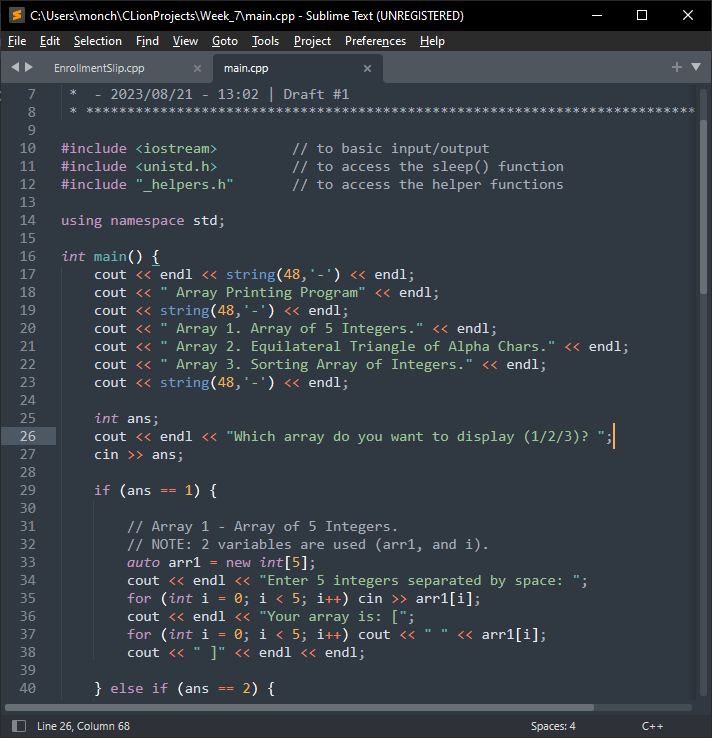
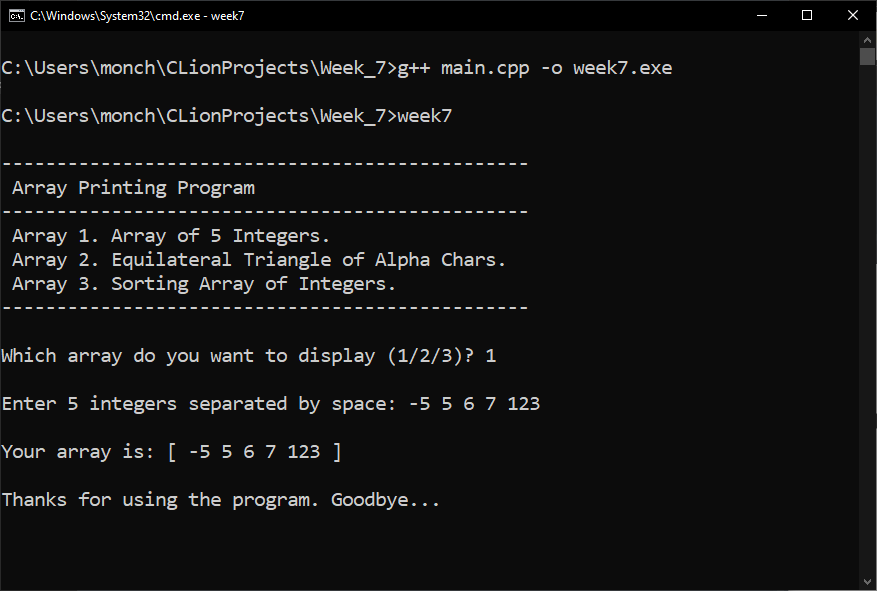
**Problem 01**

Write a program that will accept five (5) integers and display them to the users. You are limited to using only two (2) variables (including the array).

**Solution and Testing 01**

Below is the screenshot of the solution program as well as testing in the terminal.





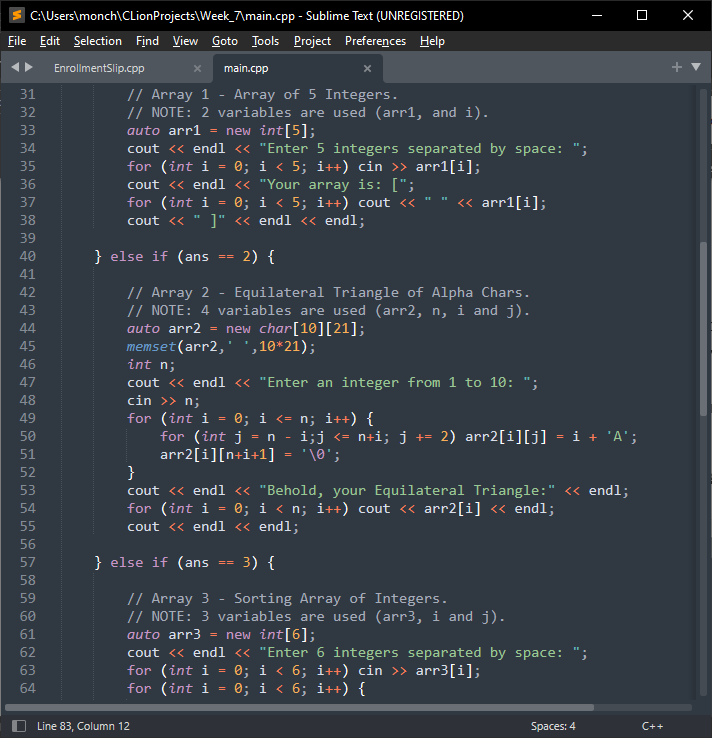
**Problem 02**

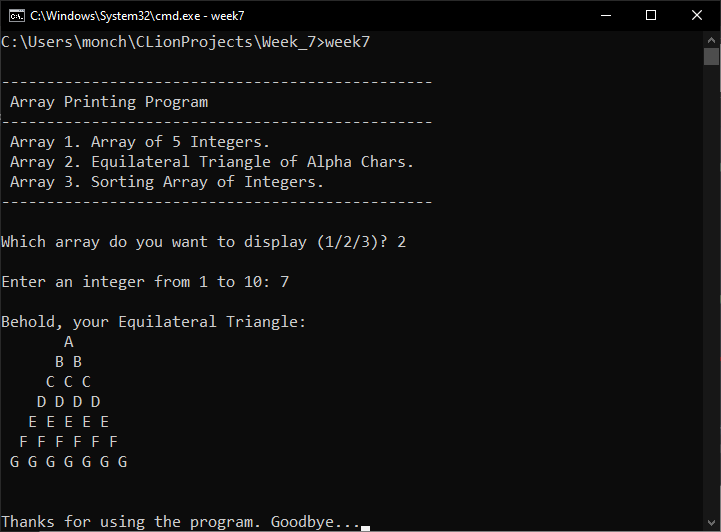
Write a program that will display an equilateral triangle with a height depending on the user. The minimum height is 1, the maximum height is 10. Use an array to display the specific character on the specific row. The array will be: { 0 := “A”, 1:= “B”, 2 := “C”, 3 := “D”, 4 := “E”, 5 := “F”, 6 := “G”, 7 := “H”, 8 := “I”, 9 := “J” }. You are limited to four (4) variables only (including the array).

**Solution and Testing 02**

It is not clear what type of array is being asked here. We shall assume that the array required is the equilateral triangle itself. That means it’s a 2-D array. We create a 2-D char array and fill it with blank spaces. So that we don’t need to do 2 double for-loops, we shall fill the end of each row with c-string terminating character (\0). This will trick cout to consider each row as a c-string and print it as such.

Below is the screenshot of the solution program as well as testing in the terminal.





**Problem 03**

Write a program that will ascendingly sort six (6) integers from the user. Use only four (4) variables (including the array).

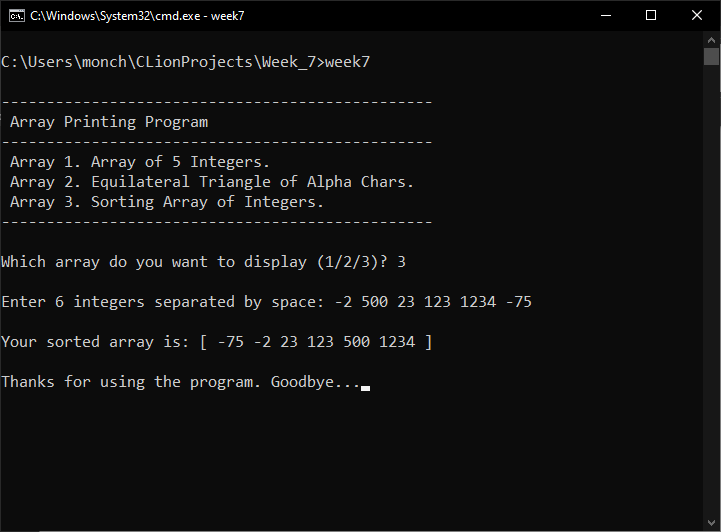
**Solution and Testing 03**

C++ already have a built-in sort method in the namespace std, called sort(). We can apply that method here to make this program relatively shorter. However, we chose to just use the algorithm provided in the module. There is a slight simplification, though. In the swap algorithm portion, since we are using integers, there is NO need to introduce a temporary variable. This saves us one memory space reducing the number of variables from 4 to 3.

Below is the screenshot of the solution program as well as testing in the terminal.

A computer screen shot of a program code

Description automatically generated



**Problem 04**

What can you conclude from this activity?

**Answer 04**

The solutions presented may not be the exact solutions the problem creator had in mind. But the solutions work and do **satisfy the problem specification**. There is always more than one way to solve a programming problem.

With regards to our comment in problem 3, if we had just used C++ built-in method, we would have made the program shorter. Not only that, but built-in functions are also normally more efficient than what we can possibly design ourselves, having passed many tests from experts. The bottom line is we should use standard library functions whenever possible instead of reinventing.

------------------------

**NOTE FOR THIS PROBLEM:**

Source code can be found on my GitHub page: <https://github.com/rvillamangca/>.