

Your CSCI1320 C++ Practicum will be in your recitation during the week of December 3rd. You can use your computer or a lab computer, a double-sided sheet of paper, and a calculator (not likely needed). You CANNOT use old assignments, textbook, or the internet. Attendance in your assigned section is mandatory (unless you had made prior arrangements for special accommodations). Makeup practicums will only be administered in the case of a documented personal or family emergency.

You can expect the C++ practicum to be of similar nature to the MATLAB practicum you took back in October. You should be able to use all of the programming tools we have learned up to Fall break, including but not limited to:

- use your development environment to write, compile, and run C++ programs
- cin/cout
- variables (data type, arithmetic)
- boolean (conditional) expressions
- shorthand operators
- branching
- loops (for, while, do-while)
- using some predefined functions (rand())
- user defined functions
 - pass-by-value
 - pass-by-reference
 - pass-by-array
- driver programs

IF YOUR CODE DOES NOT COMPILE in with a standard g++, you will automatically get a grade no higher than **50% on the entire practicum.**

Some sample problems are given on following pages.

Sample A

Task

Write a C++ program that accomplishes the following tasks:

1. Write a function that finds the average of current element and the next element in given array named as “elements” and stores the new found averages into another array called “averageArray”. Note: For the last element you can just divide the value of last element by 2.
2. Write a function that takes both the given array and average array you just created along with a variable “productOfArrays” using **pass by reference** to return the sum of products of each element from both arrays. Have a look at the example at end of description to understand the logic better.
3. Print the value of “productOfArrays” as shown in example below. If the value of “productOfArrays” is more than 1000, print the averageArray array with each element separated by space. Else, print the elements of averageArray array, one per line.

Note: you CANNOT use global variables. You can only use iostream and string libraries.

Given C++ code (you can copy this into your shell):

```
#include <iostream>
using namespace std;

int main()
{
    int elements[]={5,10,11,25};
    int productOfArrays=0;
    return 0;
}
```

Upon running your completed program, your console output should look like this:

```
Sum of products is :633
The average values array is:
7
10
18
12
```

Sample B

Task

Write a C++ function that accomplishes the following tasks:

1. Finds the maximum ASCII sum in an array of strings.
 - a. **Hint:** Initialize your `max` as `-1` (ASCII values are positive)
2. Returns a boolean `TRUE` if the maximum sum is `> 500`.
3. Marks the array as "read" by placing a `"DONE"` in the last element.

Print the `TRUE` or `FALSE` result to the screen **in your `main()`**.

*Note: you CANNOT use global variables. You can only use `iostream` and `string` libraries. From the `string` library you can only use the **`size()`** function.*

Given C++ code (you can copy this into your shell):

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    string words[] = {"This", "is", "an", "array", "of", "strings"};
    int num_words = 6;
    return 0;
}
```

Upon running your completed program, your console output should look like this:

The max was greater than 500 **(For true)**

OR

The max was not greater than 500 **(For false)**

Sample C

Name:

Signature:

Task

Write a C++ function that accomplishes the following tasks:

1. Finds the average ASCII sum in an array of strings.
2. Returns a boolean TRUE if the average sum is > 500.
3. Marks the array as “read” by placing a “DONE” in the last element.

Print the TRUE or FALSE result to the screen **in your main()**.

*Note: you CANNOT use global variables. You can only use iostream and string libraries. From the string library you can only use the **size()** function.*

Given C++ code (you can copy this into your shell):

```
#include <iostream>
#include <string>
using namespace std;
int main()
{
    string words[] = {"This", "is", "an", "array", "of", "strings"};
    int num_words = 6;

    return 0;
}
```

Upon running your completed program, your console output should look like this:

The average was greater than 500 **(For true)**

OR

The average was not greater than 500 **(For false)**

Section D

Task

Write a C++ program that detects if a person is shouting or not, and warns the user about their rude behaviour, while changing the tone of the message. Here are the steps to accomplish this:

1. Take an input word from the user.
2. Traverse through each character of the string and check if the alphabets are in upper case or not. (Hint: ASCII values of uppercase letters (A to Z) are in the range 65 to 90).
3. If you notice that there are any upper case letters but the first one, display a message in the terminal saying "That's rude, stop shouting". Convert this string of characters to lower case by adding 32 to their ASCII values, and display the more polite message.
4. If all the characters are lowercase, convert the first letter to uppercase by subtracting 32 and display the new message with the prompt "We appreciate your polite tone, but the grammatically correct message would be :". and print your corrected message.
5. Otherwise just print "Your message has been sent."

Note: you CANNOT use global variables. You can only use iostream and string libraries.

Upon running your completed program, your console output should look like this:

CASE 1:

```
Enter your message: DOGS
```

```
That's rude, stop shouting. Here's how your message should look: Dogs
```

CASE 2:

```
Enter your message: dogs
```

```
We appreciate your polite tone, but the grammatically correct message  
would be : Dogs
```

CASE 3:

```
Enter your message: Dogs  
Your message has been sent.
```