit
-----
q is a lowercase letter
-----
q is a printable character
-----
q is not a punctuation character
-----
It is not a whitespace character
-----
q is not an uppercase letter
-----
q is not a hexadecimal digit
-----
Lowercase form of q is q
-----
Uppercase form of q is Q
```

```
Enter a character: 4
-----
4 is alphanumeric
-----
4 is not an alphabet
-----
It is not a blank space
-----
It is not a control character
-----
4 is a digit
-----
4 is not a lowercase letter
-----
4 is a printable character
-----
4 is not a punctuation character
-----
It is not a whitespace character
-----
4 is not an uppercase letter
-----
4 is a hexadecimal digit
-----
Lowercase form of 4 is 4
-----
Uppercase form of 4 is 4
-----
-----
Process exited after 2.2 seconds with return value 0
```

Analysis:

- *In this program, we are working with different character functions from the <cctype> library. These functions allow us to test whether the "ch", which is our variable, compares the variable's value according to their definition. Characters in C++ check and compare their properties, such as whether they are letters, digits, punctuation marks, or whitespace. Each function depends on whether the character satisfies the given condition.*

For example:

*isalnum(ch) checks if the character is alphanumeric,
isalpha(ch) checks if it is an alphabet letter,
isdigit(ch) checks if it is a numerical digit,
islower(ch) and isupper(ch) identify lowercase and uppercase letters,
while isspace(ch) checks for whitespace,
and ispunct(ch) identifies punctuation characters.*

In addition, functions like isprint(ch) and iscntrl(ch) determine if the character is printable or a control character, respectively. Another useful function, isxdigit(ch), checks whether the character is a valid hexadecimal digit.

Supplementary 2:

Code Snippet:

```
1  #include <iostream>
2  #include <string>
3  #include <sstream>
4  using namespace std;
5
6  int main() {
7      string s1, s2, s3, s4;
8      int n1 = 0, n2 = 0, n3 = 0, n4 = 0, total;
9
10     cout << "Enter first number string: "; cin >> s1;
11     cout << "Enter second number string: "; cin >> s2;
12     cout << "Enter third number string: "; cin >> s3;
13     cout << "Enter fourth number string: "; cin >> s4;
14
15     // Converting using stringstream
16     stringstream(s1) >> n1;
17     stringstream(s2) >> n2;
18     stringstream(s3) >> n3;
19     stringstream(s4) >> n4;
20
21     total = n1 + n2 + n3 + n4;
22     cout << "The total sum is: " << total << endl;
23     return 0;
24 }
```

Output:

```
Enter first number string: 10
Enter second number string: 54
Enter third number string: 20
Enter fourth number string: 10
The total sum is: 94

-----
Process exited after 10.18 seconds with return value 0
Press any key to continue . . .
```

Analysis:

- In this program, we are asked to create a program where we have to input four strings that represent integers, convert them into integer values, and add them together. To successfully achieve this, the first and common of all programs begins by declaring four string variables (s1, s2, s3, s4), this is where we will be getting the recipients input value to store them as strings, (n1, n2, n3, n4) initialized to zero, since this is where we will be converting the strings into integer values, and a total variable for the sum of all converted string values. The recipient is then prompted to enter four numeric strings, which will be stored in the string variables as what mentioned before in the declaration of variable. Since the inputs are stored as strings, we need to convert them into integers before initializing and processing a sum operation to get the total of all values. For this purpose, "the program uses the stringstream class, which allows conversion of string values into integer values by streaming the string into an integer variable." After converting all four string inputs into integers, the program will now initialize the sum operation;

For visuals:

$$\text{Sum} = (n1 + n2 + n3 + n4)$$

and stores the result in total. Lastly, we will be displaying the sum using the cout statement. The end!

8. Conclusion

In my conclusion, I was able to identify and clarify further the functionality of the character functions, and how each function differs or work together to identify what type of variable value is declared in a given function. This shows me a lot of ways to create new programs where I can apply it in such a security program where we can still accept where it is in uppercase or lowercase etc. I think I did well, I was able to apply the functions very well and showed it with the output and sufficient examples, still I seek to develop and improve on my area.