- CPSC 275: Introduction to Computer Systems

CPSC 275: Introduction to Computer Systems

Fall 2025

- Syllabus
- Schedule
- Resources
- Upload
- Solution

Homework 23

> NOTE: You are not required to hand in the following exercises, but you are strongly encouraged to complete them to strengthen your understanding of the concepts covered in class.

The following code shows an implementation of a function that reads a line from standard input, copies the string to newly allocated storage, and returns a pointer to the result.

C Code:

```
char *getline()
{
   char buf[8];
   char *result;
   gets(buf);
   result = malloc(strlen(buf)); // allocates 8 bytes in the heap
   strcpy(result, buf);
   return result;
}
```

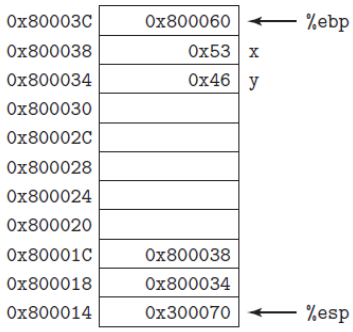
Disassembly up through call to `gets`:

```
1  080485c0 :
2  80485c0: 55                 push %ebp
3  80485c1: 89 e5              mov %esp,%ebp
4  80485c3: 83 ec 28           sub $0x28,%esp
5  80485c6: 89 5d f4           mov %ebx,-0xc(%ebp)
6  80485c9: 89 75 f8           mov %esi,-0x8(%ebp)
7  80485cc: 89 7d fc           mov %edi,-0x4(%ebp)
   Diagram stack at this point
8  80485cf: 8d 75 ec           lea -0x14(%ebp),%esi
9  80485d2: 89 34 24           mov %esi,(%esp)
10 80485d5: e8 a3 ff ff ff     call 804857d
   Modify diagram to show stack contents at this point
```

Consider the following scenario. Procedure `getline` is called with the return address equal to `0x8048643`, register `%ebp` equal to `0xbffffc94`, register `%ebx` equal to `0x1`, register `%esi` is equal to `0x2`, and register `%edi` is equal to `0x3`. You type in the string `"0123456789012345678901 23"`. The program terminates with a segmentation fault.

A. Fill in the diagram that follows, indicating as much as you can about the stack just after executing the instruction at Line 7 in the disassembly. Label the quantities stored on the stack (e.g., “Return address”) on

the right, and their hexadecimal values (if known) within the box. Each box represents 4 bytes. Indicate the position of `%ebp`.



B. Modify your diagram to show the effect of the call to `gets` (Line 10).
C. To what address does the program attempt to return?
D. What register(s) have corrupted value(s) when `getline` returns?

- **Welcome: Sean**
  - LogOut

Trinity College
HARTFORD CONNECTICUT