

SFT 221

## Workshop 1

Name: SHU-TING, HSU  
ID: 133505222  
Email: [chsu17@myseneca.ca](mailto:chsu17@myseneca.ca)  
Section: NFF

### Authenticity Declaration:

I declare this submission is the result of my own work and has not been shared with any other student or 3rd party content provider. This submitted piece of work is entirely of my own creation.

Test #	Test Description	Data Input
1	If buf[20] is 20 character	char s1[] = { "upended" }; char prefix[] = { "upendedABCDEFGHijkl" }; char suffix[] = { "ed" };
2	If buf[20] is over 20 character	char s1[] = { "upended" }; char prefix[] = { "upendedABCDEFGHijklK" }; char suffix[] = { "ed" };
3	If nothing inside the arrays	char s1[] = { "upended" }; char prefix[] = { "" }; char suffix[] = { "ed" };
4	Sensitivity of character	char s1[] = { "upended" }; char prefix[] = { "UP" }; char suffix[] = { "ed" };
5	If the input is number	char s1[] = { "upended" }; char prefix[] = { "123" }; char suffix[] = { "ed" };
6	if the input have space	char s1[] = { "upended" }; char prefix[] = { "up" }; char suffix[] = { " " };
7	If the input is symbol	char s1[] = { "upended" }; char prefix[] = { "up" }; char suffix[] = { "%/&" };

## Expected result

upended does not start with upendedABCDEFGHIJKL  
upended does end with ed

error

Display error message that Field content cannot be blank

upended does not start with UP  
upended does end with ed

upended does not start with 123  
upended does end with ed

upended does start with up  
upended does not end with

upended does start with up  
upended does not end with %/&

Actual result	Result (Fail / Pass)
if unexpected upended does not start with upendedABCDEFGHIJKL upended does end with ed	PASS
throw the error message about the instrumentation code being an out of range array access.	Fail (Bugs)
upended does start with upended does end with ed	Fail (Bugs)
upended does start with UP upended does end with ed	Fail (Bugs)
upended does not start with 123 upended does end with ed	PASS
upended does not start with up upended does not end with	Fail (Bugs)
upended does start with up upended does not end with %/&	PASS

## Fixed Code Explanation

None

According to the length of the array of s, determine the space where the buff can be placed.

```
char buf[20];
```

**change to**

```
int length = strlen(s);
```

```
char buf[length];
```

Display error message that Field content cannot be blank

It shouldn't be sensitivity of character

None

Display error message that Field content cannot consist only of spaces.

None

A reflection where you consider whether testing or inspection identified more bugs in this case. State why you think one way worked better than the other. How could you improve the technique that worked less well?

In this case, testing proved to be the more effective method for identifying bugs. Testing allowed for the dynamic execution of the code, uncovering runtime errors and revealing issues that might not have been apparent through inspection alone. When I initially attempted to inspect the code without a deep understanding, I missed some subtle logical errors. However, through testing, I was able to simulate different scenarios and uncover bugs more systematically.

To improve the inspection technique, I could enhance my understanding of the codebase before attempting a detailed review. Comprehensive documentation or code comments could aid in this process. Additionally, peer reviews where multiple team members inspect the code collectively can enhance the effectiveness of inspections by bringing in diverse perspectives and expertise.

Nevertheless, it's essential to strike a balance between testing and inspection, as both approaches complement each other in ensuring software quality.

Did you find it difficult to find the bugs in this assignment? If not, what helped find the quickly? If you did find it difficult, what made finding the bugs so difficult?

Yes, it was hard to find the bugs at first. But trying to spot mistakes before I really understood how the code worked wasn't the right way to go about it. Later, after I fully understood how the program ran, and after giving it some more thought, I managed to spot the bugs. Understanding the process made a big difference in finding and fixing the errors. It taught me that a solid understanding of the code is essential for effective debugging.