

Author: Cary Wu

RUN 1: first three letters of your first name, first three letters of your last name

table 1

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0xf0	0xe0	0xff	0xff	0xff	0x7f	0x00	0x00
char representation	-	-	-	-	-	-	'\0'	'\0'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
char representation	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0xe0	0x07	0x40	0x00	0x00	0x00	0x00	0x00
char representation	-	-	'@'	'\0'	'\0'	'\0'	'\0'	'\0'

table 2

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0xf0	0xe0	0xff	0xff	0xff	0x7f	0x00	0x00
char representation	-	-	-	-	-	-	'\0'	'\0'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x00	0x00	0x4c	0x00	0x00	0x00	0x00	0x00
char representation	'\0'	'\0'	'L'	'\0'	'\0'	'\0'	'\0'	'\0'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0x10	0x20	0x60	0x00	0x00	0x00	0x00	0x00
char representation	-	' '	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'

table 3

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0x43	0x61	0x72	0x00	0xff	0x57	0x75	0x57
char representation	'C'	'a'	'r'	'\0'	-	'W'	'u'	'W'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x00	0x00	0x4c	0x00	0x00	0x00	0x00	0x00
char representation	'\0'	'\0'	'L'	'\0'	'\0'	'\0'	'\0'	'\0'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0x10	0x20	0x60	0x00	0x00	0x00	0x00	0x00
char representation	-	' '	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'

RUN 1 Explanation

output:

First: Car
Last: WuW
Initial: L
Age: 21
Salary: 70000

how did the program terminate? (normally, segfault, something else?)

normally

was output of the program as expected? if not, try to explain why

Yes, the output of the program is as expected.

RUN 2: first five letters of your first name, first three letters of your last name

table 1

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0xf0	0xe0	0xff	0xff	0xff	0x7f	0x00	0x00
char representation	-	-	-	-	-	-	'\0'	'\0'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
char representation	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0xe0	0x07	0x40	0x00	0x00	0x00	0x00	0x00
char representation	-	-	'@'	'\0'	'\0'	'\0'	'\0'	'\0'

table 2

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0xf0	0xe0	0xff	0xff	0xff	0x7f	0x00	0x00
char representation	-	-	-	-	-	-	'\0'	'\0'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x00	0x00	0x4c	0x00	0x00	0x00	0x00	0x00
char representation	'\0'	'\0'	'L'	'\0'	'\0'	'\0'	'\0'	'\0'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0x10	0x20	0x60	0x00	0x00	0x00	0x00	0x00
char representation	-	' '	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'

table 3

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0x43	0x61	0x72	0x79	0x43	0x57	0x75	0x57
char representation	'C'	'a'	'r'	'y'	'C'	'W'	'u'	'W'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x00	0x00	0x4c	0x00	0x00	0x00	0x00	0x00
char representation	'\0'	'\0'	'L'	'\0'	'\0'	'\0'	'\0'	'\0'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0x10	0x20	0x60	0x00	0x00	0x00	0x00	0x00
char representation	-	' '	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'

RUN 2 Explanation

output:

First: CaryCWuW
Last: WuW
Initial: L
Age: 21
Salary: 70000

how did the program terminate? (normally, segfault, something else?)

normally

was output of the program as expected? if not, try to explain why

No, the output was not as expected. The output for **First:** had the last name inputted on the command line appended to it, so the printed output for **First** was not as expected. The reason for this is because the program assigned `NAME_SIZE` to be 5, meaning First and Last name strings (which are char arrays) can only be 5 characters with the last character in the char `first[]` field of the person struct to be `'\0'`, which is the string terminator. Otherwise, when printing `p.first`, the program will keep reading until it hits a null `'\0'` character telling it to stop. Since not enough space was allocated for a 5 char first name input, there was no `'\0'` separating the field **first** from **last**.

RUN 3: first three letters of your first name, first ten letters of your last name

table 1

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0xf0	0xe0	0xff	0xff	0xff	0x7f	0x00	0x00
char representation	-	-	-	-	-	-	'\0'	'\0'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
char representation	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0xe0	0x07	0x40	0x00	0x00	0x00	0x00	0x00
char representation	-	-	'@'	'\0'	'\0'	'\0'	'\0'	'\0'

table 2

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0xf0	0xe0	0xff	0xff	0xff	0x7f	0x00	0x00
char representation	-	-	-	-	-	-	'\0'	'\0'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x00	0x00	0x4c	0x00	0x00	0x00	0x00	0x00
char representation	'\0'	'\0'	'L'	'\0'	'\0'	'\0'	'\0'	'\0'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0x10	0x20	0x60	0x00	0x00	0x00	0x00	0x00
char representation	-	' '	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'

table 3

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0x43	0x61	0x72	0x00	0xff	0x57	0x75	0x57
char representation	'C'	'a'	'r'	'\0'	-	'W'	'u'	'W'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x75	0x57	0x75	0x57	0x75	0x57	0x75	0x00
char representation	'u'	'W'	'u'	'W'	'u'	'W'	'u'	'\0'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0x10	0x20	0x60	0x00	0x00	0x00	0x00	0x00
char representation	-	' '	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'

RUN 3 Explanation

output:

First: Car
Last: WuWuWuWuWu
Initial: u
Age: 21
Salary: 70000

how did the program terminate? (normally, segfault, something else?)

normally

was output of the program as expected? if not, try to explain why

No, the output of **Initial:** was not as expected because the program sets the initial field to be 'L'. The reason the output was not 'L' is because since for this run, I inputted 10 characters for the last name, yet only 5 bytes are reserved for char last[] in the person struct, the initial field was overwritten by what I inputted to be my last name. Also, since the last name field should only be a string of 4 characters plus '\0', the output for **Last:** with 10 characters was not as expected either. The reason is the same as what happened in run 2, there was no '\0' character to terminate the last name to fit only in 5 bytes.

RUN 4: first three letters of your first name, first eleven letters of your last name

table 1

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0xf0	0xe0	0xff	0xff	0xff	0x7f	0x00	0x00
char representation	-	-	-	-	-	-	'\0'	'\0'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
char representation	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0xe0	0x07	0x40	0x00	0x00	0x00	0x00	0x00
char representation	-	-	'@'	'\0'	'\0'	'\0'	'\0'	'\0'

table 2

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0xf0	0xe0	0xff	0xff	0xff	0x7f	0x00	0x00
char representation	-	-	-	-	-	-	'\0'	'\0'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x00	0x00	0x4c	0x00	0x00	0x00	0x00	0x00
char representation	'\0'	'\0'	'L'	'\0'	'\0'	'\0'	'\0'	'\0'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0x10	0x20	0x60	0x00	0x00	0x00	0x00	0x00
char representation	-	' '	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'

table 3

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0x43	0x61	0x72	0x00	0xff	0x57	0x75	0x57
char representation	'C'	'a'	'r'	'\0'	-	'W'	'u'	'W'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x75	0x57	0x75	0x57	0x75	0x57	0x75	0x57
char representation	'u'	'W'	'u'	'W'	'u'	'W'	'u'	'W'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0x00	0x20	0x60	0x00	0x00	0x00	0x00	0x00
char representation	'\0'	' '	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'

RUN 4 Explanation

output:

First: Car
Last: WuWuWuWuWuW
Initial: u
Age: 0
Salary: 33

how did the program terminate? (normally, segfault, something else?)

normally

was output of the program as expected? if not, try to explain why

No, the **Initial:** , **Age:** , **Salary:** , did not have expected outputs. Also, since the last name field should only be a string of 4 characters plus '\0', the output for **Last:** with 11 characters was not as expected either. The reason for this is because since I inputted 11 characters for the last name, some fields in the person struct was overwritten. The char m field (at m+10), as well as the 5 bytes of padding (at m+11) were overwritten as well. Part of the memory address of the pointer to record struct in the personal_record field was overwritten by a '\0' (at m+16) which comes after the 11 character last name string. So in a way the memory address the pointer to the record struct was "corrupted". The new memory address that changed from 0x602010 became 0x602000 and the data at that memory won't give us the correct age and salary values.

RUN 5: first three letters of your first name, first thirteen letters of your last name

table 1

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0xf0	0xe0	0xff	0xff	0xff	0x7f	0x00	0x00
char representation	-	-	-	-	-	-	'\0'	'\0'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
char representation	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0xe0	0x07	0x40	0x00	0x00	0x00	0x00	0x00
char representation	-	-	'@'	'\0'	'\0'	'\0'	'\0'	'\0'

table 2

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0xf0	0xe0	0xff	0xff	0xff	0x7f	0x00	0x00
char representation	-	-	-	-	-	-	'\0'	'\0'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x00	0x00	0x4c	0x00	0x00	0x00	0x00	0x00
char representation	'\0'	'\0'	'L'	'\0'	'\0'	'\0'	'\0'	'\0'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0x10	0x20	0x60	0x00	0x00	0x00	0x00	0x00
char representation	-	' '	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'

table 3

memory address	m+0	m+1	m+2	m+3	m+4	m+5	m+6	m+7
hex representation	0x43	0x61	0x72	0x00	0xff	0x57	0x75	0x57
char representation	'C'	'a'	'r'	'\0'	-	'W'	'u'	'W'
memory address	m+8	m+9	m+10	m+11	m+12	m+13	m+14	m+15
hex representation	0x75	0x57	0x75	0x57	0x75	0x57	0x75	0x57
char representation	'u'	'W'	'u'	'W'	'u'	'W'	'u'	'W'
memory address	m+16	m+17	m+18	m+19	m+20	m+21	m+22	m+23
hex representation	0x75	0x57	0x00	0x00	0x00	0x00	0x00	0x00
char representation	'u'	'W'	'\0'	'\0'	'\0'	'\0'	'\0'	'\0'

RUN 5 Explanation

output:

First: Car
Last: WuWuWuWuWuWuW
Initial: u
Segmentation fault

how did the program terminate? (normally, segfault, something else?)

segfault

was output of the program as expected? if not, try to explain why

No, there were errors with the output. There were problems similar to RUN 4 but this time, there was a segmentation fault and no output at all for Age and Salary which is part of the pointer to the record struct. The memory address of the pointer is overwritten and altered so that when trying to access this new memory address stored at the pointer to record, we get a segmentation fault for accessing memory that is not initialized. For this process, the program is trying to access a memory address that does not exist for the program.