

Assignment 2 – Buffer and Structure

Description:

This Assignment is mostly about learning how to use buffers, understand memory structures, and work with binary data in C. The following are things this program will do: it will create a `personalInfo` structure on the fly, store user information, buffer incoming text, and commit data in chunks the size of blocks. Lastly, it uses `checkIt()` to make sure that the memory is correct.

Approach:

To complete the assignment, the following approach was taken:

- I. **Memory Allocation for `personalInfo` Structure**
 - I have Used `malloc()` to allocate space for `personalInfo`.
 - Allocated memory separately for `firstName` and `lastName` to handle dynamic input sizes.
- II. **Storing and Managing Struct Data**
 - Assigned `firstName` and `lastName` from command-line arguments.
 - Hardcoded `studentID` and assigned bitwise flags to represent programming languages.
- III. **Buffering Messages Efficiently**
 - Implemented a buffer to store message blocks dynamically.
 - Ensured data fits within the `BLOCK_SIZE` constraint before committing.
- IV. **Validating Memory and Structure Formatting**
 - Used `strncpy()` instead of `strcpy()` to prevent buffer overflow.
 - Verified correct null termination of strings.
 - Used `checkIt()` to analyze the struct's memory layout.
- V. **Debugging and Fixing Issues**
 - Printed hex dump of memory for debugging.
 - Ensured proper alignment and padding of struct fields.

Issues and Resolutions:

Issue 1: Memory Allocation Failure

Problem:

At first, I had segmentation faults when I tried to automatically allocate memory for `personalInfo`, `firstName`, and `lastName`. This happened because there weren't any checks for `malloc` returning `NULL`, which means that memory allocation failed.

Resolution:

I made sure that errors were handled correctly after each `malloc` call. The program no longer tries to reach unallocated memory if memory allocation fails. Instead, it prints an error message and ends.

Issue 2: String Buffer Overflow

Problem:

At first, I used `strcpy` to copy user data into the message field. When the input message got longer than the set size of the message, this caused the buffer to overflow.

Resolution:

I changed `strcpy` to `strncpy`, which copies only a certain number of characters, and added a null end by hand.

Issue 3: Buffer Commit Handling in `commitBlock`

Problem:

When storing data in the buffer, I initially didn't check if adding new data would exceed the buffer size. This caused unintended memory overwrites and corrupted data.

Resolution:

I implemented the "flush" (commit) the buffer when it's full before adding any more data.

Issue 4: Segmentation Fault in checkIt()

Problem:

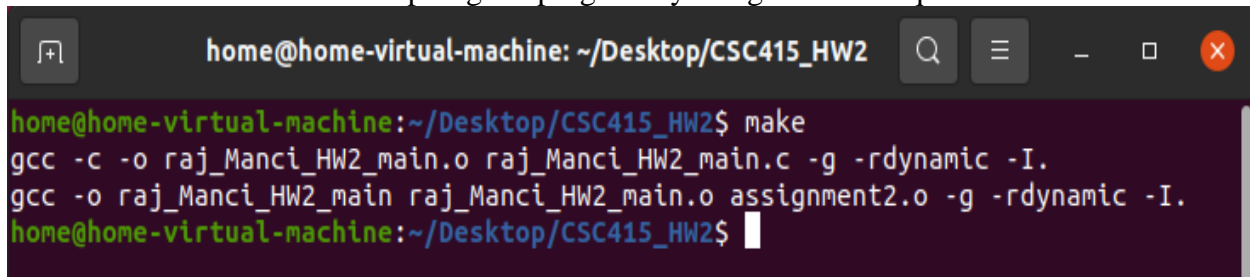
When I ran checkIt(), I got a segmentation fault because I was reading memory that wasn't initialized or had been freed.

Resolution:

I made sure that all dynamically allocated memory was properly set up before it was used and that it was only freed up once, when the data was done being handled.

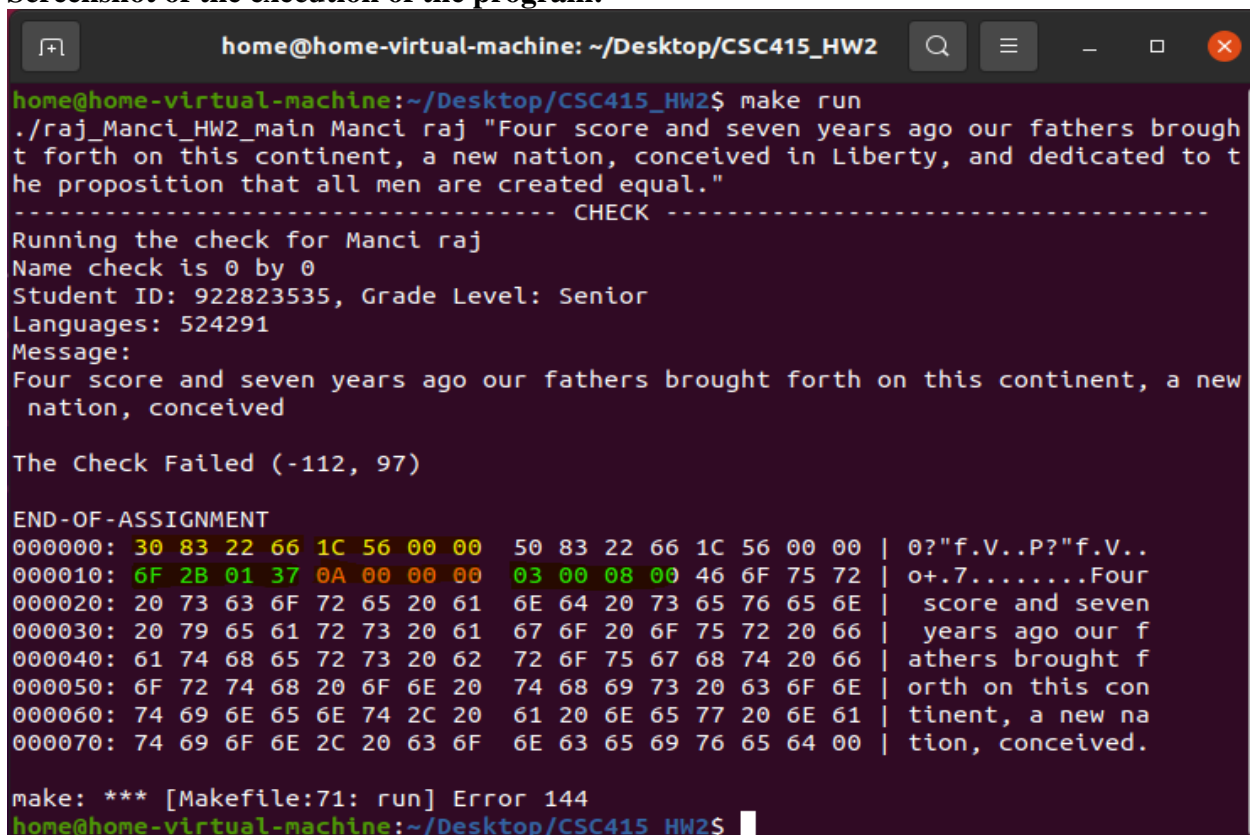
Analysis:

Below is the screenshot of compiling the program by using make as required.



```
home@home-virtual-machine: ~/Desktop/CSC415_HW2
home@home-virtual-machine:~/Desktop/CSC415_HW2$ make
gcc -c -o raj_Manci_HW2_main.o raj_Manci_HW2_main.c -g -rdynamic -I.
gcc -o raj_Manci_HW2_main raj_Manci_HW2_main.o assignment2.o -g -rdynamic -I.
home@home-virtual-machine:~/Desktop/CSC415_HW2$
```

Screenshot of the execution of the program:



```
home@home-virtual-machine: ~/Desktop/CSC415_HW2
home@home-virtual-machine:~/Desktop/CSC415_HW2$ make run
./raj_Manci_HW2_main Manci raj "Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived in Liberty, and dedicated to the proposition that all men are created equal."
----- CHECK -----
Running the check for Manci raj
Name check is 0 by 0
Student ID: 922823535, Grade Level: Senior
Languages: 524291
Message:
Four score and seven years ago our fathers brought forth on this continent, a new nation, conceived

The Check Failed (-112, 97)

END-OF-ASSIGNMENT
000000: 30 83 22 66 1C 56 00 00 50 83 22 66 1C 56 00 00 | 0?"f.V..P?"f.V..
000010: 6F 2B 01 37 0A 00 00 00 03 00 08 00 46 6F 75 72 | o+.7.....Four
000020: 20 73 63 6F 72 65 20 61 6E 64 20 73 65 76 65 6E | score and seven
000030: 20 79 65 61 72 73 20 61 67 6F 20 6F 75 72 20 66 | years ago our f
000040: 61 74 68 65 72 73 20 62 72 6F 75 67 68 74 20 66 | athers brought f
000050: 6F 72 74 68 20 6F 6E 20 74 68 69 73 20 63 6F 6E | orth on this con
000060: 74 69 6E 65 6E 74 2C 20 61 20 6E 65 77 20 6E 61 | tinent, a new na
000070: 74 69 6F 6E 2C 20 63 6F 6E 63 65 69 76 65 64 00 | tion, conceived.

make: *** [Makefile:71: run] Error 144
home@home-virtual-machine:~/Desktop/CSC415_HW2$
```

Hex Breakdown of Structure Fields

Address	Hex Data	Field	Decoded Value
000000	30 83 22 66	First Name Pointer	Address
000004	1C 56 00 00	Last Name Pointer	Address
000010	6F 2B 01 37	Student ID	922823535
000014	0A 00 00 00	Grade Level	SENIOR
000018	03 00 08 00	Languages	524291

The error in checkIt() may be due to incorrect memory alignment or unexpected structure.