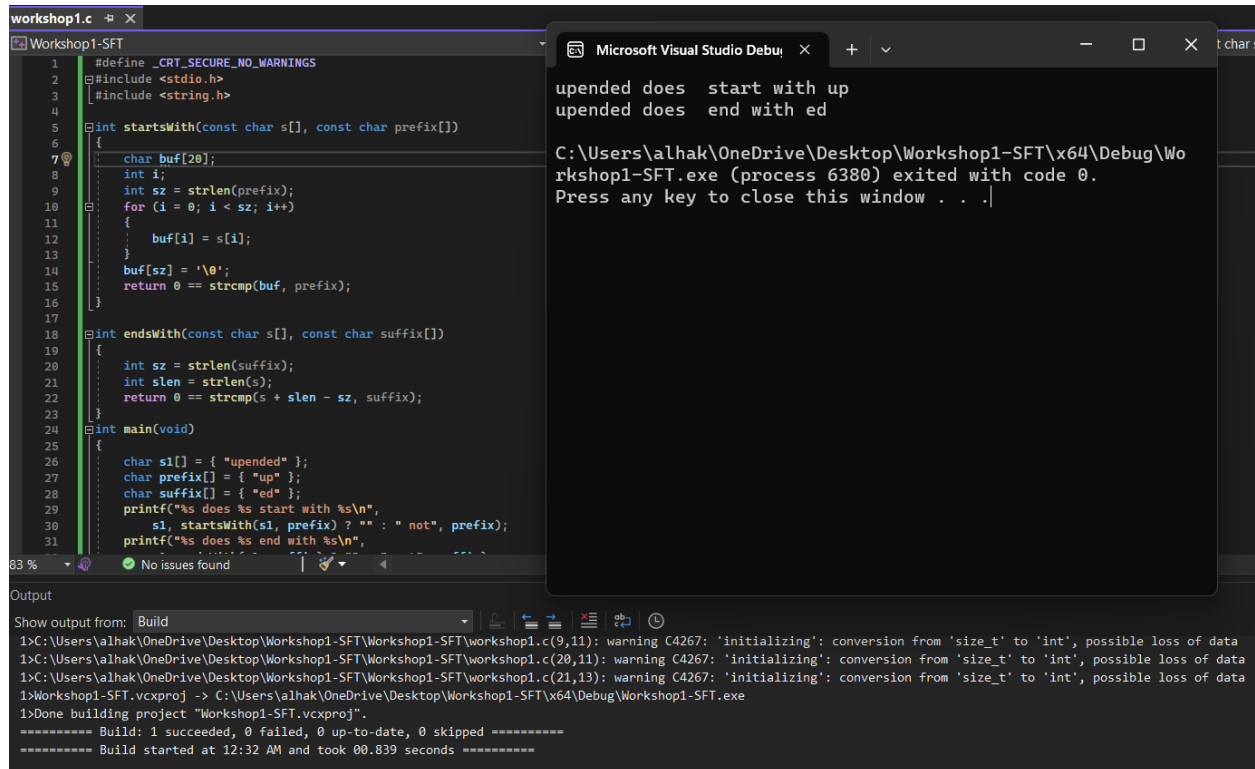


SFT221 – Workshop 1

The image below shows the output of the source code without any changes:



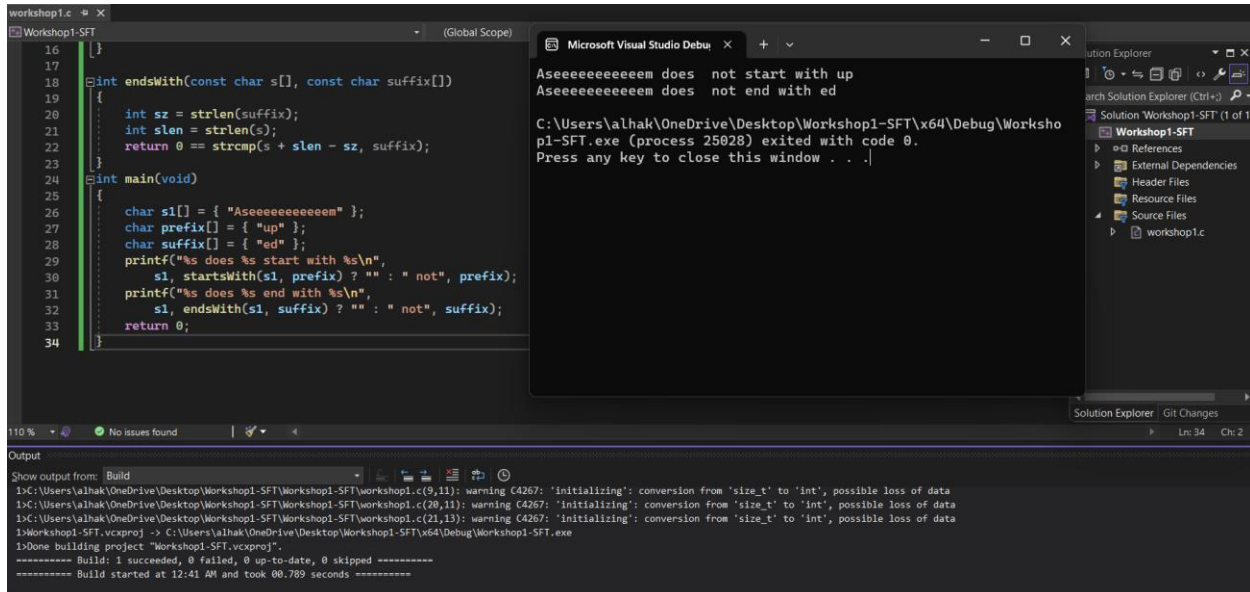
```
workshop1.c
1  #define _CRT_SECURE_NO_WARNINGS
2  #include <stdio.h>
3  #include <string.h>
4
5  int startsWith(const char s[], const char prefix[])
6  {
7      char buf[20];
8      int i;
9      int sz = strlen(prefix);
10     for (i = 0; i < sz; i++)
11     {
12         buf[i] = s[i];
13     }
14     buf[sz] = '\0';
15     return 0 == strcmp(buf, prefix);
16 }
17
18 int endsWith(const char s[], const char suffix[])
19 {
20     int sz = strlen(suffix);
21     int slen = strlen(s);
22     return 0 == strcmp(s + slen - sz, suffix);
23 }
24
25 int main(void)
26 {
27     char s1[] = { "upended" };
28     char prefix[] = { "up" };
29     char suffix[] = { "ed" };
30     printf("%s does %s start with %s\n",
31           s1, startsWith(s1, prefix) ? "" : " not", prefix);
32     printf("%s does %s end with %s\n",
33           s1, endsWith(s1, suffix) ? "" : " not", suffix);
34 }
```

```
Microsoft Visual Studio Debug Console
upended does start with up
upended does end with ed

C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\x64\Debug\Wo
rkshop1-SFT.exe (process 6380) exited with code 0.
Press any key to close this window . . .
```

```
Output
Show output from: Build
1>C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\Workshop1-SFT\workshop1.c(9,11): warning C4267: 'initializing': conversion from 'size_t' to 'int', possible loss of data
1>C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\Workshop1-SFT\workshop1.c(20,11): warning C4267: 'initializing': conversion from 'size_t' to 'int', possible loss of data
1>C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\Workshop1-SFT\workshop1.c(21,13): warning C4267: 'initializing': conversion from 'size_t' to 'int', possible loss of data
1>Workshop1-SFT.vcxproj -> C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\x64\Debug\Workshop1-SFT.exe
1>Done building project "Workshop1-SFT.vcxproj".
***** Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped *****
***** Build started at 12:32 AM and took 00.839 seconds *****
```

My first test was on line 26 when I tried to change s1 to a different word that does not start with up and does not end with ed, which doesn't match prefix and suffix, and this was the output:



The screenshot shows the Visual Studio IDE with a C program in `workshop1.c` and its output in the Debug Console. The program defines a function `endsWith` and a `main` function. In `main`, a string `s1` is set to `"Aseeeeeeeeeem"`, a prefix `prefix` is `"up"`, and a suffix `suffix` is `"ed"`. The program prints whether `s1` starts with `prefix` and ends with `suffix`. The output shows that `s1` does not start with `up` and does not end with `ed`. The build output shows several warnings about conversions from `size_t` to `int`.

```
16 }
17
18 int endsWith(const char s[], const char suffix[])
19 {
20     int sz = strlen(suffix);
21     int slen = strlen(s);
22     return 0 == strcmp(s + slen - sz, suffix);
23 }
24
25 int main(void)
26 {
27     char s1[] = { "Aseeeeeeeeeem" };
28     char prefix[] = { "up" };
29     char suffix[] = { "ed" };
30     printf("%s does %s start with %s\n",
31           s1, startsWith(s1, prefix) ? "" : "not", prefix);
32     printf("%s does %s end with %s\n",
33           s1, endsWith(s1, suffix) ? "" : "not", suffix);
34     return 0;
35 }
```

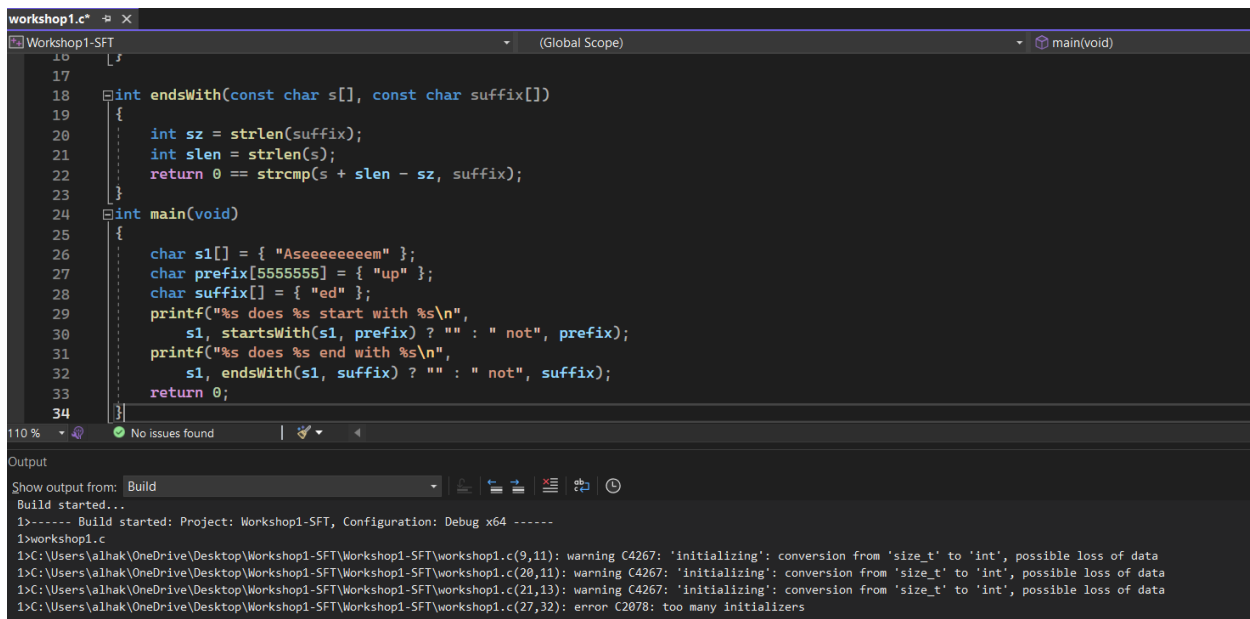
Microsoft Visual Studio Debug Console:

```
Aseeeeeeeeeem does not start with up
Aseeeeeeeeeem does not end with ed
C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\x64\Debug\Workshop1-SFT.exe (process 25028) exited with code 0.
Press any key to close this window . . .
```

Output:

```
Show output from: Build
1>C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\Workshop1-SFT\workshop1.c(9,11): warning C4267: 'initializing': conversion from 'size_t' to 'int', possible loss of data
1>C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\Workshop1-SFT\workshop1.c(20,11): warning C4267: 'initializing': conversion from 'size_t' to 'int', possible loss of data
1>C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\Workshop1-SFT\workshop1.c(21,13): warning C4267: 'initializing': conversion from 'size_t' to 'int', possible loss of data
1>Workshop1-SFT.vcxproj -> C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\x64\Debug\Workshop1-SFT.exe
1>Done building project "Workshop1-SFT.vcxproj".
----- Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped -----
----- Build started at 12:41 AM and took 00.789 seconds -----
```

For my second test, I tried to change the array size of the buffer [20] to a different number in line 27 and the app crashed, and this would lead to a buffer overflow, so I got errors multiple times, and this was the output:



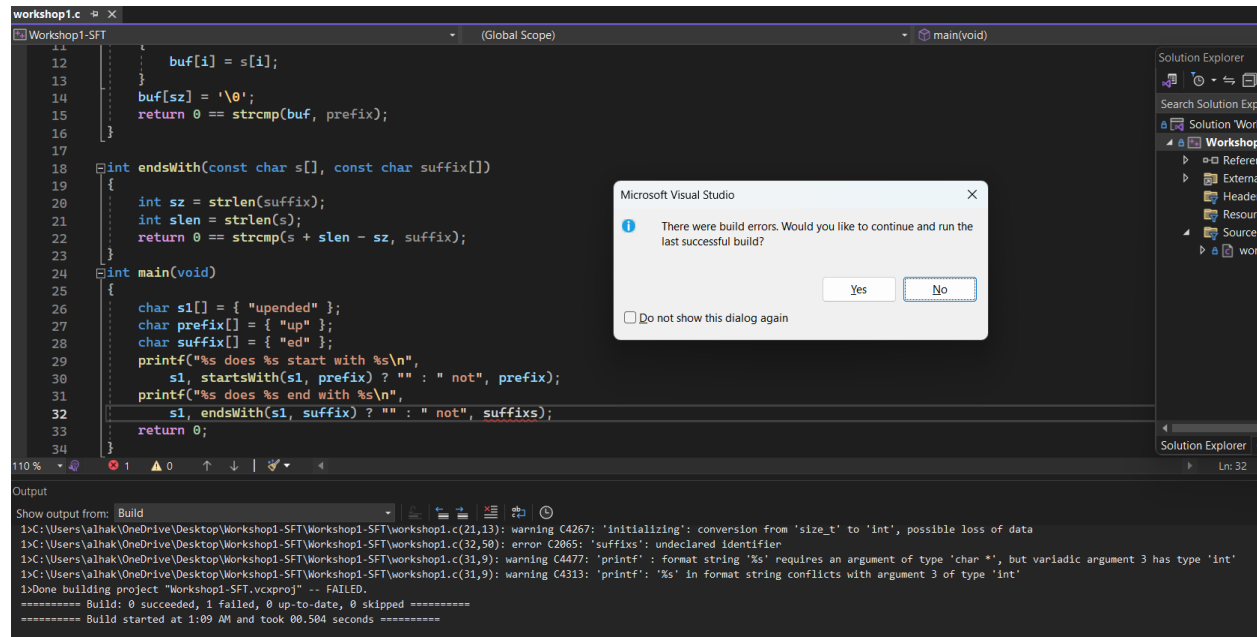
The screenshot shows the Visual Studio IDE with the same C program as before, but with the `prefix` array size changed to 55 in line 27. The program is run in Debug mode, and the output shows the same results as before. However, the build output shows a new error: `error C2078: too many initializers` at line 27, column 32. This is because the array `prefix` is declared with a size of 55, but only 2 characters are initialized.

```
16 }
17
18 int endsWith(const char s[], const char suffix[])
19 {
20     int sz = strlen(suffix);
21     int slen = strlen(s);
22     return 0 == strcmp(s + slen - sz, suffix);
23 }
24
25 int main(void)
26 {
27     char s1[] = { "Aseeeeeeeeeem" };
28     char prefix[5555555] = { "up" };
29     char suffix[] = { "ed" };
30     printf("%s does %s start with %s\n",
31           s1, startsWith(s1, prefix) ? "" : "not", prefix);
32     printf("%s does %s end with %s\n",
33           s1, endsWith(s1, suffix) ? "" : "not", suffix);
34     return 0;
35 }
```

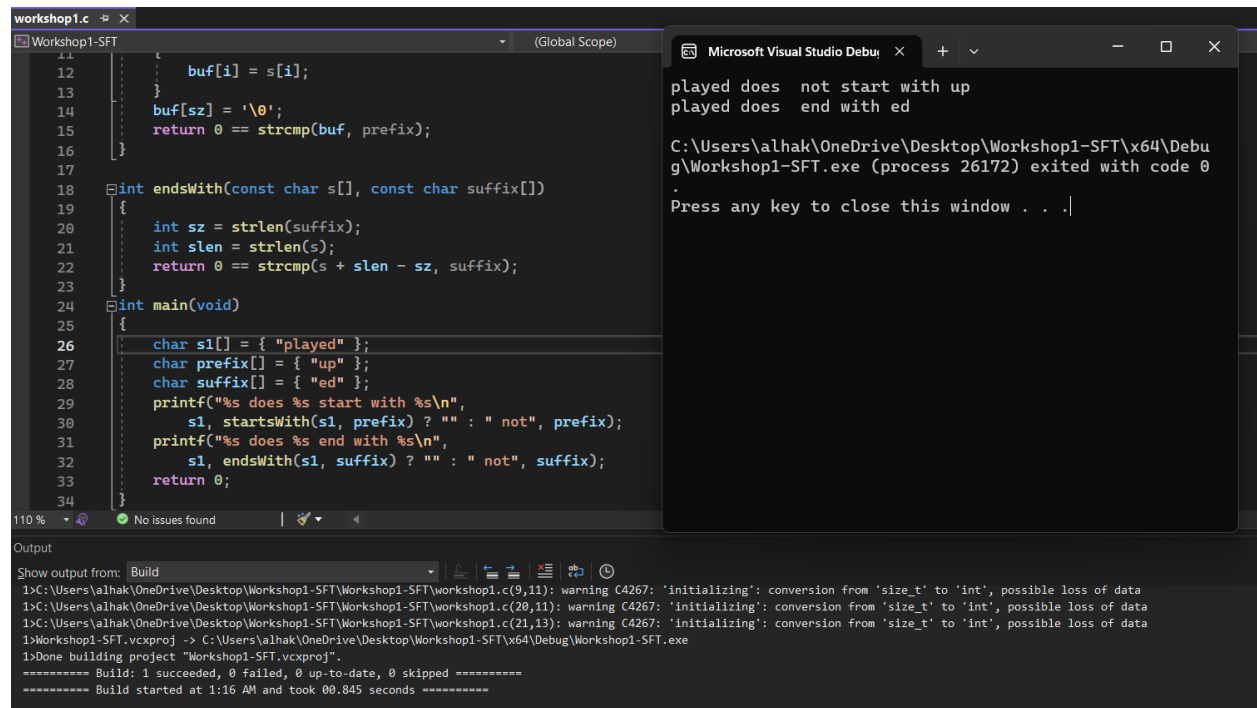
Output:

```
Show output from: Build
Build started...
1>----- Build started: Project: Workshop1-SFT, Configuration: Debug x64 -----
1>workshop1.c
1>C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\Workshop1-SFT\workshop1.c(9,11): warning C4267: 'initializing': conversion from 'size_t' to 'int', possible loss of data
1>C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\Workshop1-SFT\workshop1.c(20,11): warning C4267: 'initializing': conversion from 'size_t' to 'int', possible loss of data
1>C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\Workshop1-SFT\workshop1.c(21,13): warning C4267: 'initializing': conversion from 'size_t' to 'int', possible loss of data
1>C:\Users\alhak\OneDrive\Desktop\Workshop1-SFT\Workshop1-SFT\workshop1.c(27,32): error C2078: too many initializers
```

In my third test, I made a semantic error by putting the wrong variable name in line 32, which caused a bug. Here is an example in the image below:



For the last test, I tried to put in a different word where the ending matches the suffix, but the beginning does not, and this was the output:



Reflection

1. 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 fact, both testing and inspection are often the best ways to find and fix bugs in code. Testing can help find bugs while the program is running, but inspecting can focus on the quality of the code and any problems in the design. But in my previous results, I found that testing was a better way to try out different techniques because I could put in different inputs and see what happened while executing. In general, both methods must be used to make the program more stable and easier to keep up with.

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

- It wasn't hard to find the bugs in my previous tests and scenarios that I provided for the code that was given because it had clear problems and mistakes that were easy to identify. In addition, I have knowledge of some of the basic programming mistakes.

FULL NAME: Asem Al-Hakami

STUDENT ID: 156837205

EMAIL: aal-hakami@myseneca.ca