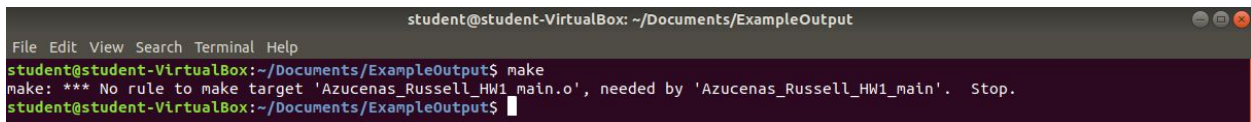


## Assignment 1 - Command Line Arguments

The first assignment, the Command Line Arguments project, consists of a program that displays the total number of arguments passed and lists each argument in the console. Arguments are passed through the 'RUNOPTIONS' parameter when using the 'make run' command.

Firstly, I edited the 'ROOTNAME' variable of the Makefile to correctly show my full name. Since I wasn't sure of the exact purpose of the Makefile at the time, I attempted running the 'make' command and proceeded to get the following error.

(Note: Reproduced error in another folder, not sure if we need direct shots of errors)

A screenshot of a terminal window titled 'student@student-VirtualBox: ~/Documents/ExampleOutput'. The terminal shows the command 'make' being executed. The output is an error message: 'make: \*\*\* No rule to make target 'Azucenas\_Russell\_HW1\_main.o', needed by 'Azucenas\_Russell\_HW1\_main'. Stop.' The prompt 'student@student-VirtualBox:~/Documents/ExampleOutput\$' is visible at the bottom.

```
student@student-VirtualBox: ~/Documents/ExampleOutput
File Edit View Search Terminal Help
student@student-VirtualBox:~/Documents/ExampleOutput$ make
make: *** No rule to make target 'Azucenas_Russell_HW1_main.o', needed by 'Azucenas_Russell_HW1_main'. Stop.
student@student-VirtualBox:~/Documents/ExampleOutput$
```

To solve this issue, I went on Slack to see if anyone has already ran into this issue and talked about it in the CSC415 channel. From what I saw, I figured out that the Makefile required a '.c' file with the same name as the decided 'ROOTNAME' (and must have a main function). After creating the required file, the current issue disappeared and it was time to write the program's code.

Outside of getting familiar with the C language and finding basic utility functions like 'printf()' (after including 'stdio.h'), the only issue I had was finding out how to receive the 'ROOTNAME' arguments through the main function of the program. After a few Google searches, I found out that the main function can receive two parameters: 'int argc' and 'char \*argv[]'. It seems that 'argc' represents the total argument count and 'argv' represents a pointer array containing the list of arguments.

```
int main(int argc, char *argv[]) {
```

With that knowledge in mind, I completed the assignment containing the correct expected output.

```
student@student-VirtualBox: ~/Documents/assignment-1-command-line-arguments-Ninjaruss
File Edit View Search Terminal Help
student@student-VirtualBox:~/Documents/assignment-1-command-line-arguments-Ninjaruss$ make run RUNOPTIONS="arg1 arg2 CSC 415 4 2 0"
gcc -c -o Azucenas_Russell_HW1_main.o Azucenas_Russell_HW1_main.c -g -I.
gcc -o Azucenas_Russell_HW1_main Azucenas_Russell_HW1_main.o -g -I.
./Azucenas_Russell_HW1_main arg1 arg2 CSC 415 4 2 0
Argument #0: ./Azucenas_Russell_HW1_main
Argument #1: arg1
Argument #2: arg2
Argument #3: CSC
Argument #4: 415
Argument #5: 4
Argument #6: 2
Argument #7: 0
student@student-VirtualBox:~/Documents/assignment-1-command-line-arguments-Ninjaruss$
```