

CS211 Lab Assignment - 7

Assigned: 11/04/2019

Due: 11/14/2019 (Midnight)

Write a C++ program that takes an input string from the user and prints out the addresses of the first and last index of the input string. Additionally, print out the difference between the addresses.

You are NOT Allowed to Use any Predefined Functions from any string Library. Only #include <iostream> is allowed as an include library. Nothing else. {using namespace std; is allowed}

You will need to use pointers to store the input string. Assume a maximum string length of 1000.

Your code will be tested on other test cases not listed here. Make sure your output matches the formatting as shown in the test cases below.

Please properly comment your code before submission.

Name your source file as **PrintStringAddresses_WSUID.cpp**. For example, if your user ID is A999B999 name your file as **PrintStringAddresses_A999B999.cpp**.

Sample Test Cases (your addresses may vary):

<u>Test Case 1:</u> Enter Input String: How are you? Start address: 00E52FF8 End address: 00E53003 00E53003 - 00E52FF8 := 11	<u>Test Case 2:</u> Enter Input String: London bridge is falling down Start address: 00DE2EC8 End address: 00DE2EE4 00DE2EE4 - 00DE2EC8 := 28
--	---

Hint:

cout behaves differently for character pointers and integer pointers

```
int* arr = new int[3];
arr[0] = 2;
arr[1] = 4;
arr[2] = 6;
cout << arr << endl;    //prints the first address correctly //00D530E9
```

However,

```
char* carr = new char[3];
carr[0] = 'a';
carr[1] = 'b';
carr[2] = '\0';
cout << carr << endl;    //prints the string "ab" instead of the address
```

In order to make cout print out the address of the string, we need to cast it to a **(void *)** pointer type. A **(void *)** type value is similar to an integer type pointer and is often used in such cases.

Casting to void* is very simple. Rewriting the above code for character pointers:

CS211 Lab Assignment - 7

Assigned: 11/04/2019

Due: 11/14/2019 (Midnight)

```
char* carr = new char[3];
carr[0] = 'a';
carr[1] = 'b';
carr[2] = '\0';
cout<< (void *)carr<<endl;    //prints the string "ab" instead of the address
```

Getting the difference between two pointers is exactly the same as getting the difference between integers.

```
char* c = new char[1000];
cin.getline(c, 1000);
c1 = &c[5];
c2 = &c[10];
int diff = c2 - c1;
```

The above code will get you the difference between addresses of the 6th and the 10th element of the input string.