

Assignment 1 – Command Line Arguments

Description:

The objective of this project is to utilize the C language for command-line parameter handling. It involves showcasing each received parameter on the terminal and concluding by displaying the total count of parameters.

Approach:

I use C language to complete this project. Preprocessor directives, which include the necessary header files for input/output operations (stdio.h) and dynamic memory allocation functions (stdlib.h).

The main function, which is the entry point of a C program. It takes two parameters: argc (an integer representing the number of command-line arguments) and argv (an array of strings representing the actual command-line arguments).

Uses printf to print the number of command-line arguments (argc) to the console.

Declares an integer variable i and initializes it to 0. This variable will be used as a counter in the upcoming while loop.

Using a while loop that will iterate as long as the value of i is less than the number of command-line arguments (argc).

Inside the loop, using printf to print the current command-line argument (argv[i]) along with its index (i). The %.2d format specifier ensures that the index is printed with at least two digits.

After printing the current argument, i is incremented to move to the next argument in the next iteration of the loop.

Finally, returns 0 from the main function, indicating successful completion of the program. A return value of 0 conventionally signifies that the program executed without errors.

Issues and Resolutions:

My first issue was typing "argc" inside the quote mark led to a compile error. I resolved it by deleting the "argc" inside the quote mark. Then the code will successfully compile.

Analysis: (If required for the assignment)

Screen shot of compilation:

```
student@student:~/csc415-assignment-1-eric915c$ make
gcc -c -o Chen_PoHan_HW1_main.o Chen_PoHan_HW1_main.c -g -I.
gcc -o Chen_PoHan_HW1_main Chen_PoHan_HW1_main.o -g -I.
student@student:~/csc415-assignment-1-eric915c$
```

Screen shot(s) of the execution of the program:

```
student@student:~/csc415-assignment-1-eric915c$ make run RUNOPTIONS="Hello, these are overridden options 3 6 9"
gcc -c -o Chen_PoHan_HW1_main.o Chen_PoHan_HW1_main.c -g -I.
gcc -o Chen_PoHan_HW1_main Chen_PoHan_HW1_main.o -g -I.
./Chen_PoHan_HW1_main Hello, these are overridden options 3 6 9
9 arguments on command line.

argument00:./Chen_PoHan_HW1_main
argument01:Hello,
argument02:these
argument03:are
argument04:overridden
argument05:options
argument06:3
argument07:6
argument08:9

student@student:~/csc415-assignment-1-eric915c$ make clean
```