

The following problem concerns the following, low-quality code:

```
void foo(int x)
{
    int a[3];
    char buf[4];
    a[0] = 0xF0F1F2F3;
    a[1] = x;
    gets(buf);
    printf("a[0] = 0x%x, a[1] = 0x%x, buf = %s\n", a[0], a[1], buf);
}
```

In a program containing this code, procedure `foo` has the following disassembled form on an IA32 machine:

```
080485d0 <foo>:
080485d0: 55          pushl   %ebp
080485d1: 89 e5       movl    %esp,%ebp
080485d3: 83 ec 10    subl    $0x10,%esp
080485d6: 53          pushl   %ebx
080485d7: 8b 45 08    movl    0x8(%ebp),%eax
080485da: c7 45 f4 f3 f2 movl    $0xf0f1f2f3,0xffffffff4(%ebp)
080485df: f1 f0
080485e1: 89 45 f8    movl    %eax,0xffffffff8(%ebp)
080485e4: 8d 5d f0    leal    0xffffffff0(%ebp),%ebx
080485e7: 53          pushl   %ebx
080485e8: e8 b7 fe ff ff call    80484a4 <_init+0x54> # gets
080485ed: 53          pushl   %ebx
080485ee: 8b 45 f8    movl    0xffffffff8(%ebp),%eax
080485f1: 50          pushl   %eax
080485f2: 8b 45 f4    movl    0xffffffff4(%ebp),%eax
080485f5: 50          pushl   %eax
080485f6: 68 ec 90 04 08 pushl    $0x80490ec
080485fb: e8 94 fe ff ff call    8048494 <_init+0x44> # printf
08048600: 8b 5d ec    movl    0xffffffffec(%ebp),%ebx
08048603: 89 ec       movl    %ebp,%esp
08048605: 5d          popl    %ebp
08048606: c3          ret
08048607: 90          nop
```

For the following questions, recall that:

- `gets` is a standard C library routine.
- IA32 machines are little-endian.
- C strings are null-terminated (i.e., terminated by a character with value 0x00).
- Characters '0' through '9' have ASCII codes 0x30 through 0x39.

Problem 35. (6 points):

Fill in the following table indicating where on the stack the following program values are located. Express these as decimal offsets (positive or negative) relative to register `%ebp`:

Program Value	Decimal Offset
<code>a</code>	
<code>a[2]</code>	
<code>x</code>	
<code>buf</code>	
<code>buf[3]</code>	
Saved value of register <code>%ebx</code>	