
CONNECTION
52 WEST
A FILM BY CHRISTOPHER NOLAN
INCEPTION
FROM THE DIRECTOR OF THE DARK KNIGHT

Recursion

Adalbert **Gerald** Soosai Raj

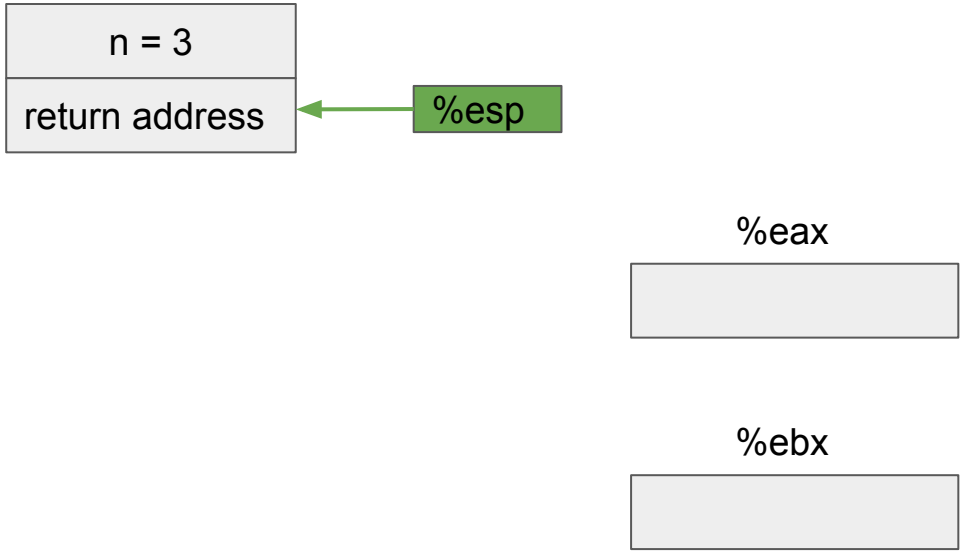
Recursive Factorial in C

```
int rfact(int n)
{
    int result;

    if (n <= 1)
        result = 1;
    else
        result = n * rfact(n-1);

    return result;
}
```

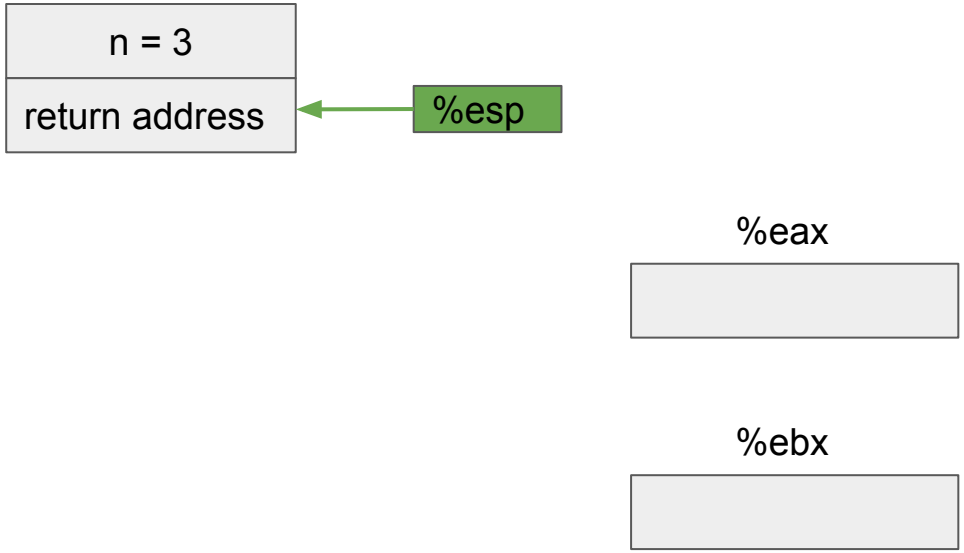
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```

Recursive Factorial in Assembly!

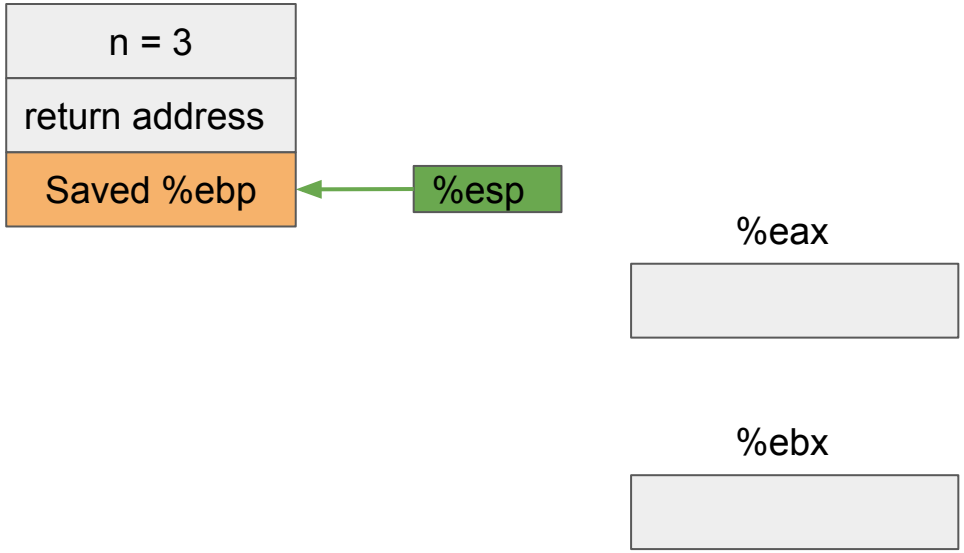
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



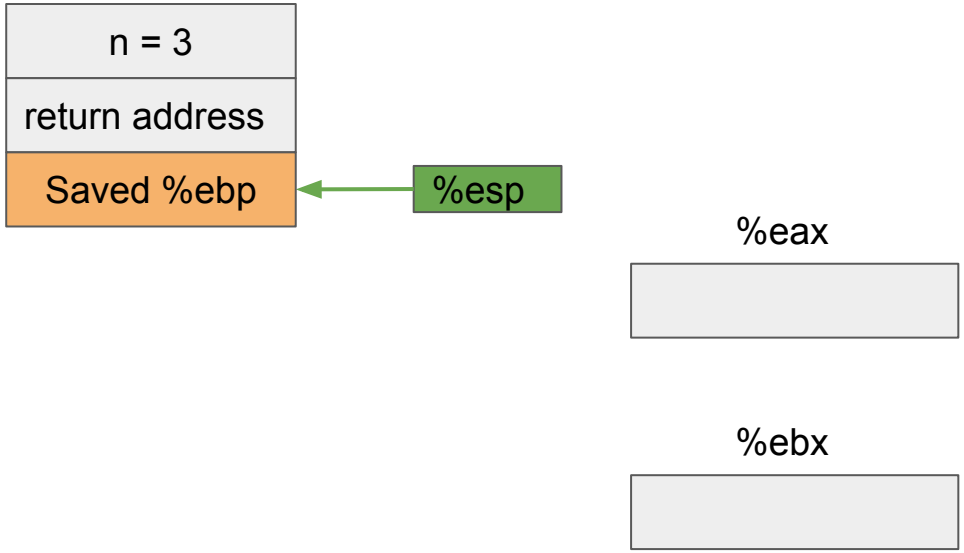
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



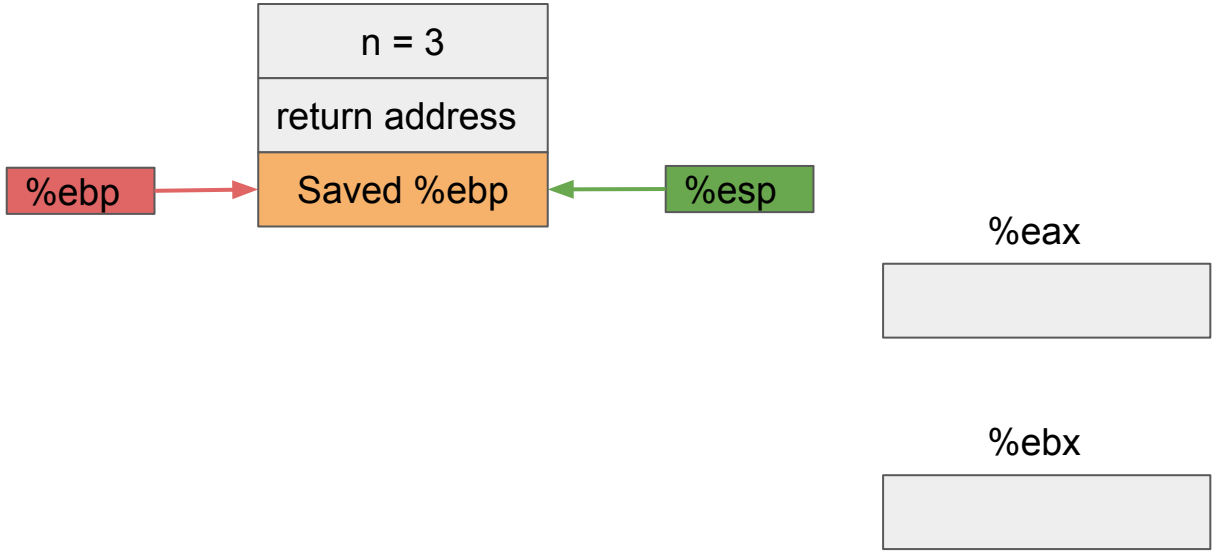
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



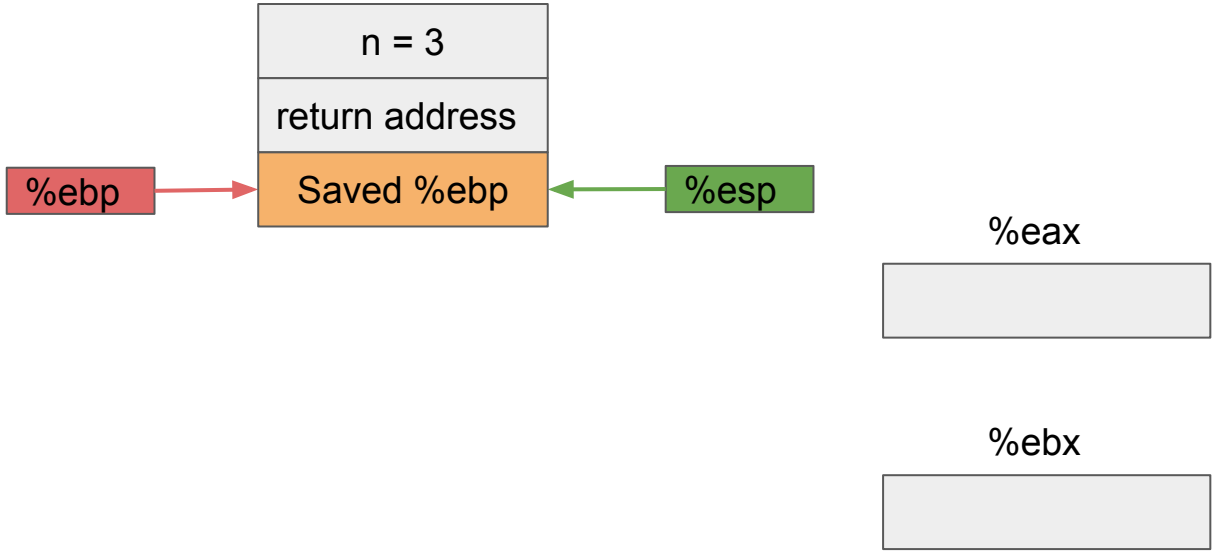
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



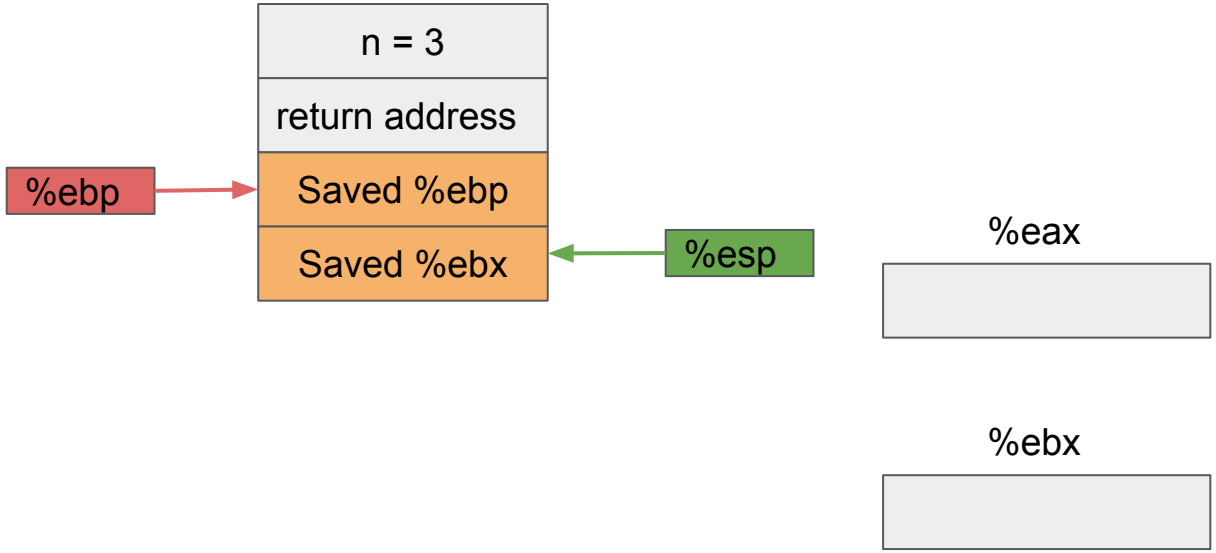
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



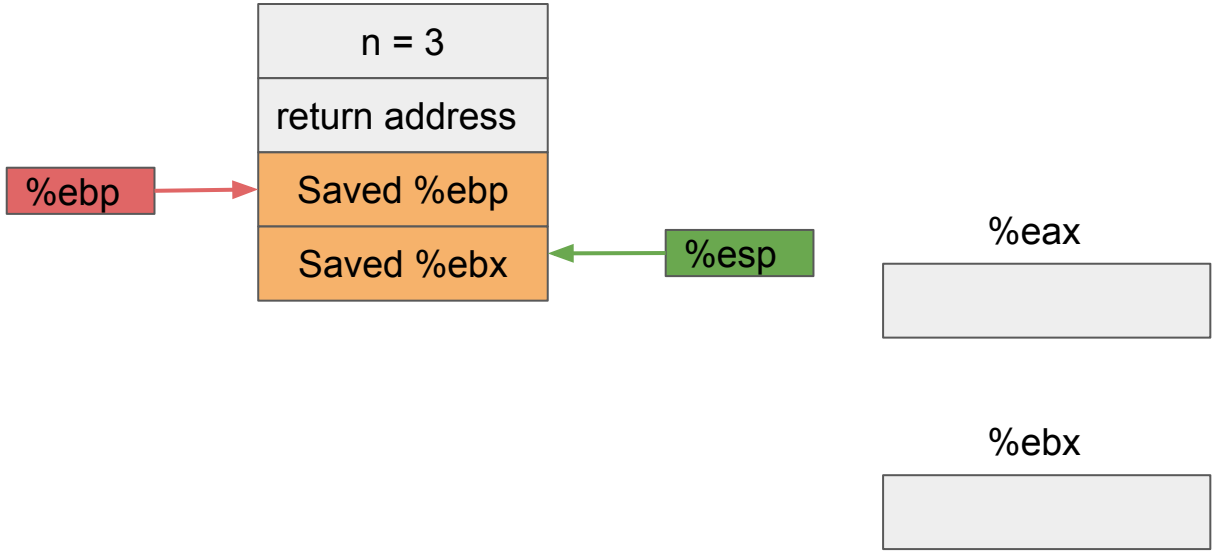
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



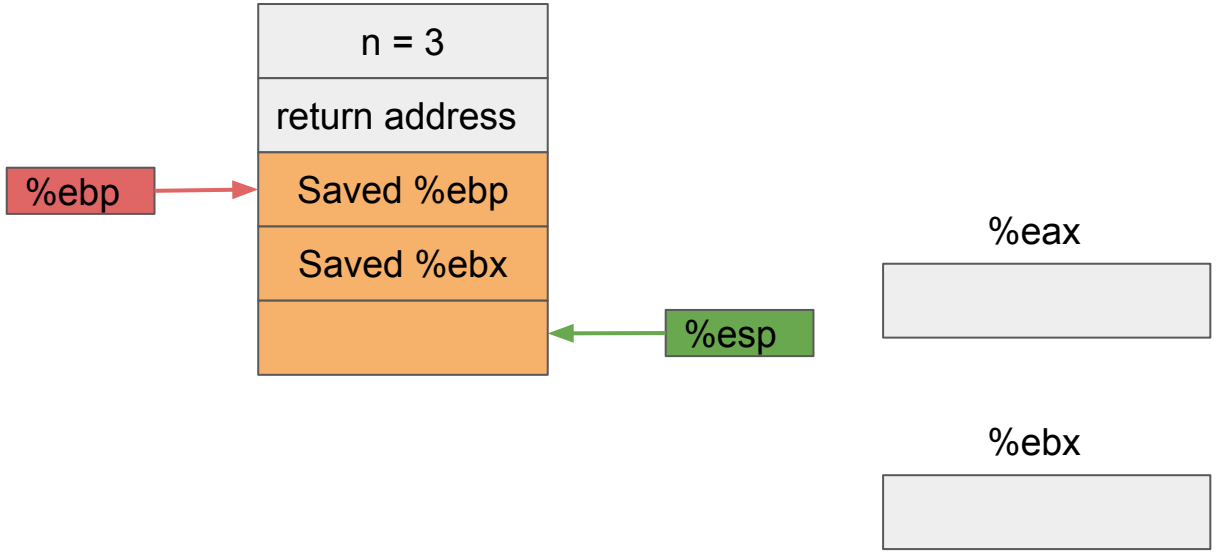
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



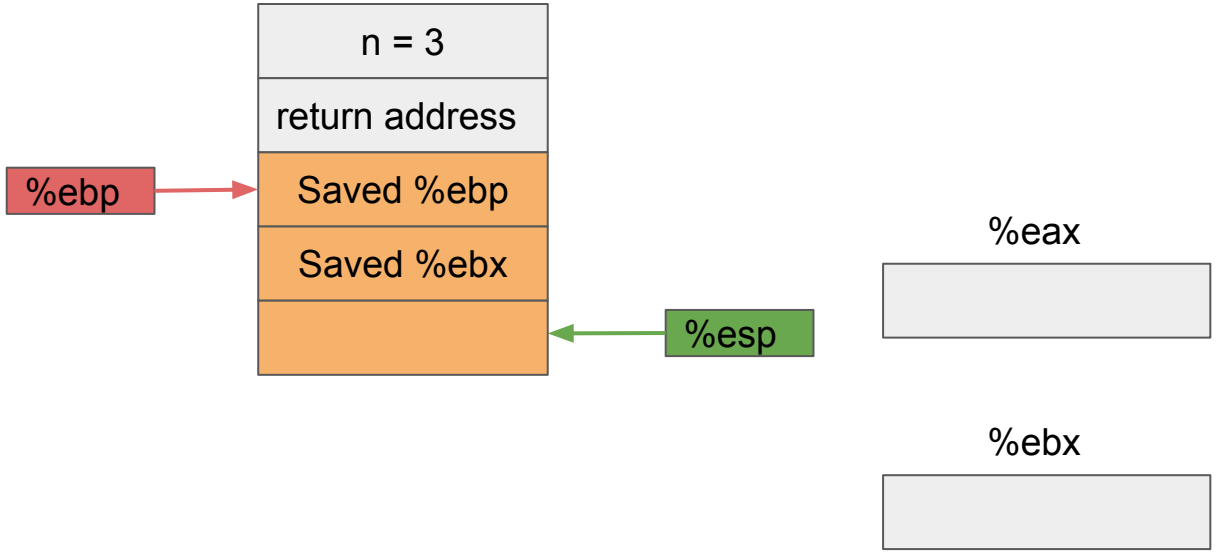
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



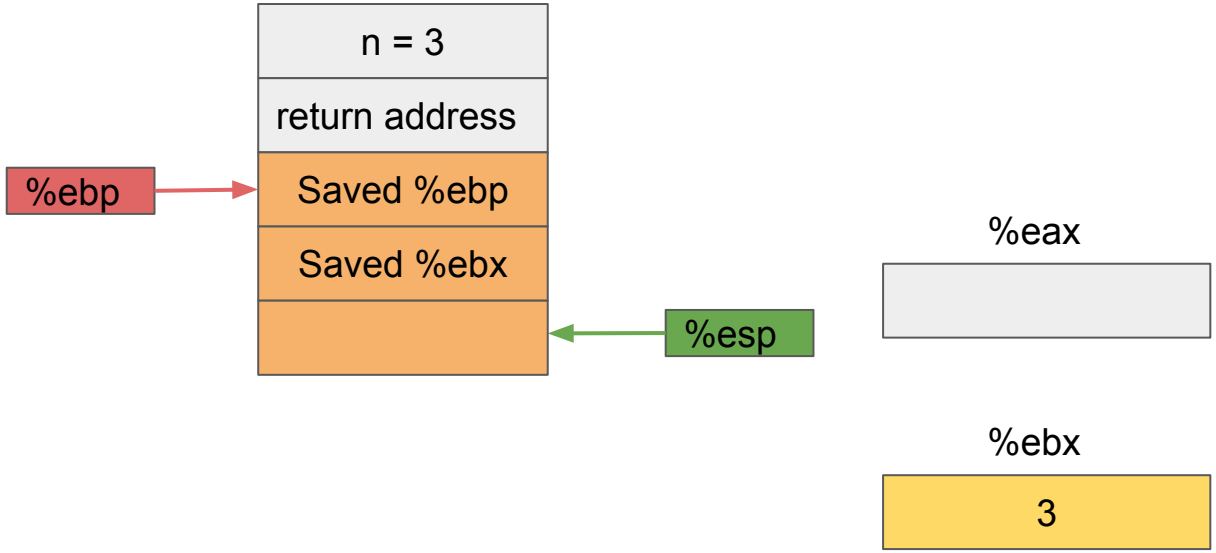
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



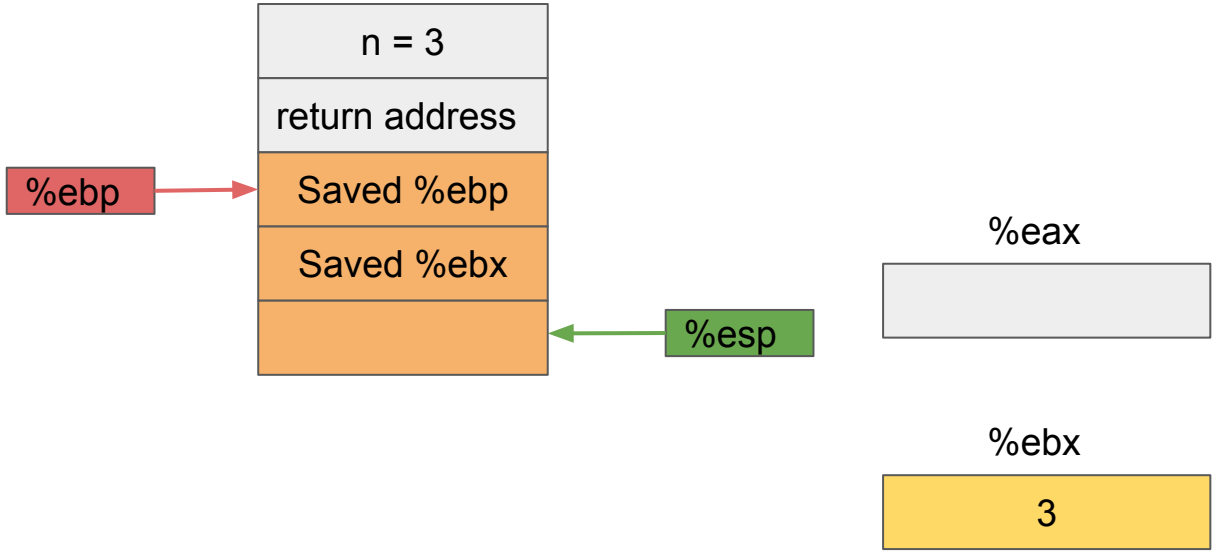
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



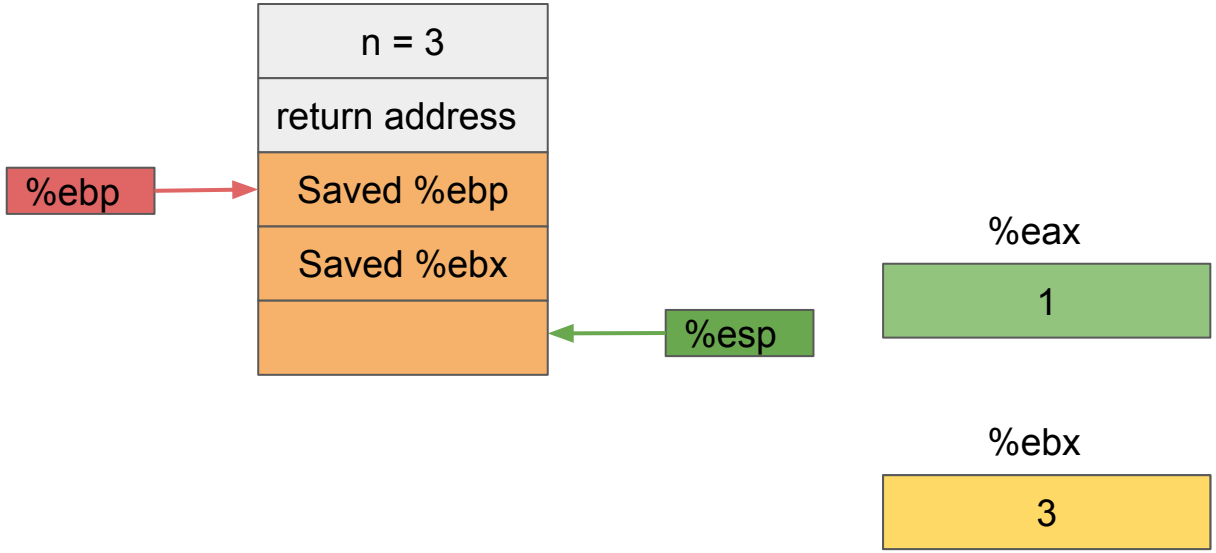
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



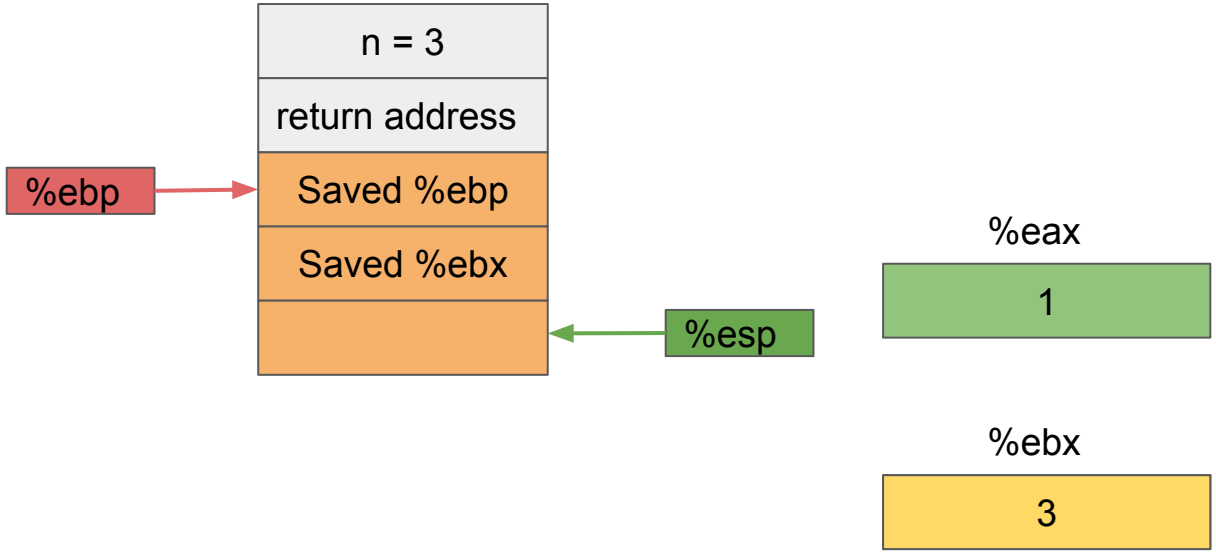
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



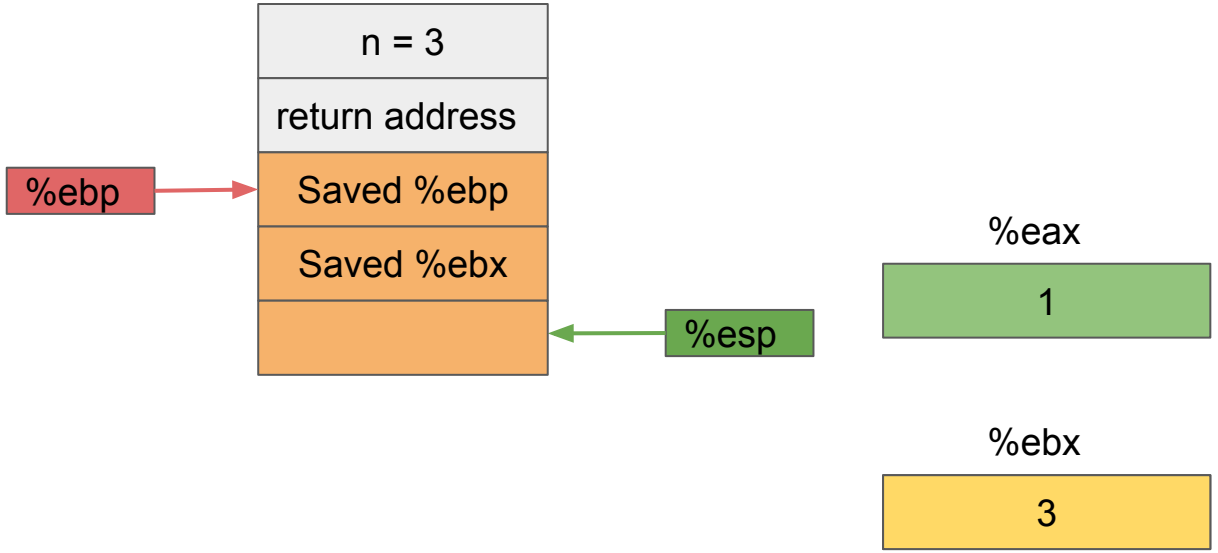
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```

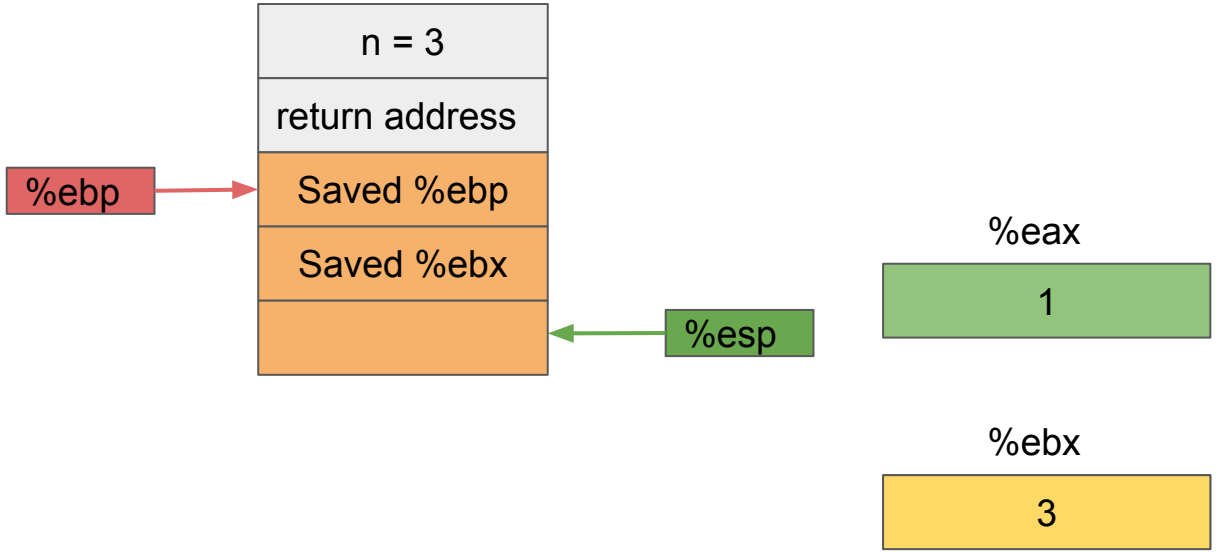


```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```

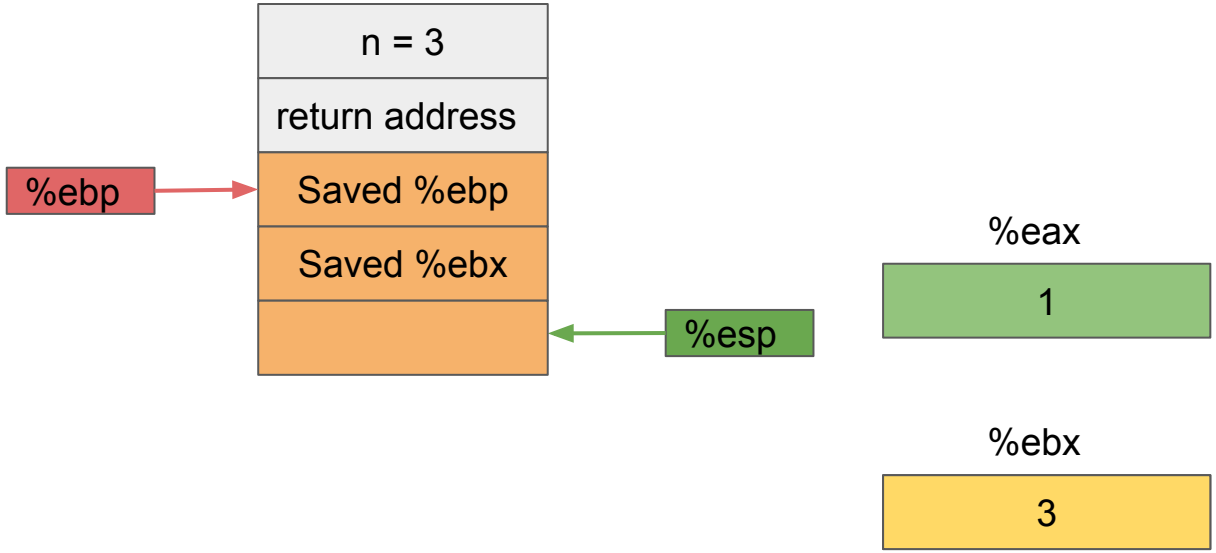


Compare 3 and 1 based on the condition 3 - 1 and update the Control Code Registers!

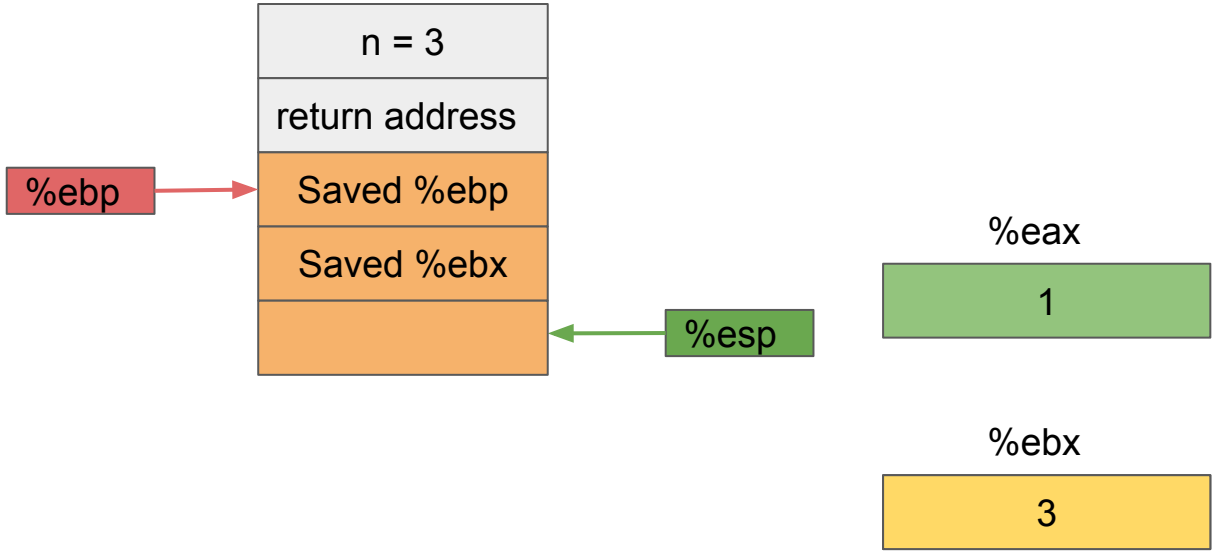
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```

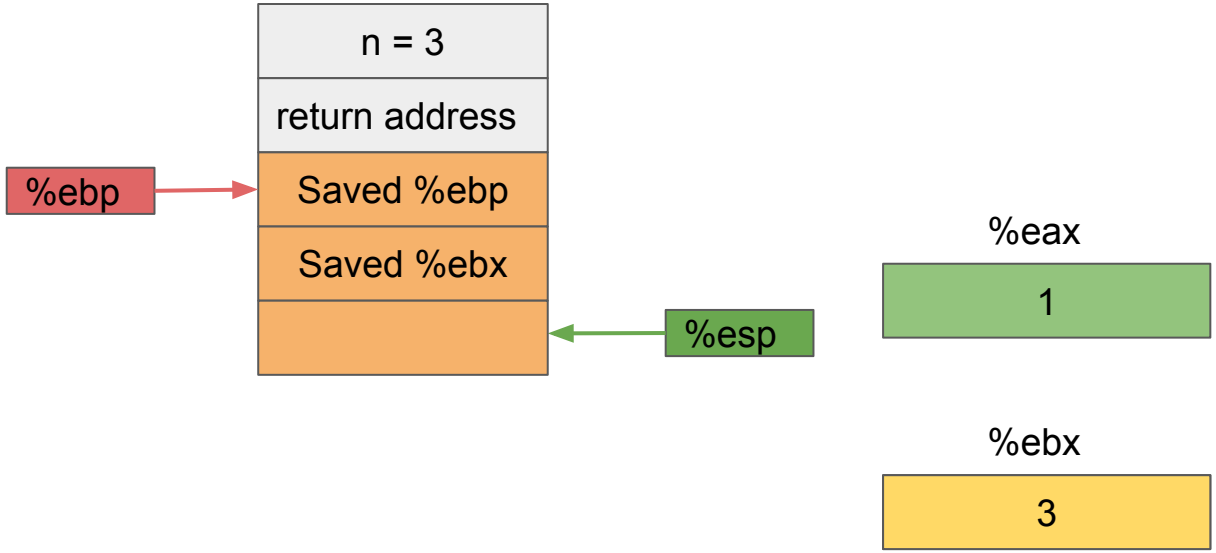


```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```

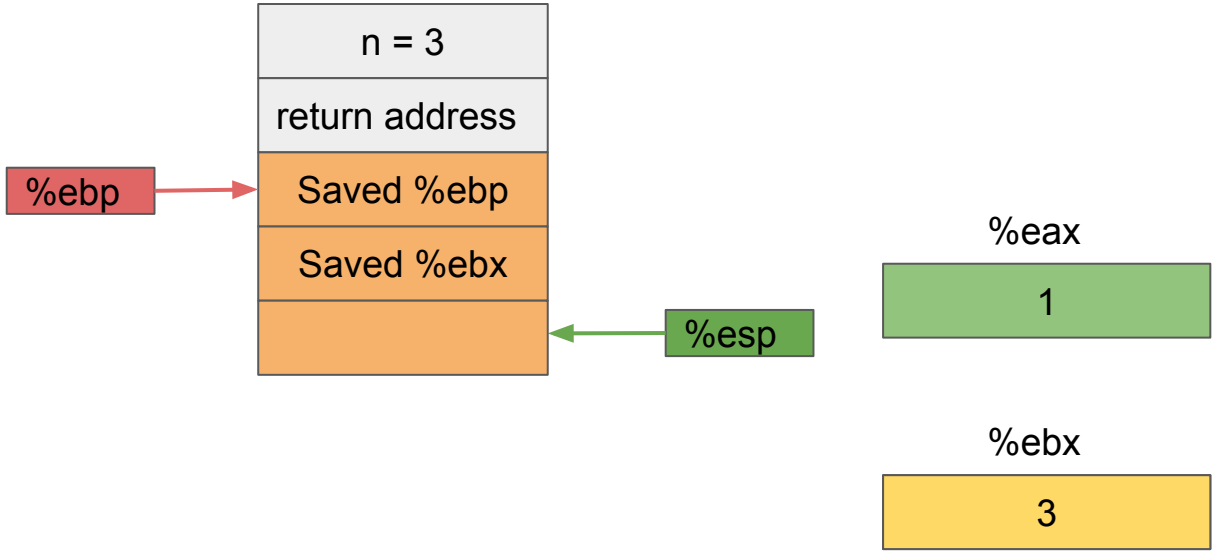


Jump if 3 <= 1

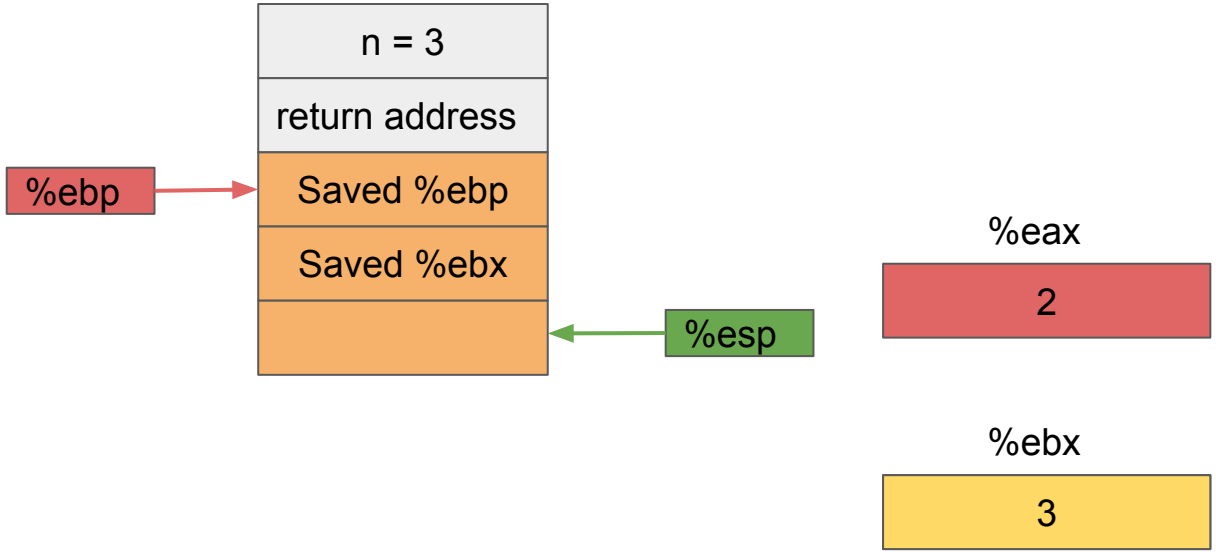
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



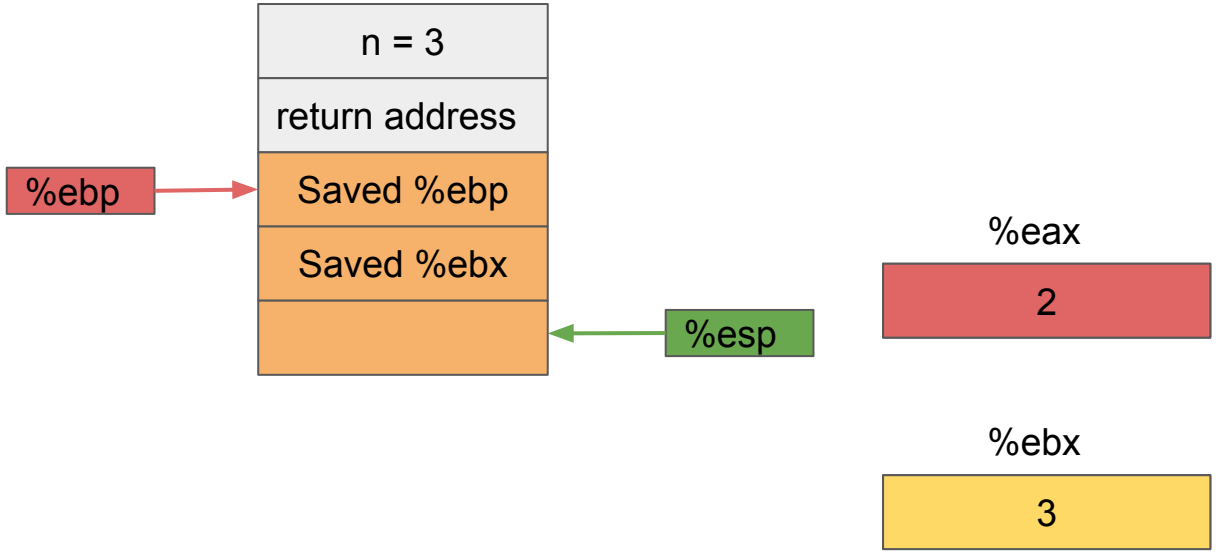
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



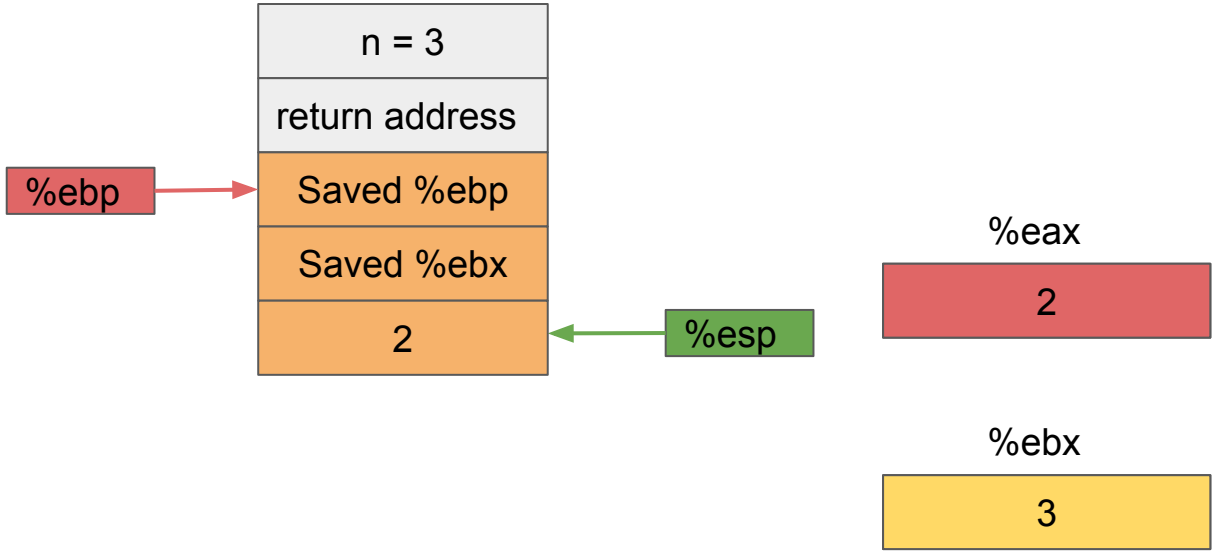
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



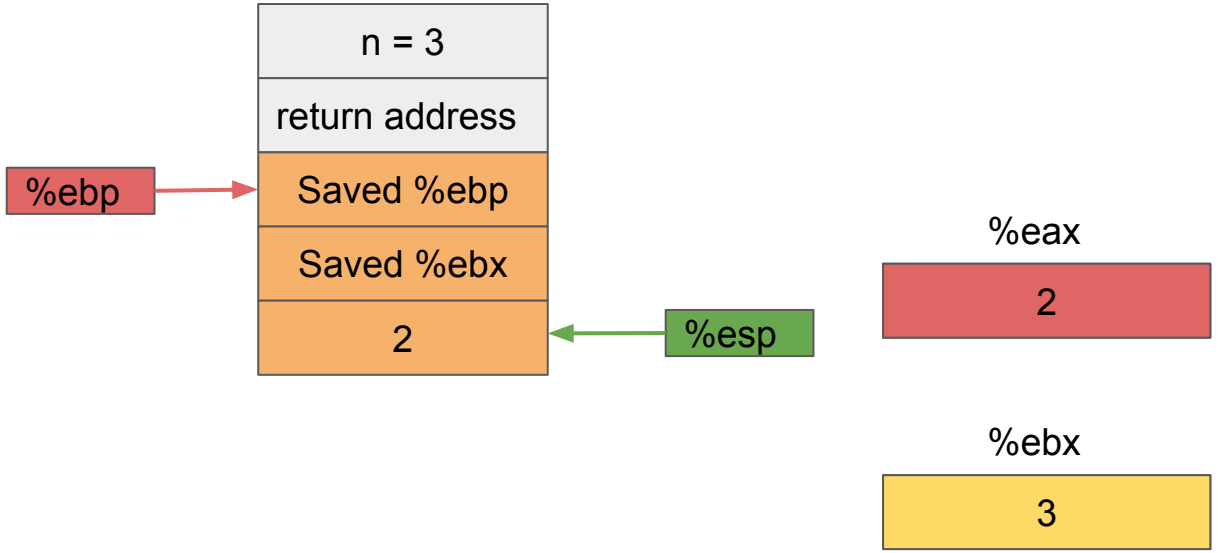
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



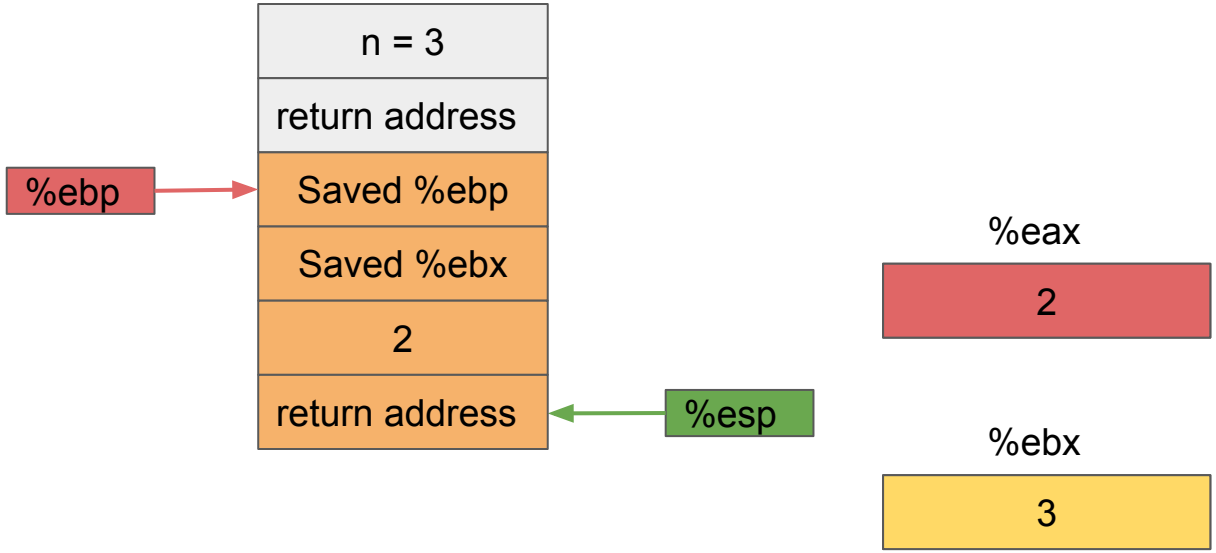
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



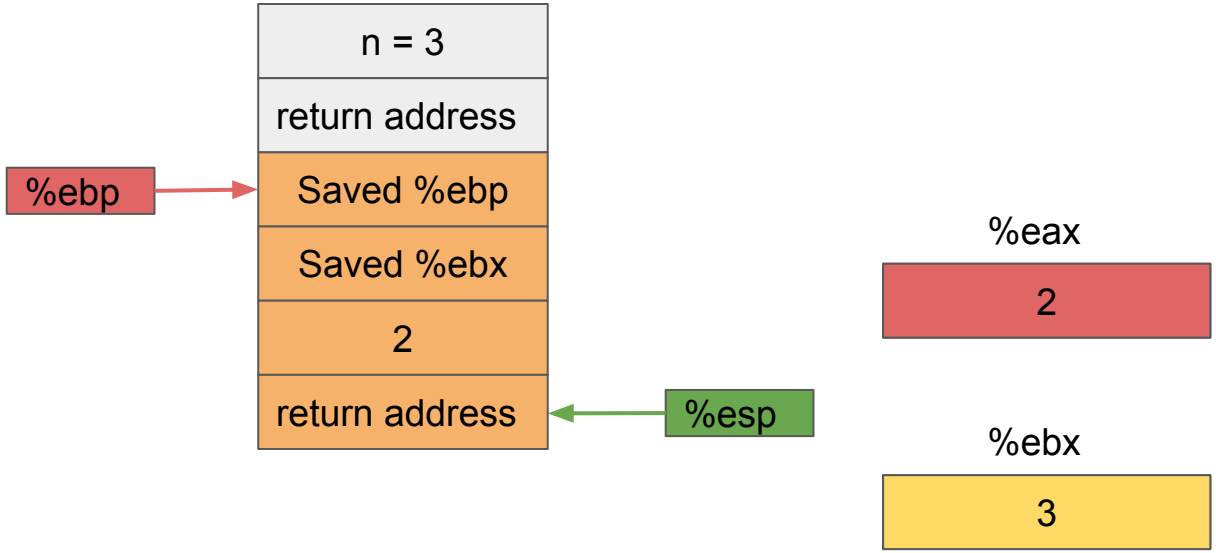
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



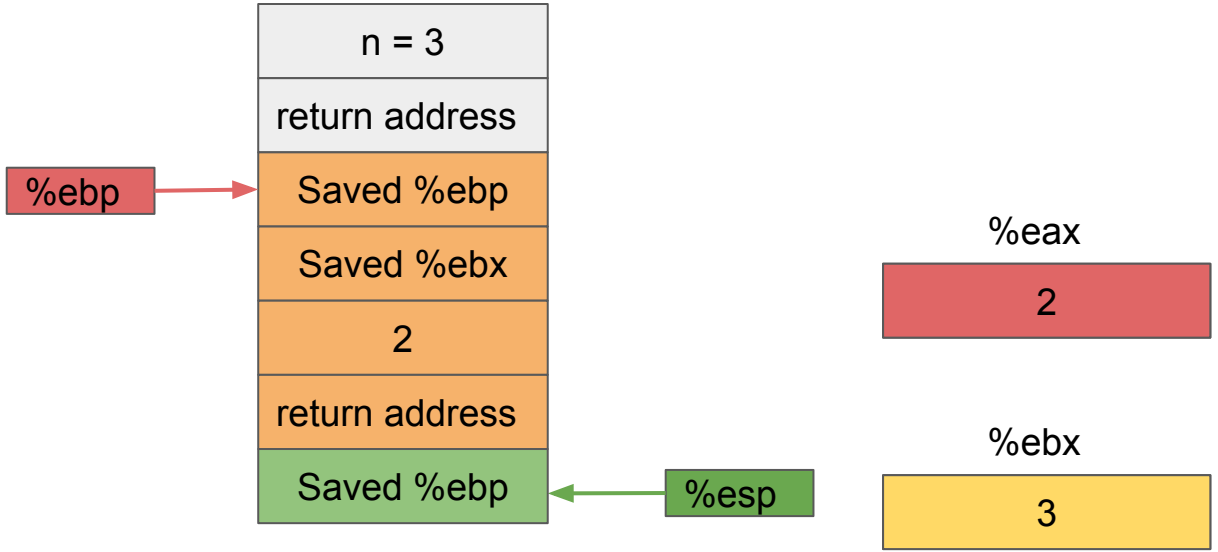
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



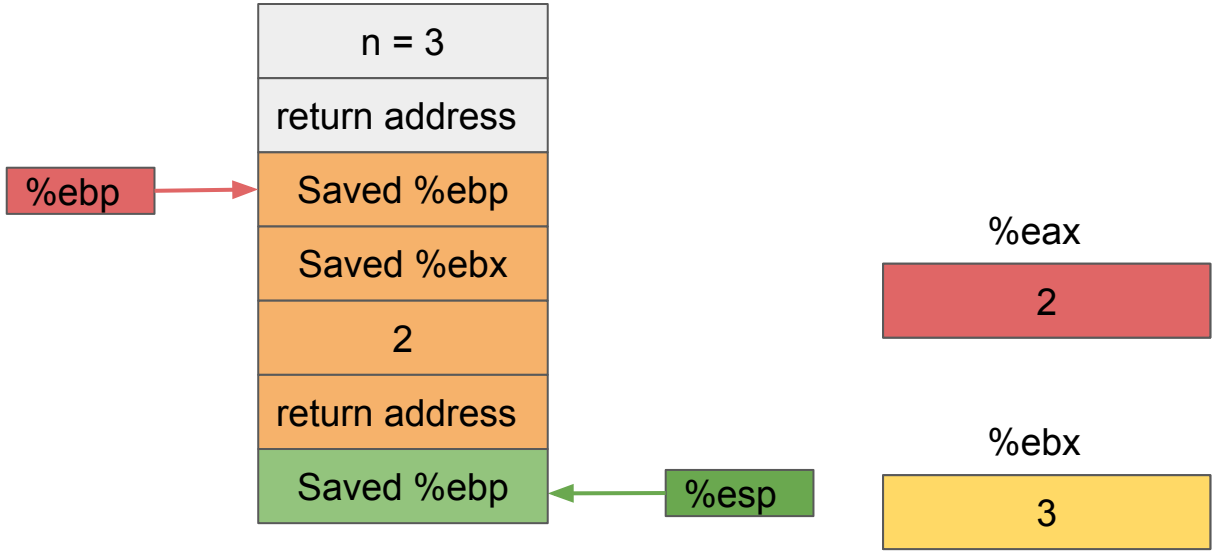
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



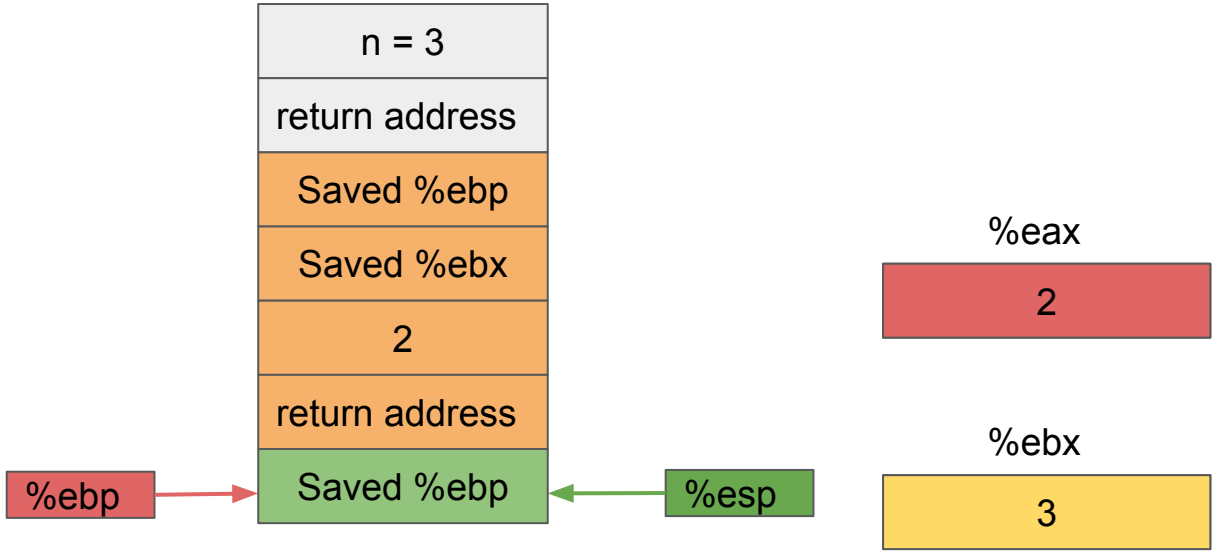
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



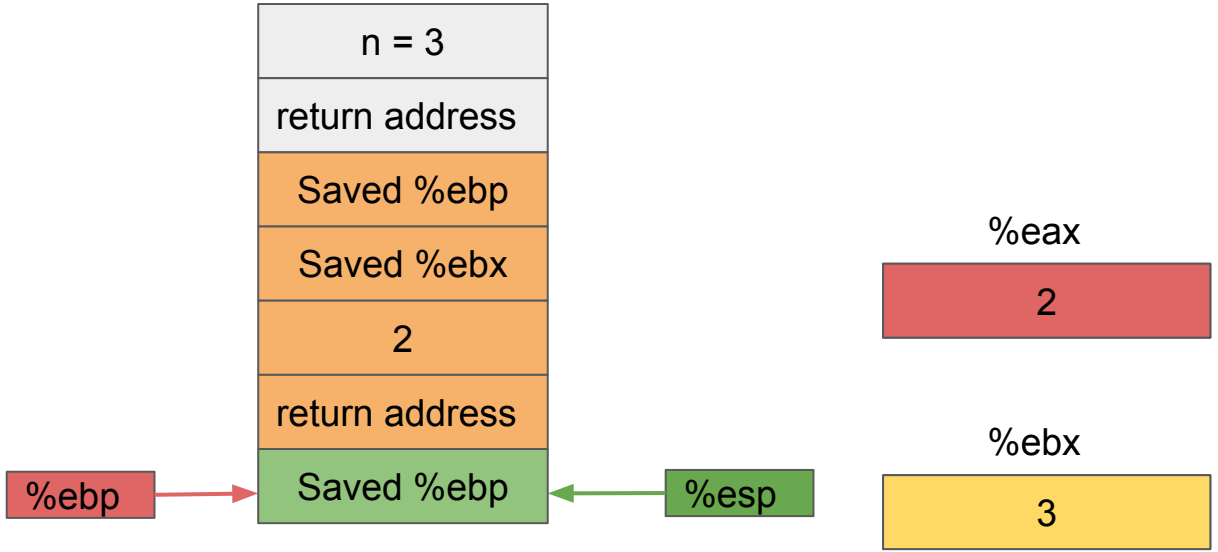
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



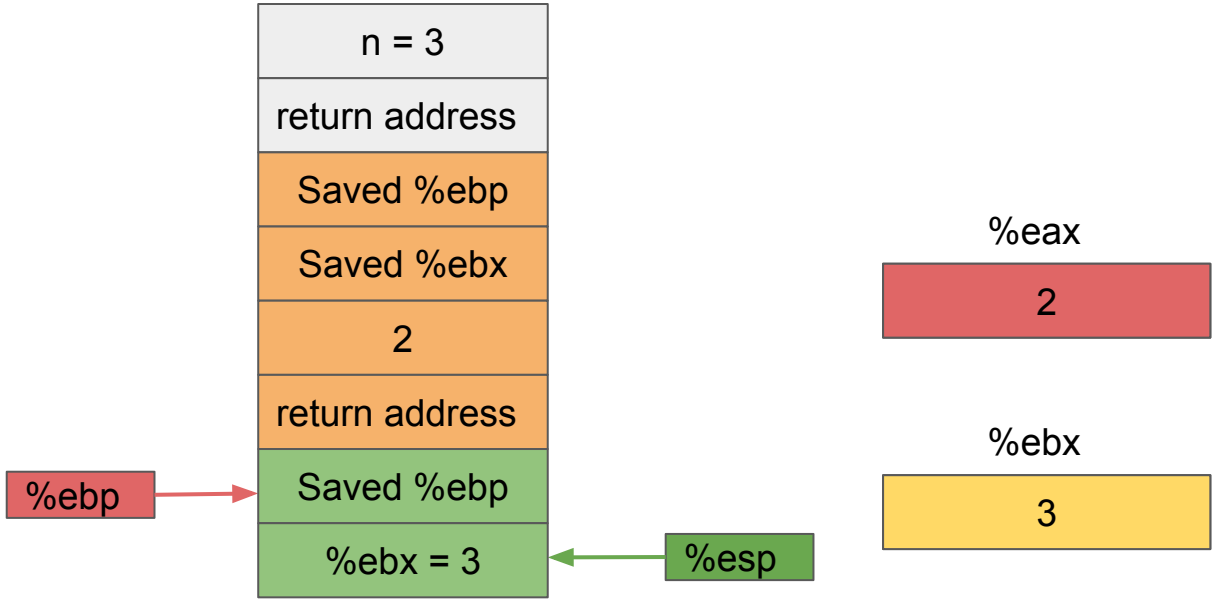
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



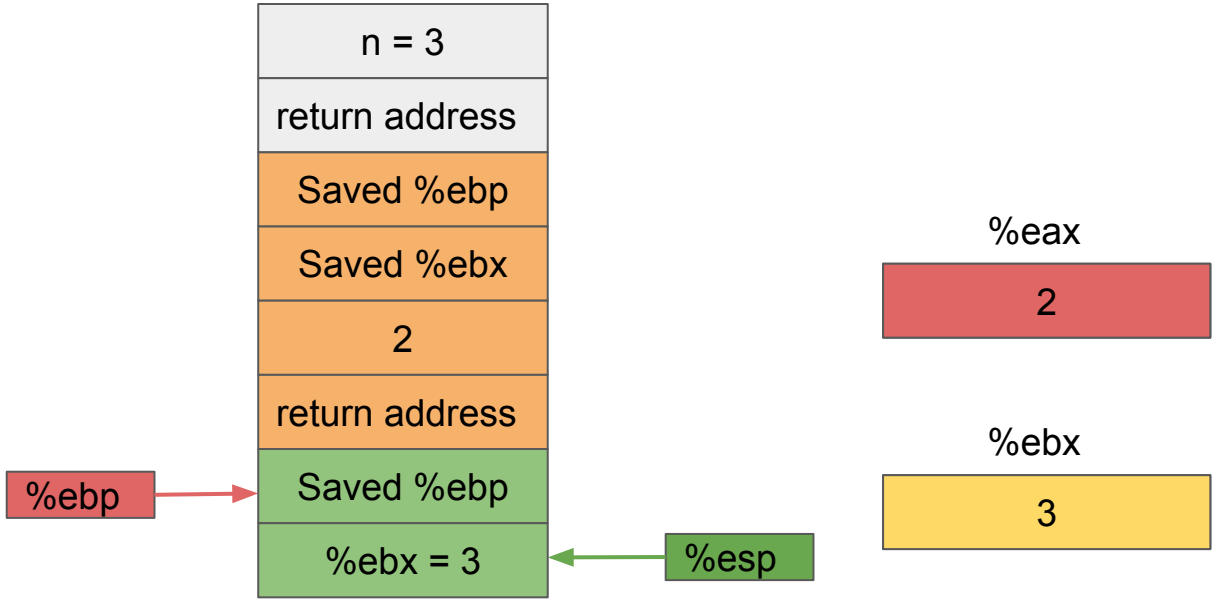
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



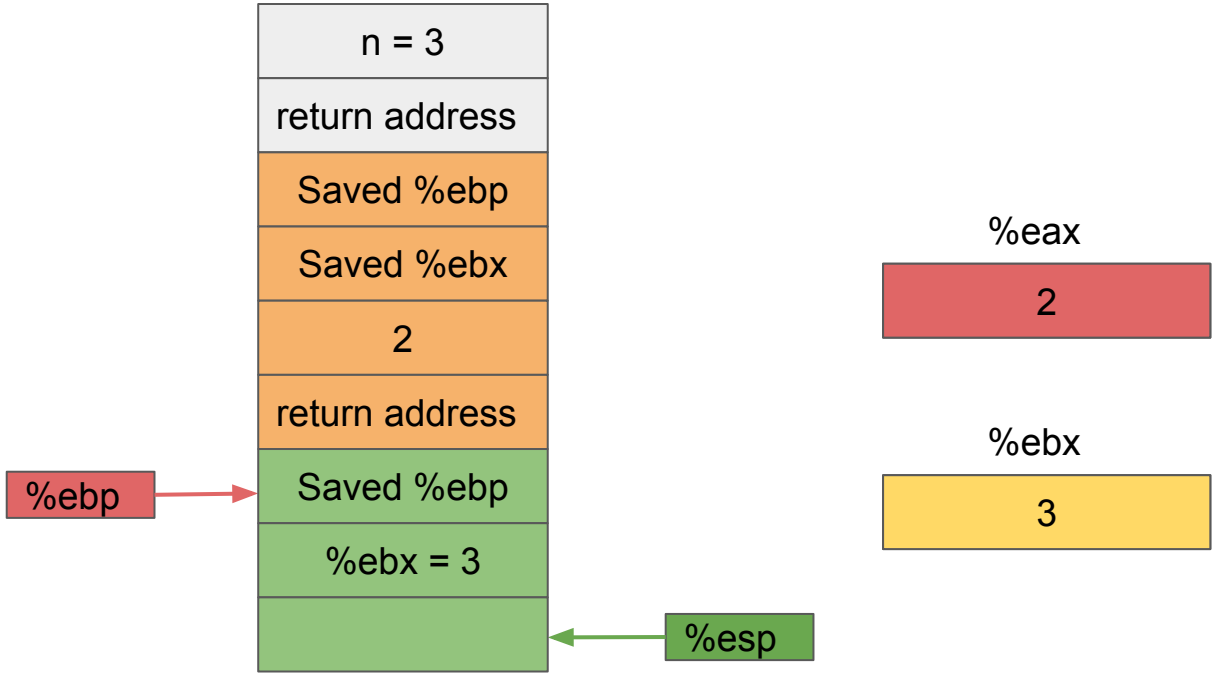
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



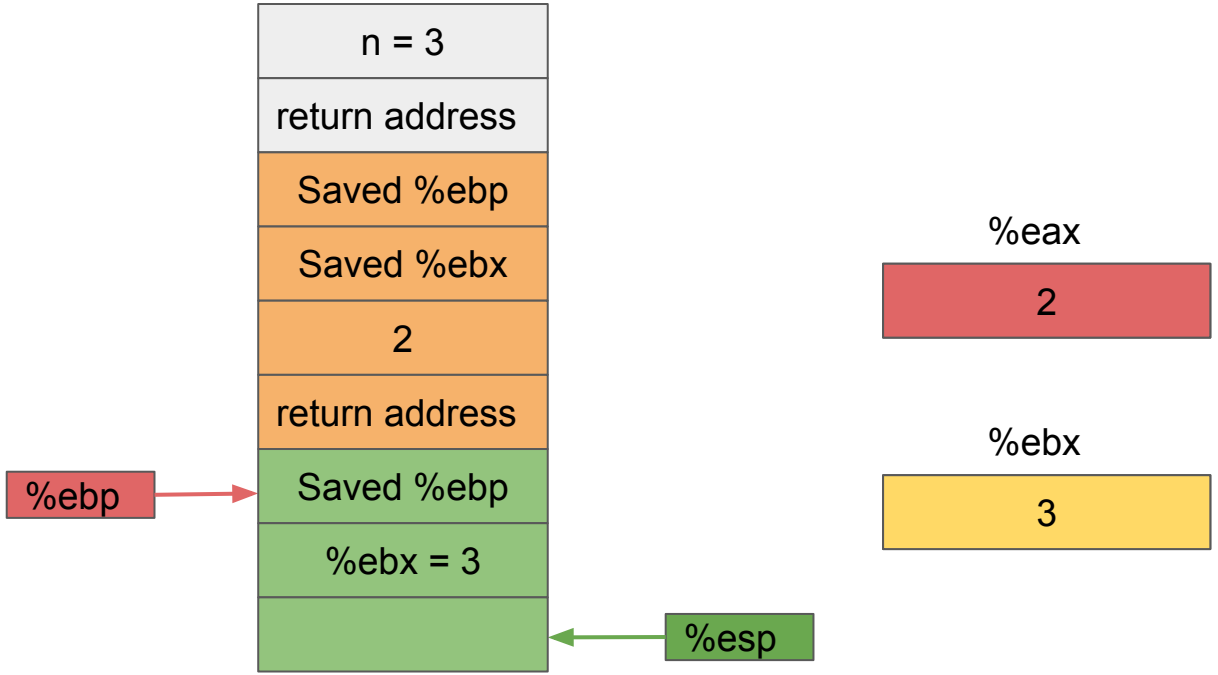
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



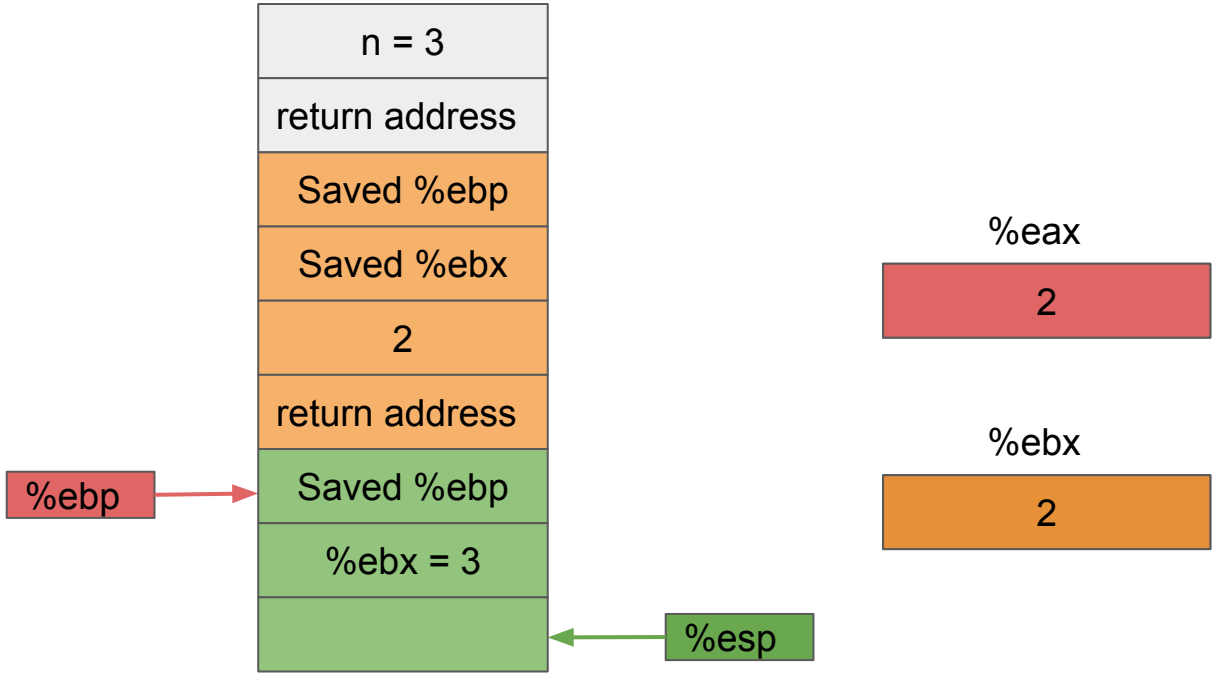
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



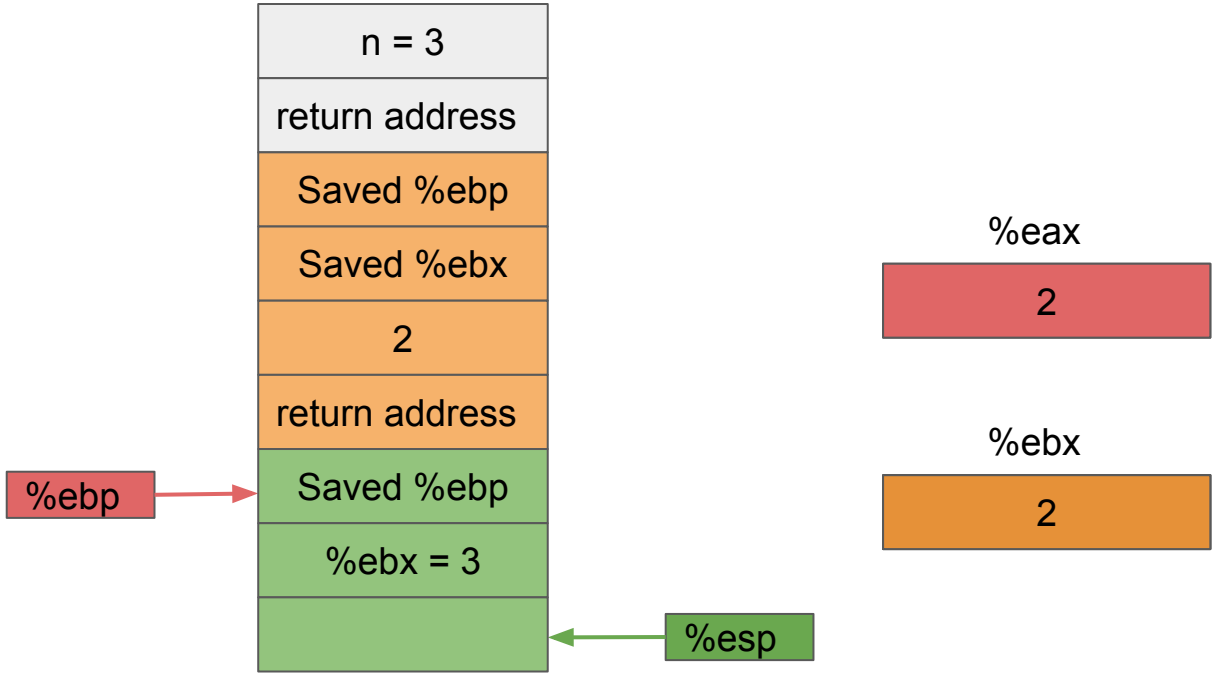
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



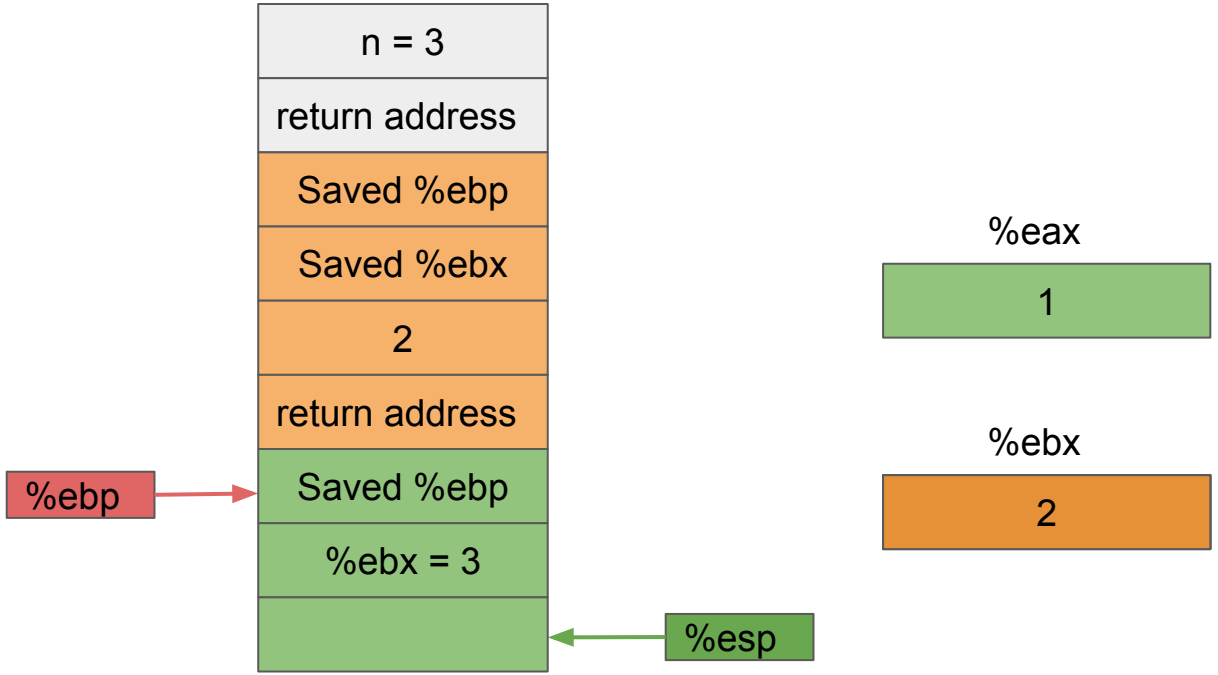
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



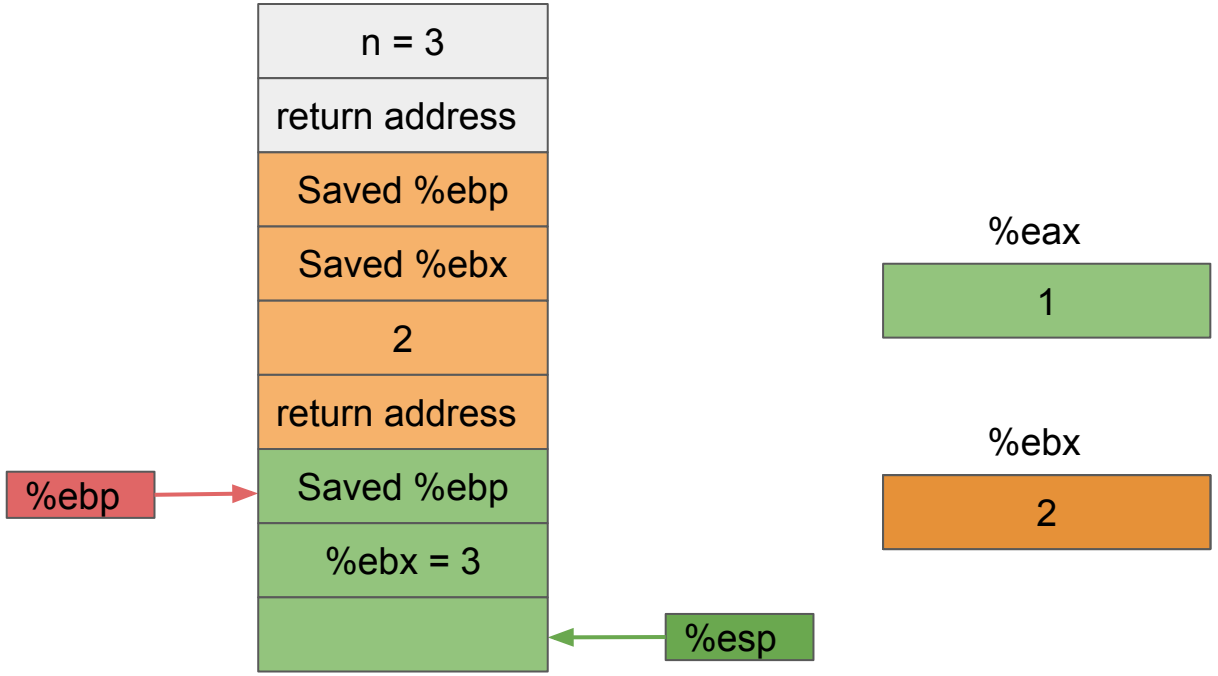
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



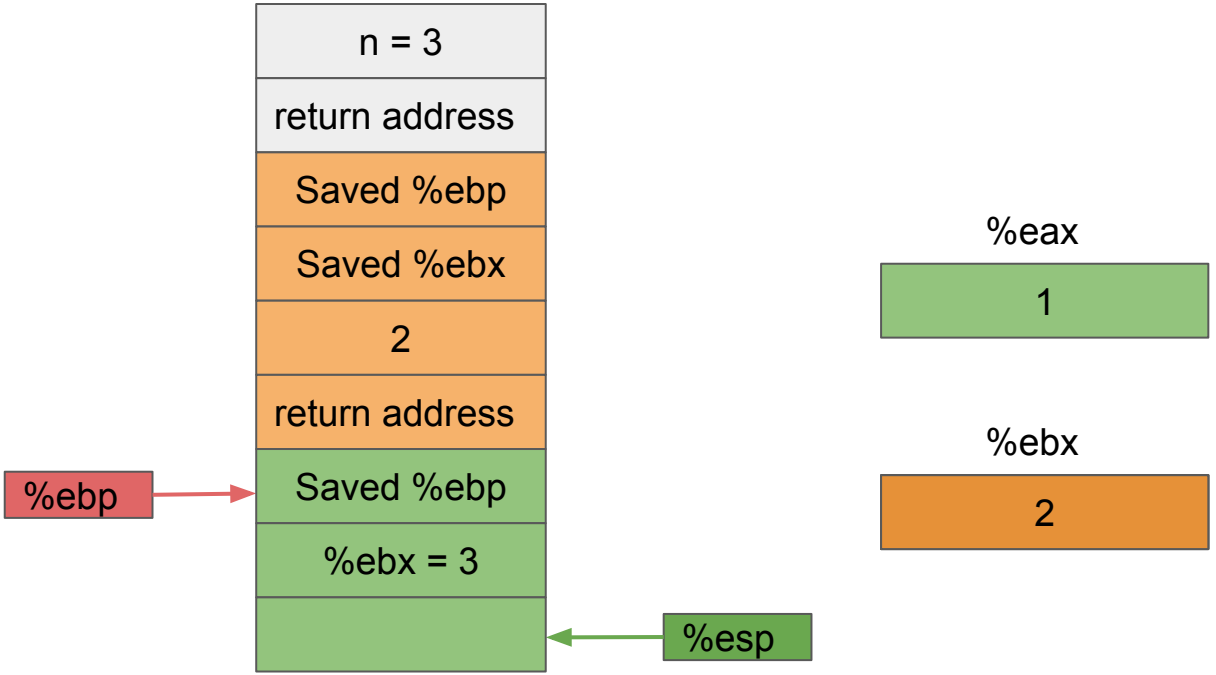
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```

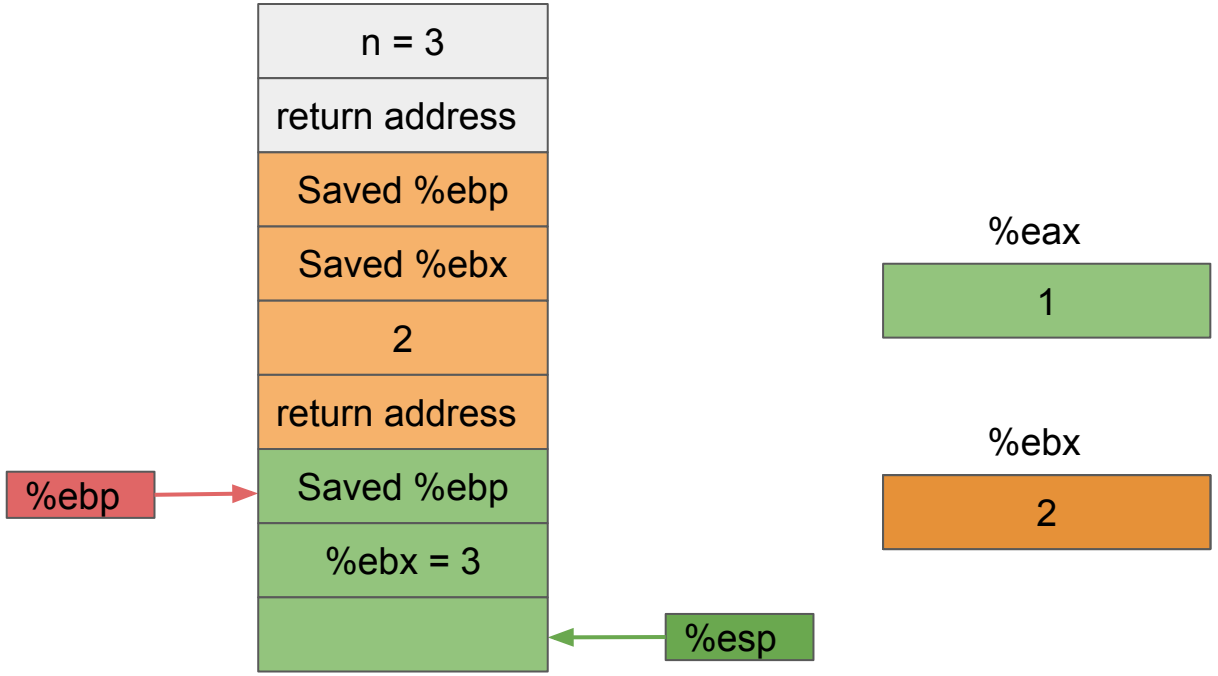


```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```

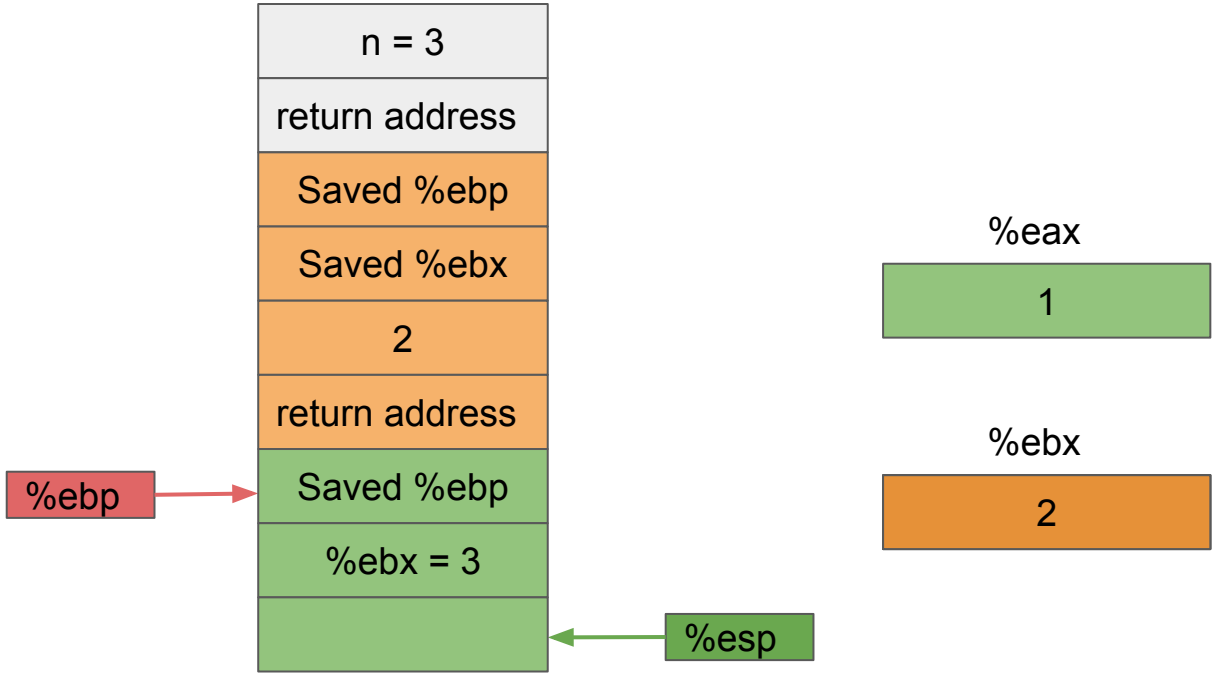


Compare 2 and 1 based on the condition 2 - 1 and update the Control Code Registers!

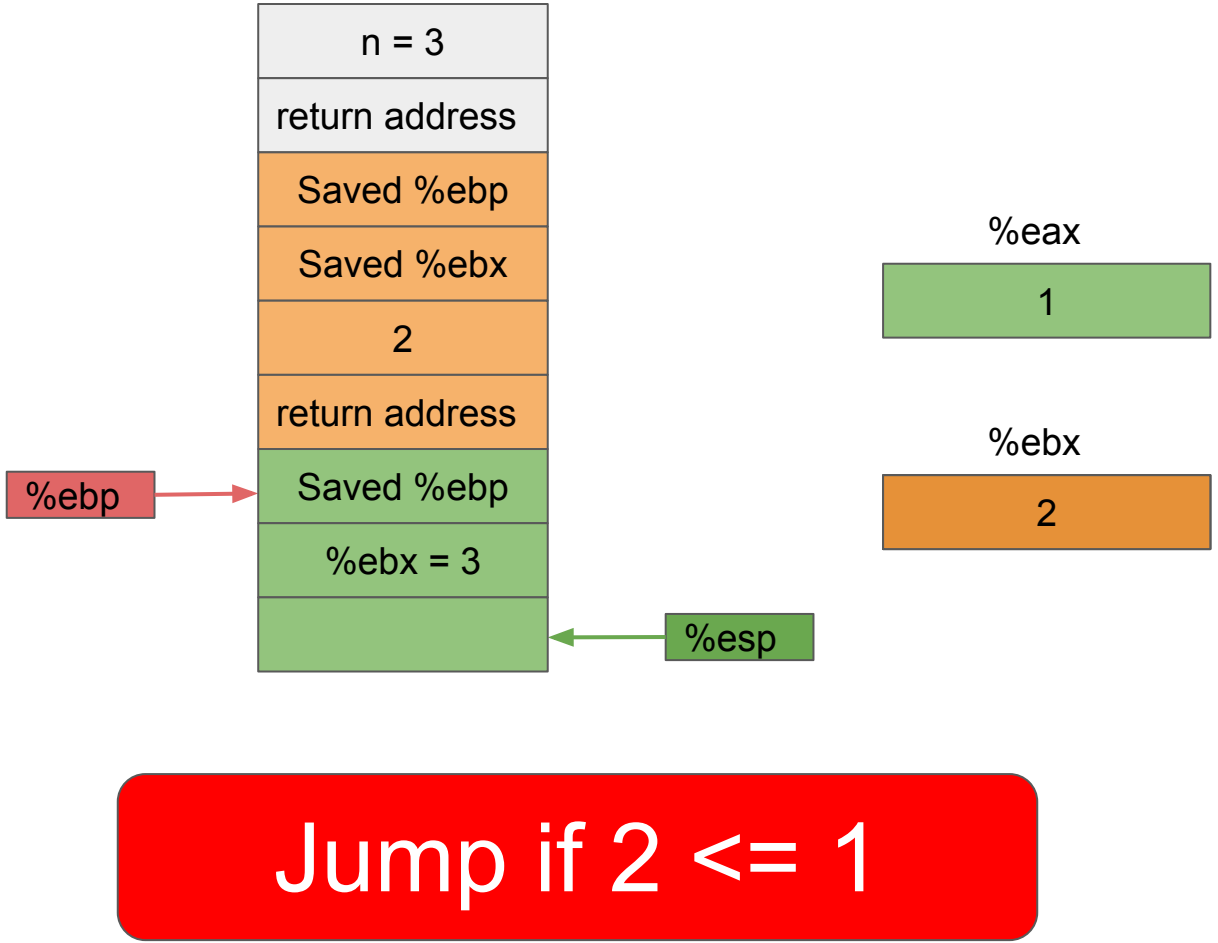
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



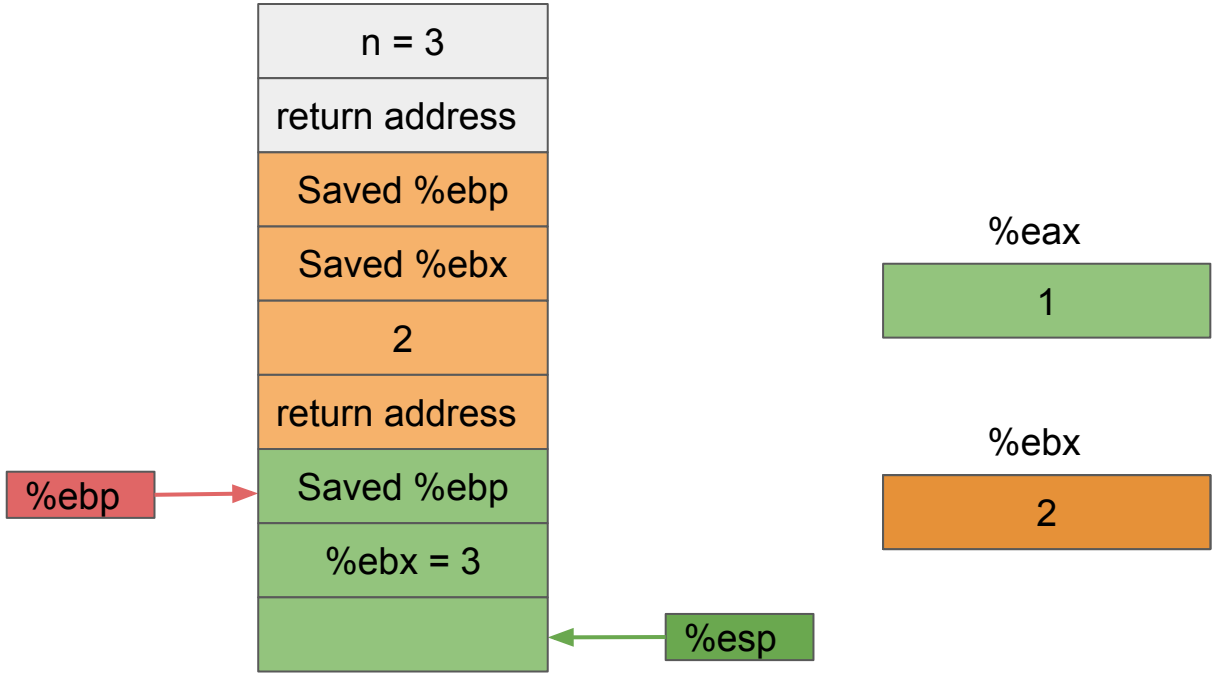
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



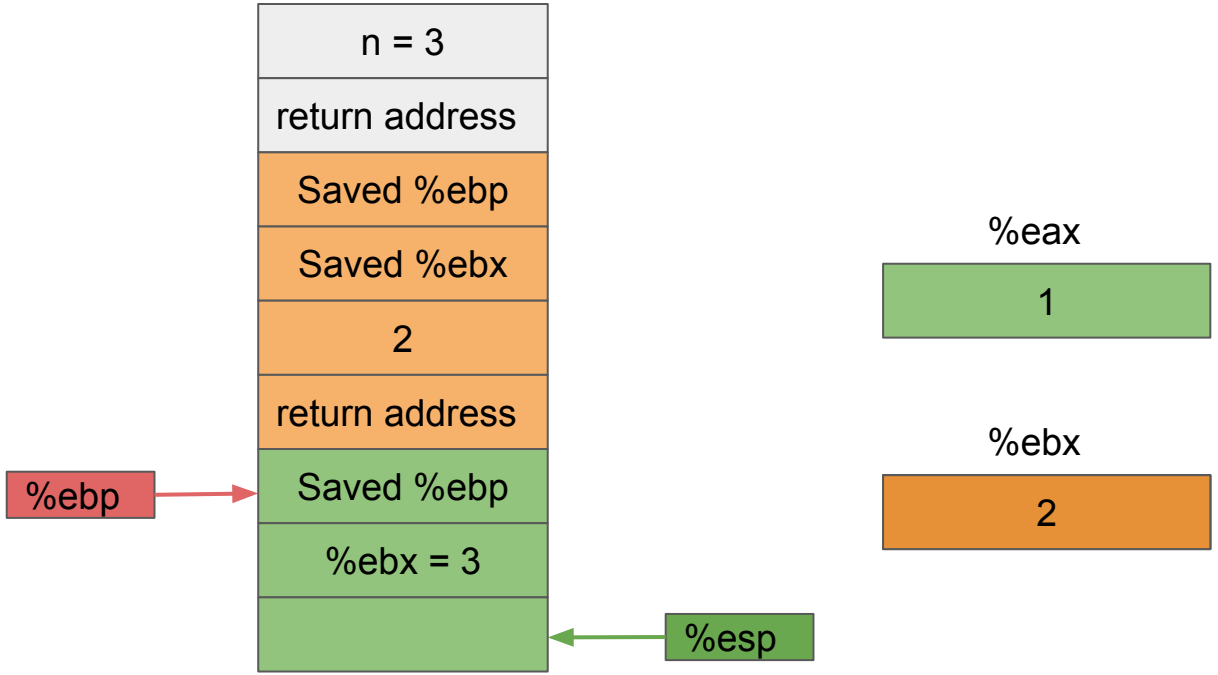
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



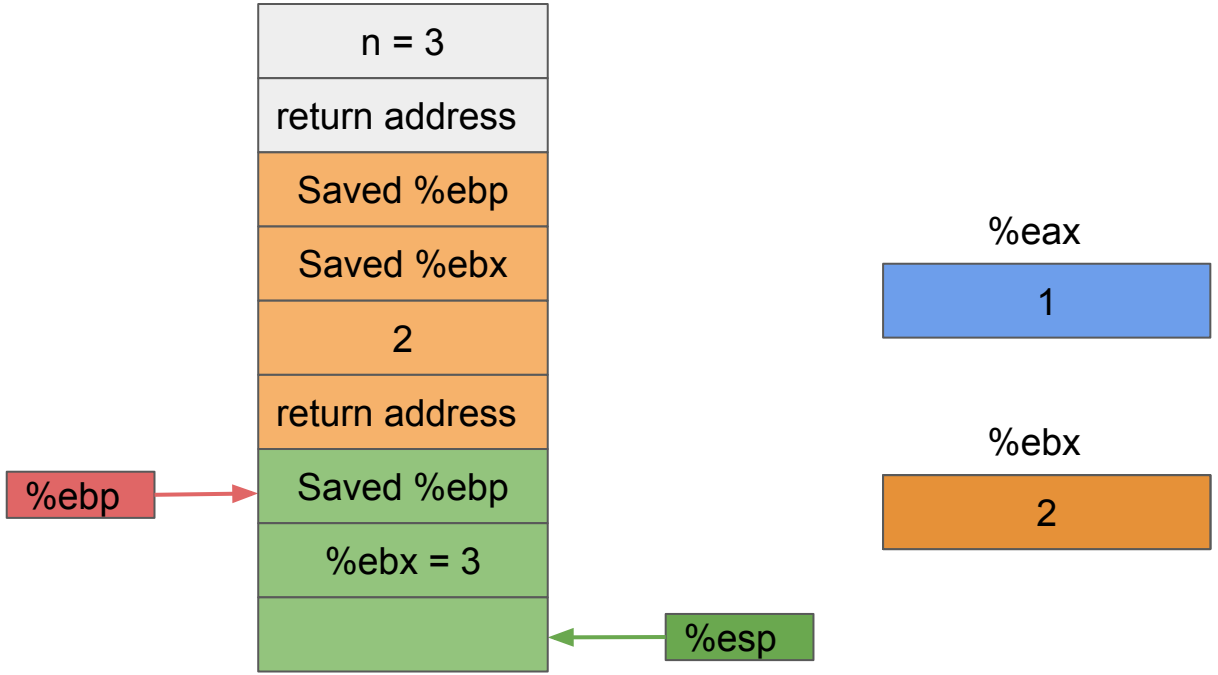
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



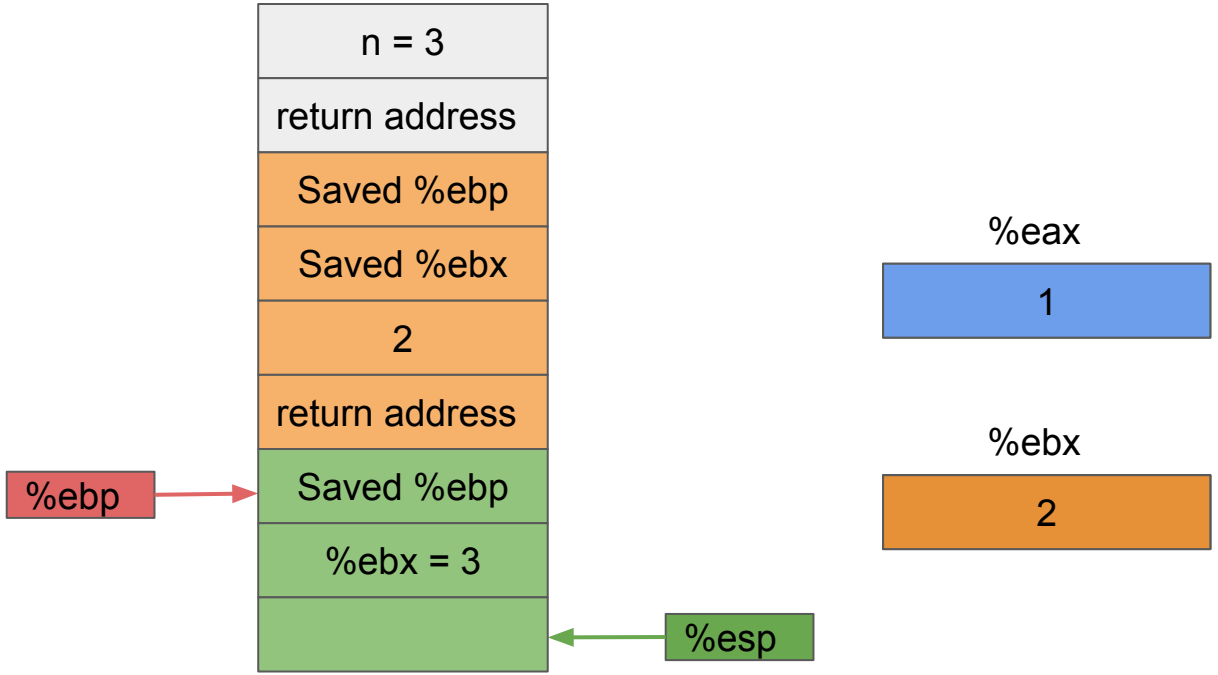
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



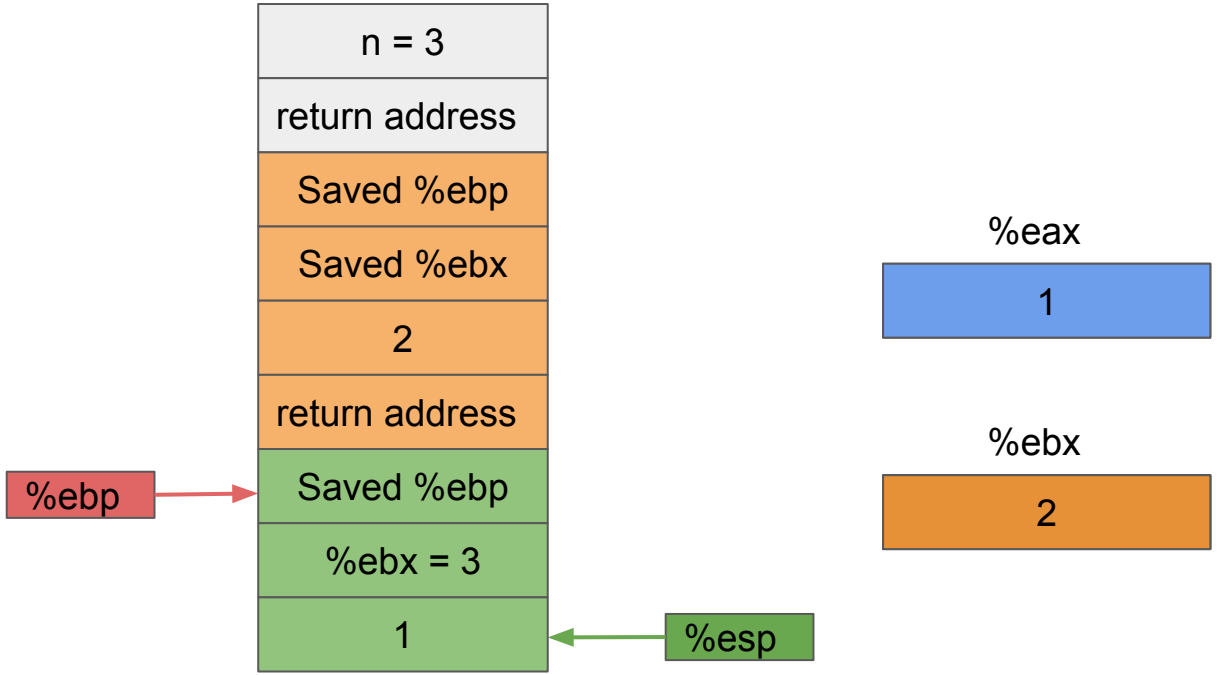
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



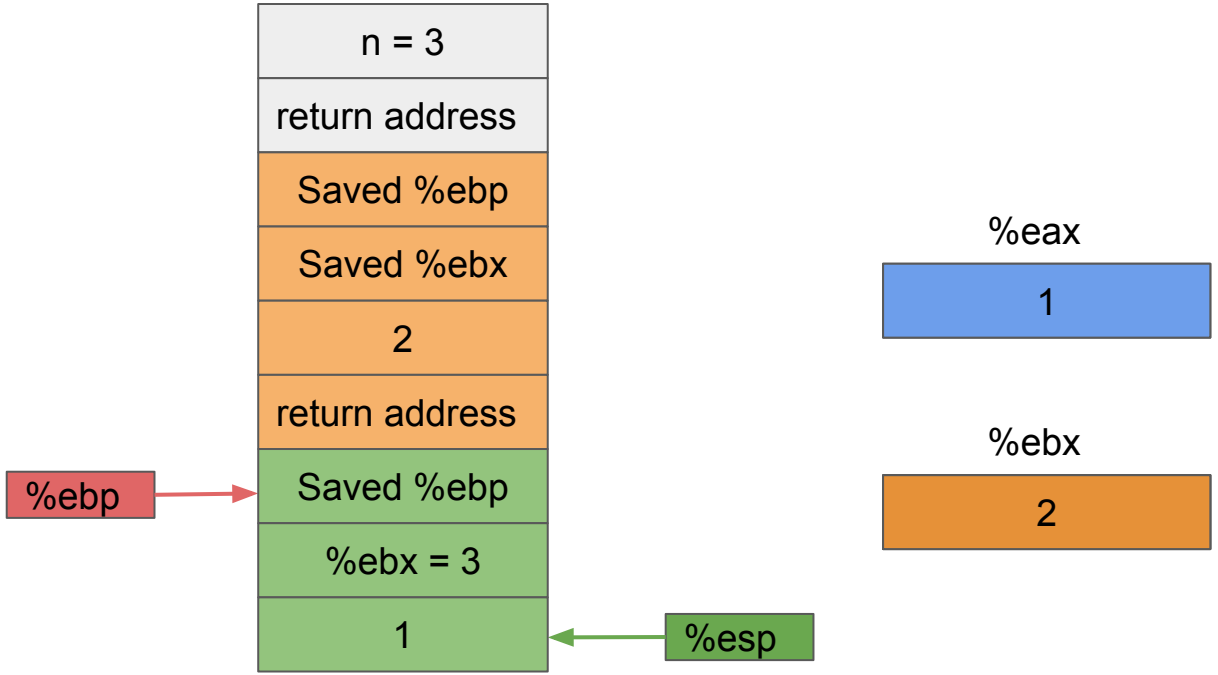
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



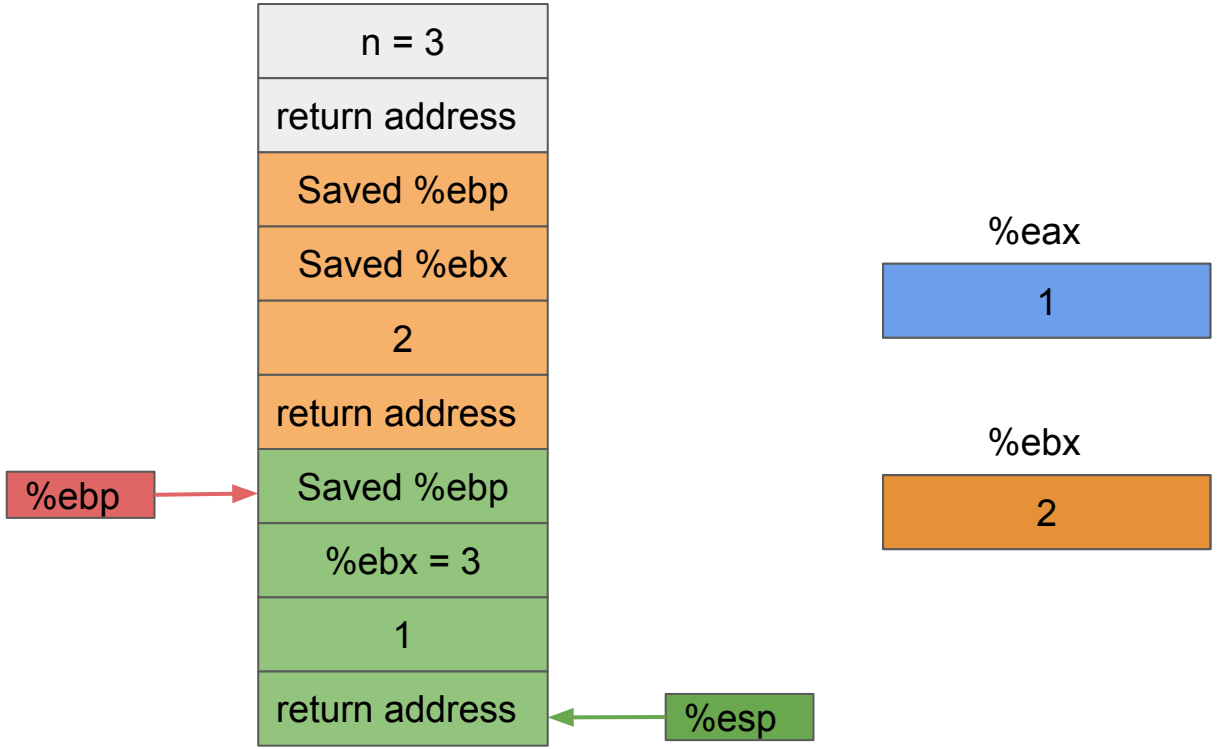
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



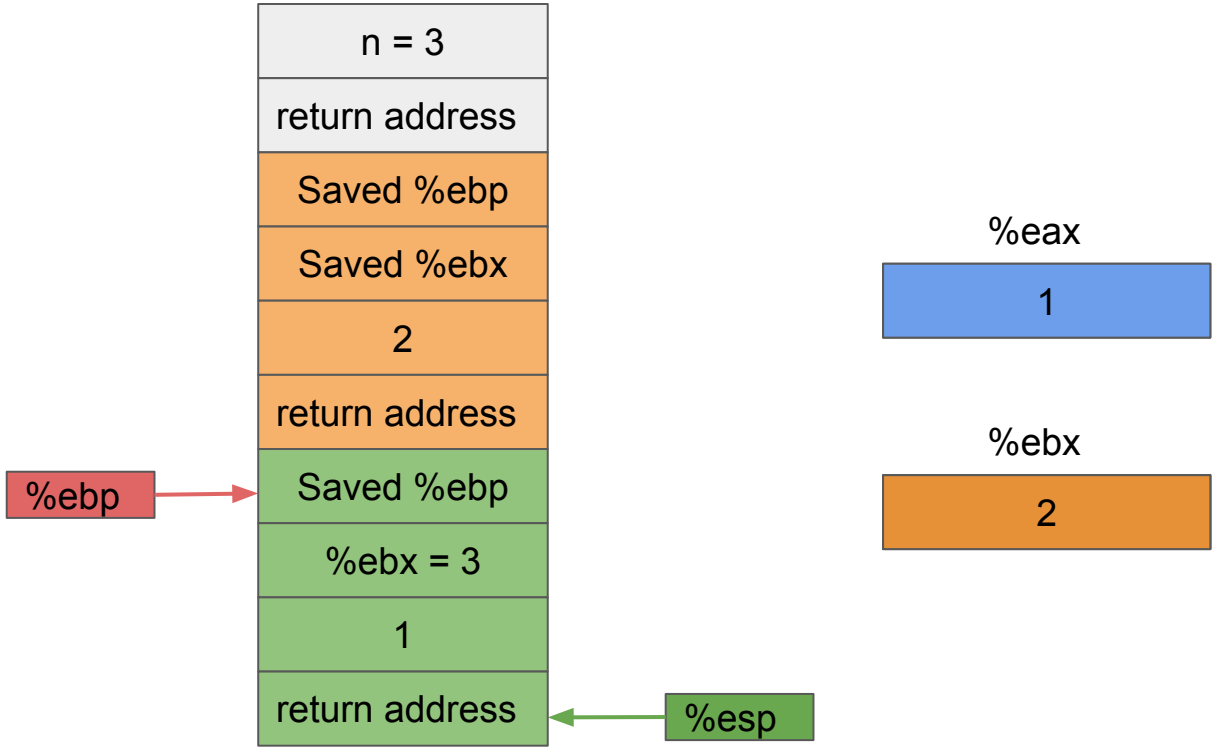
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



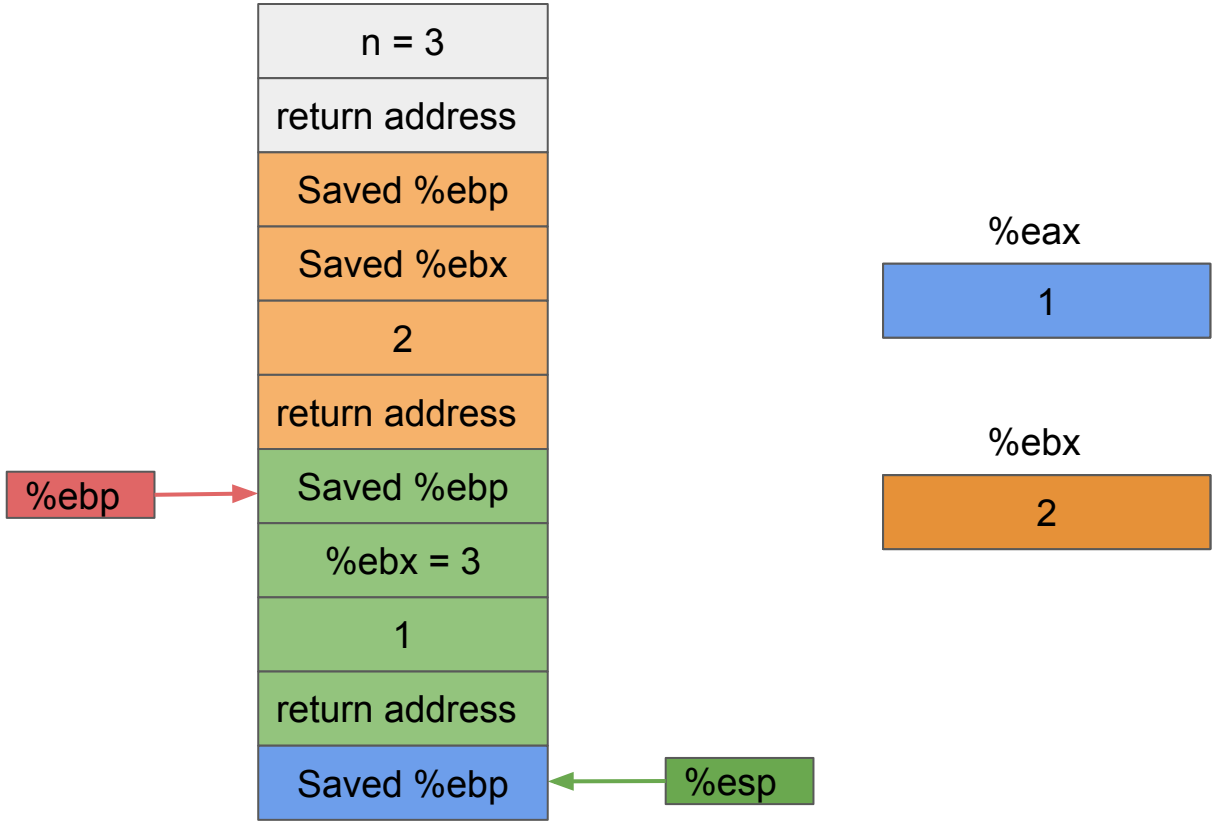
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



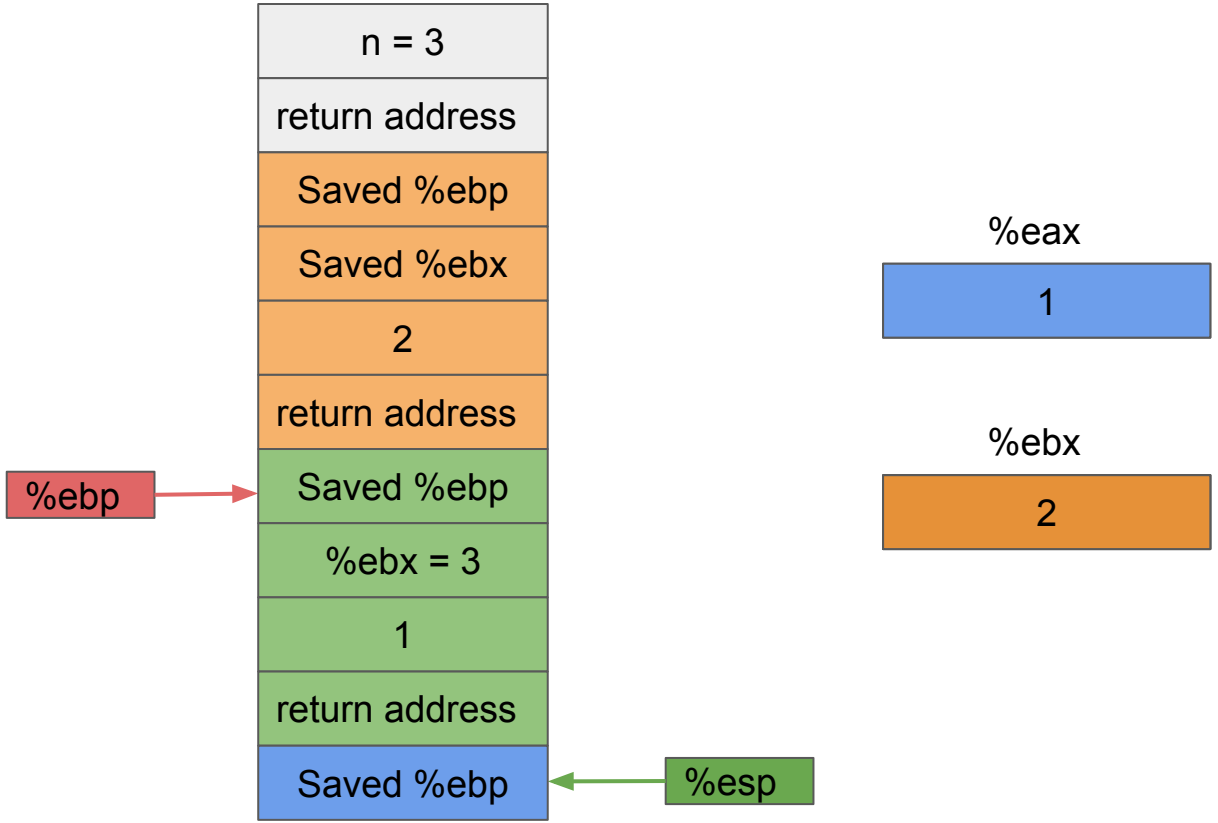
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



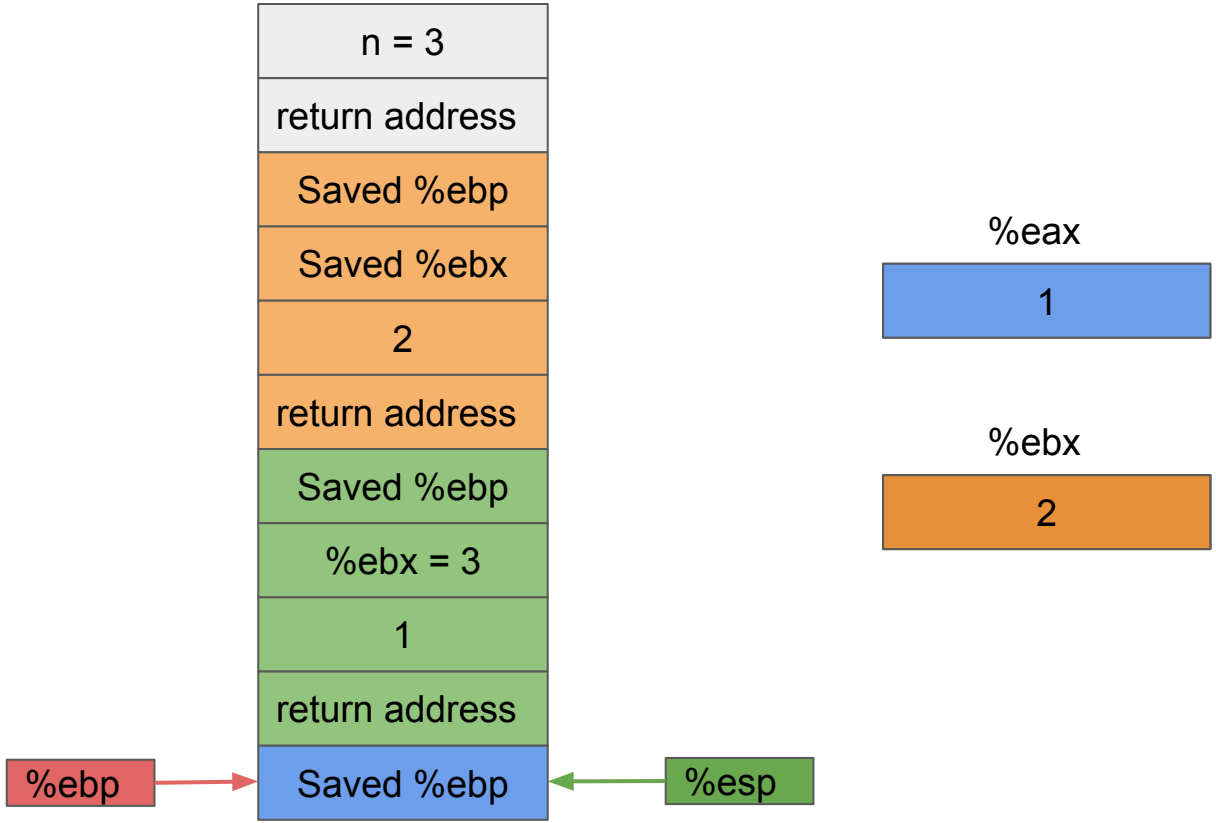
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



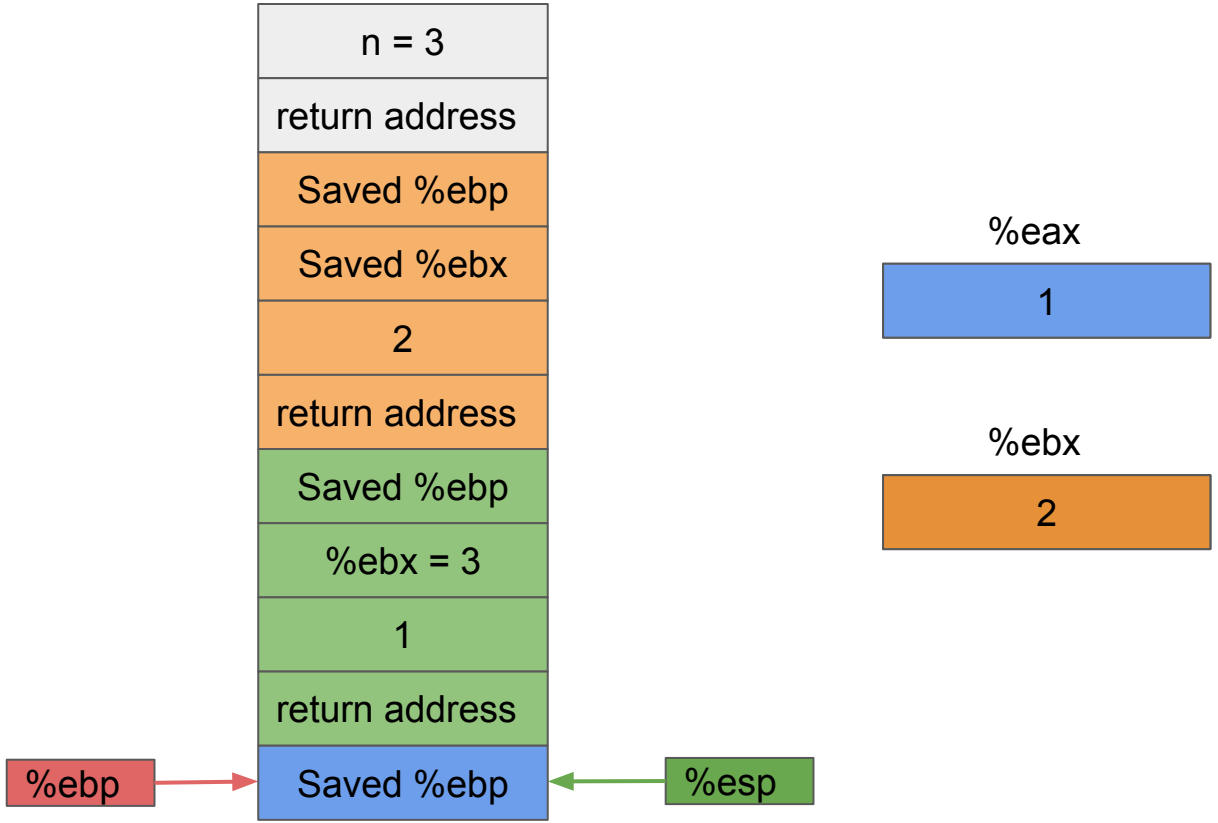
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



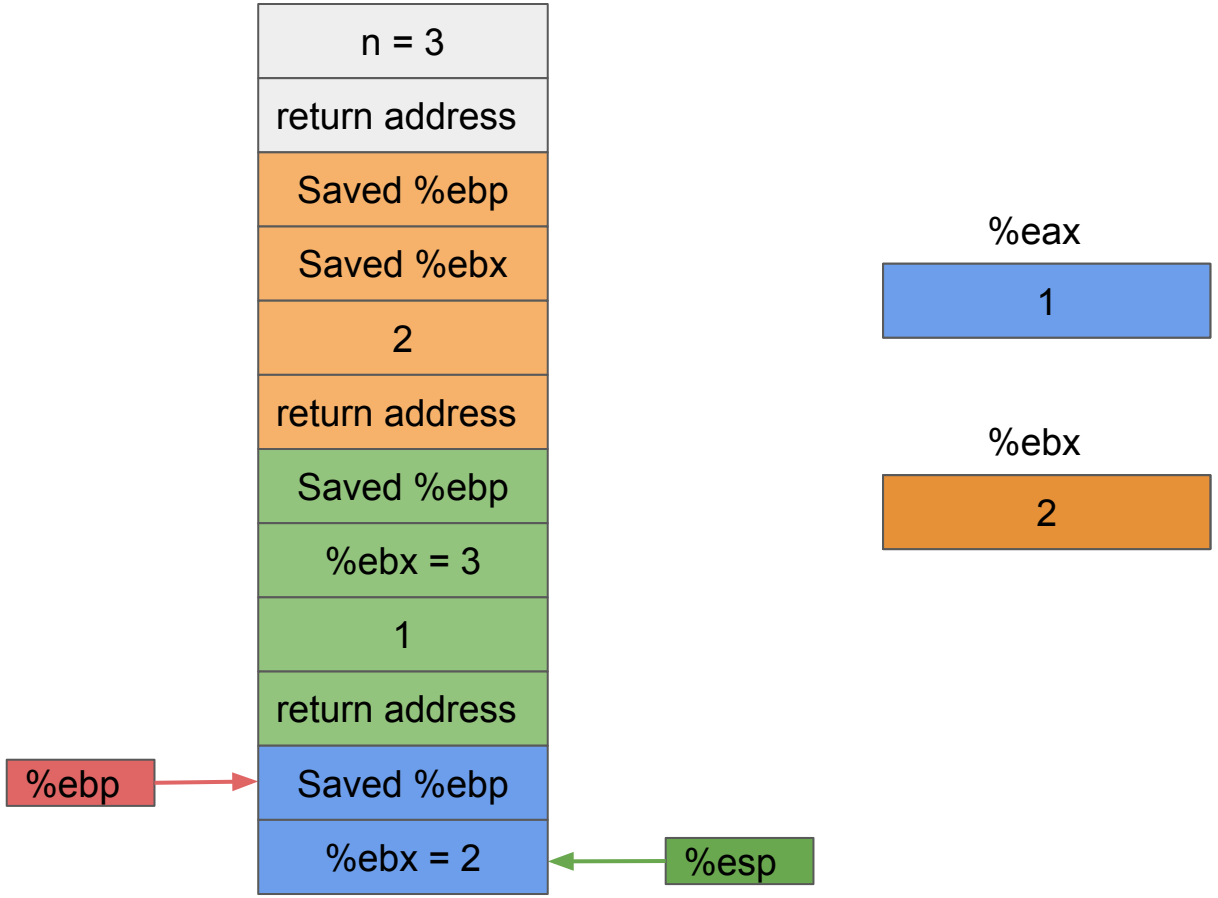
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



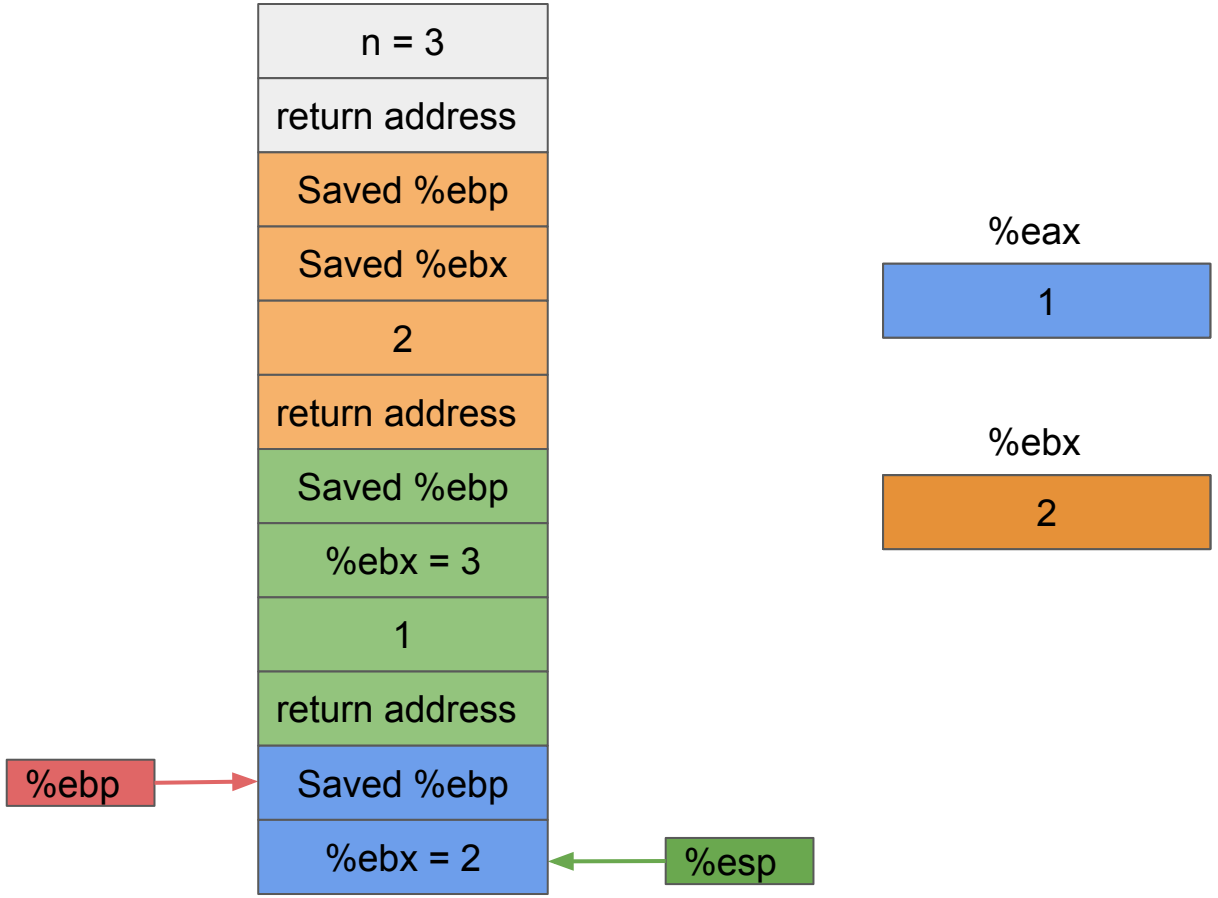
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



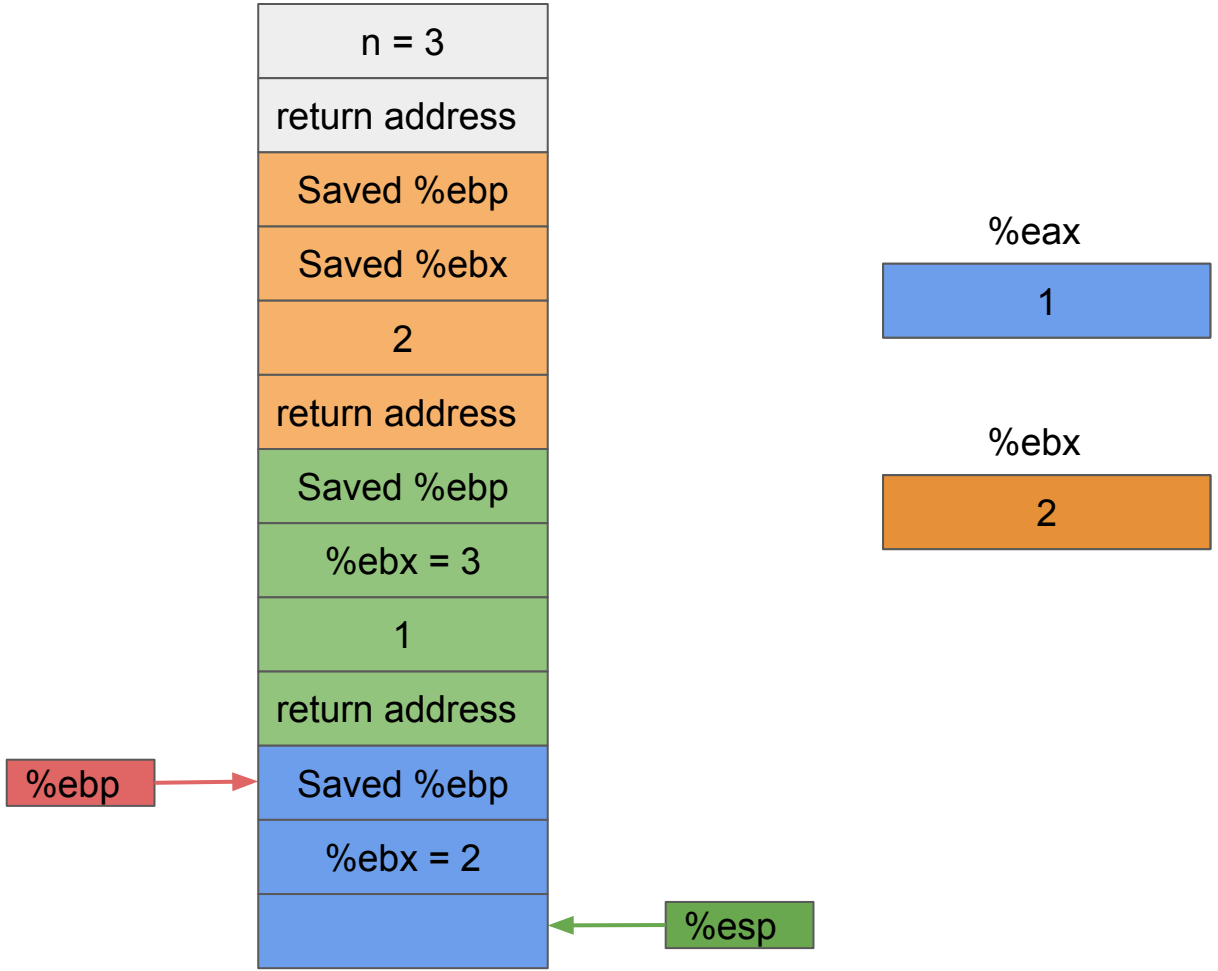
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



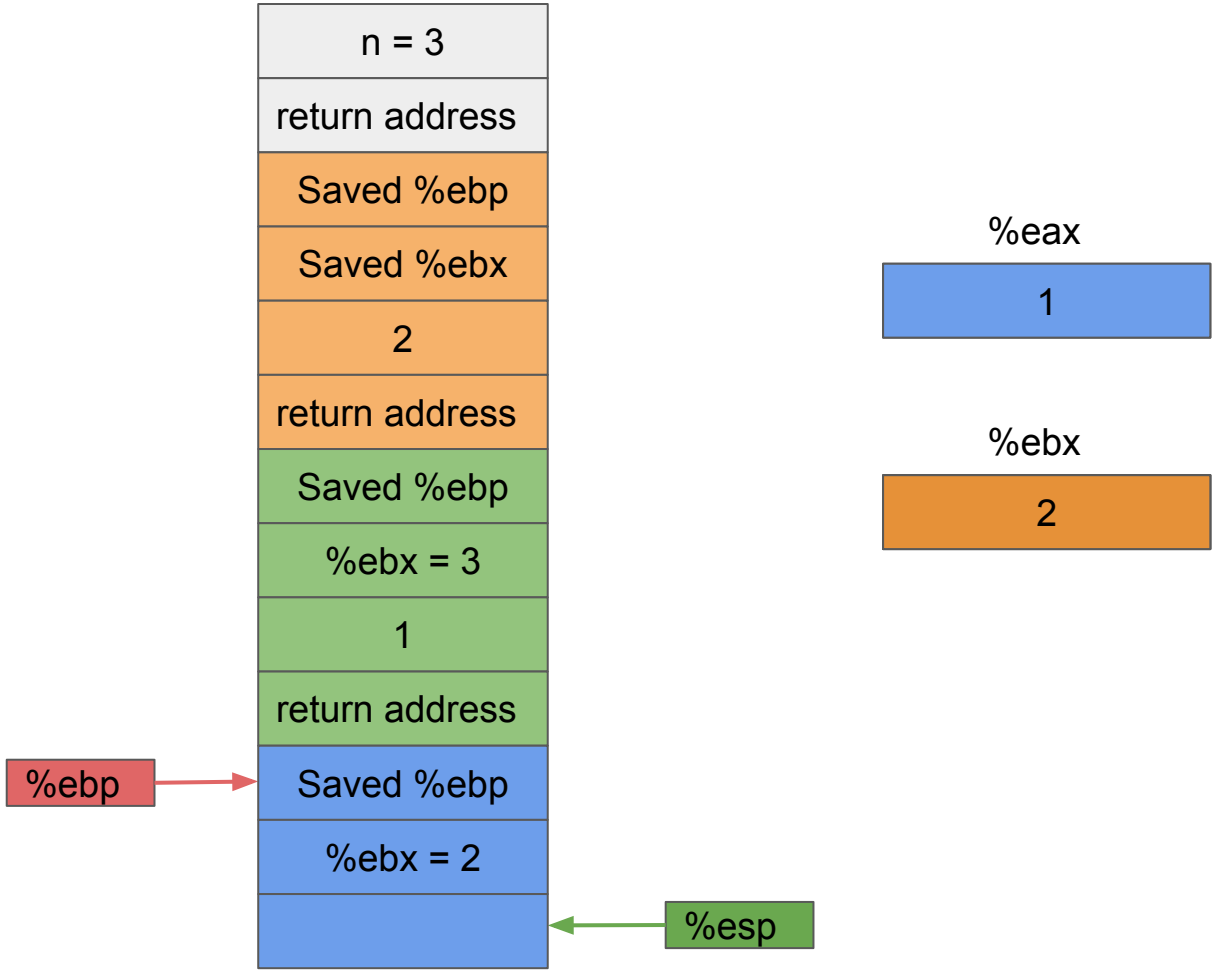
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



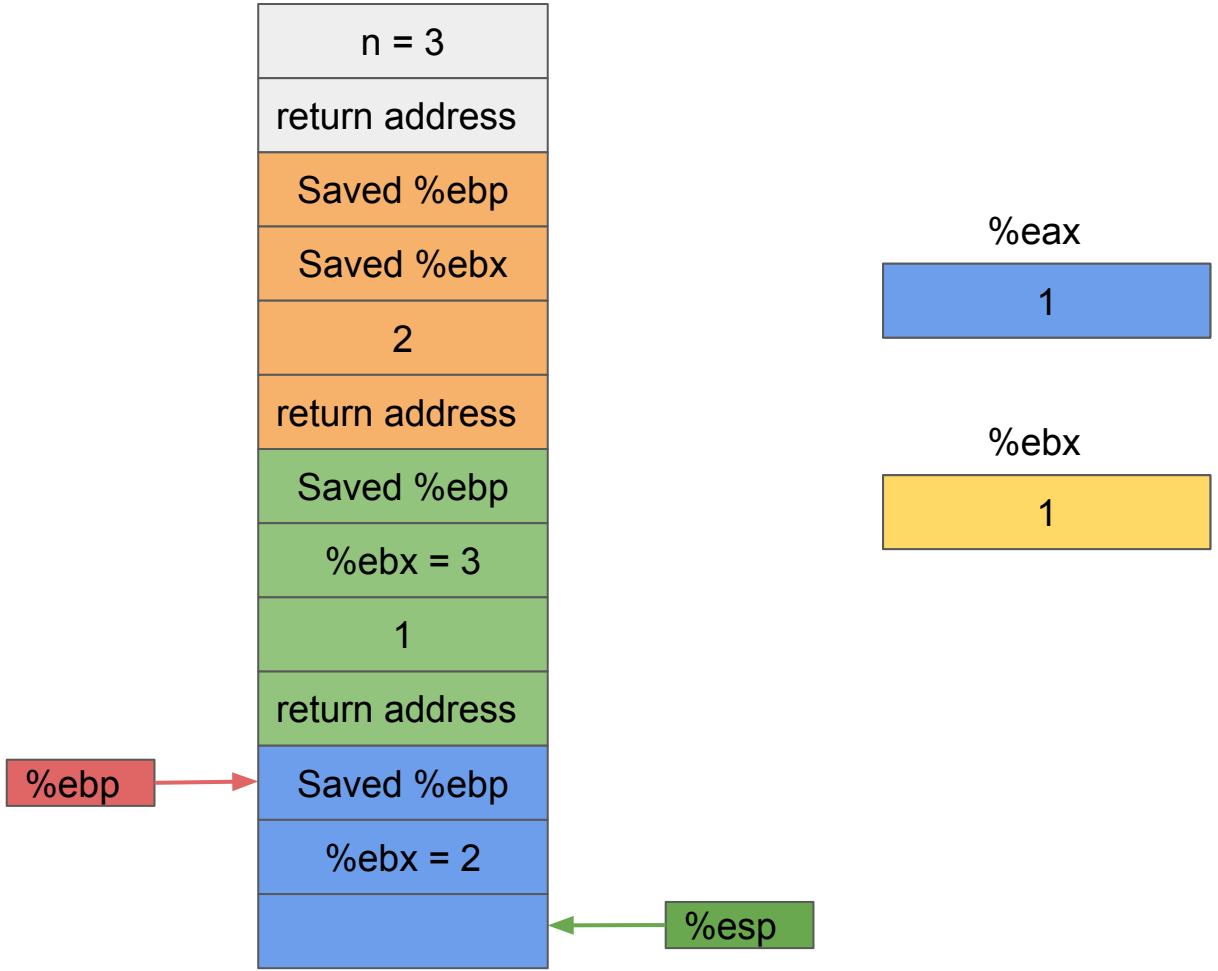
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



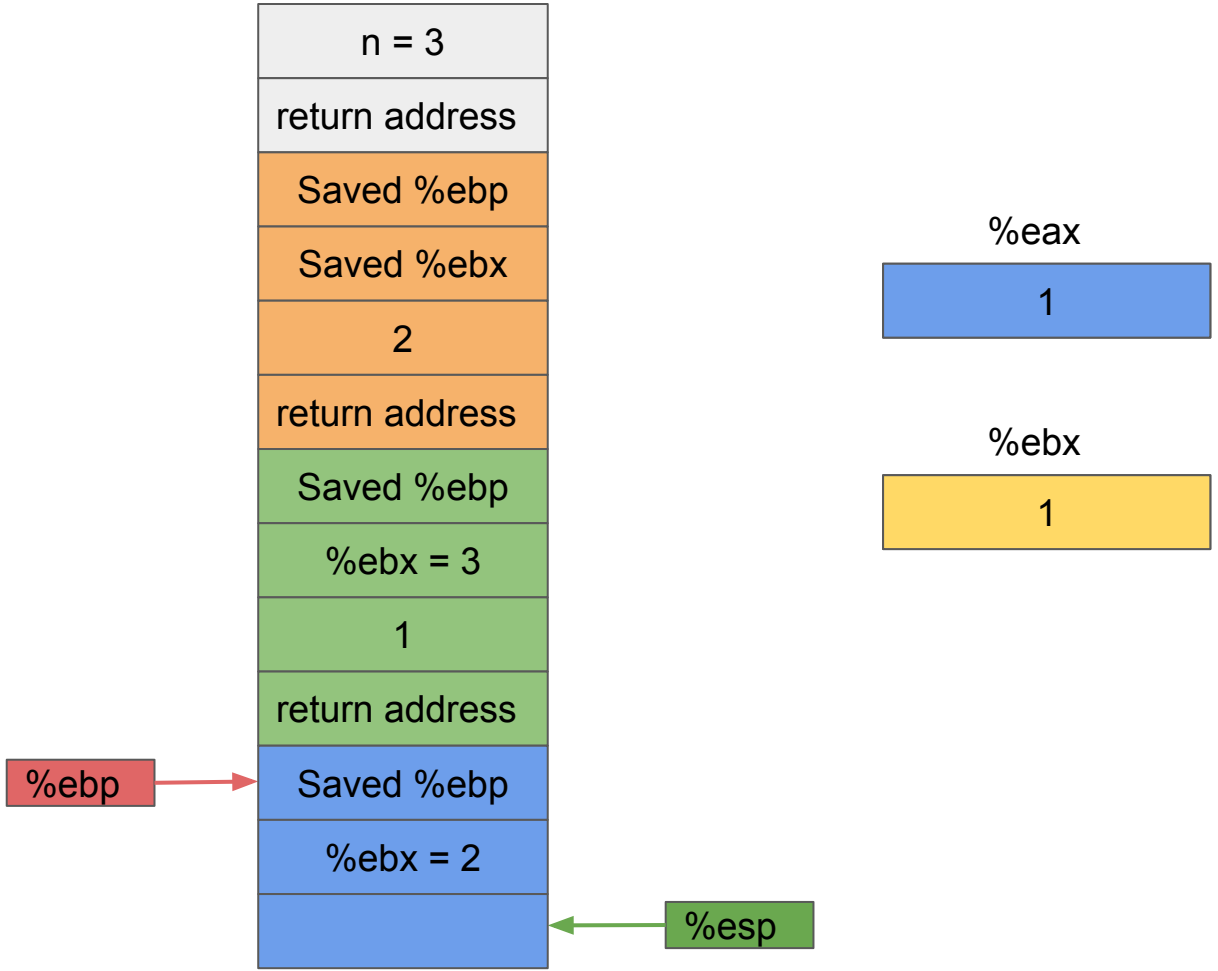
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



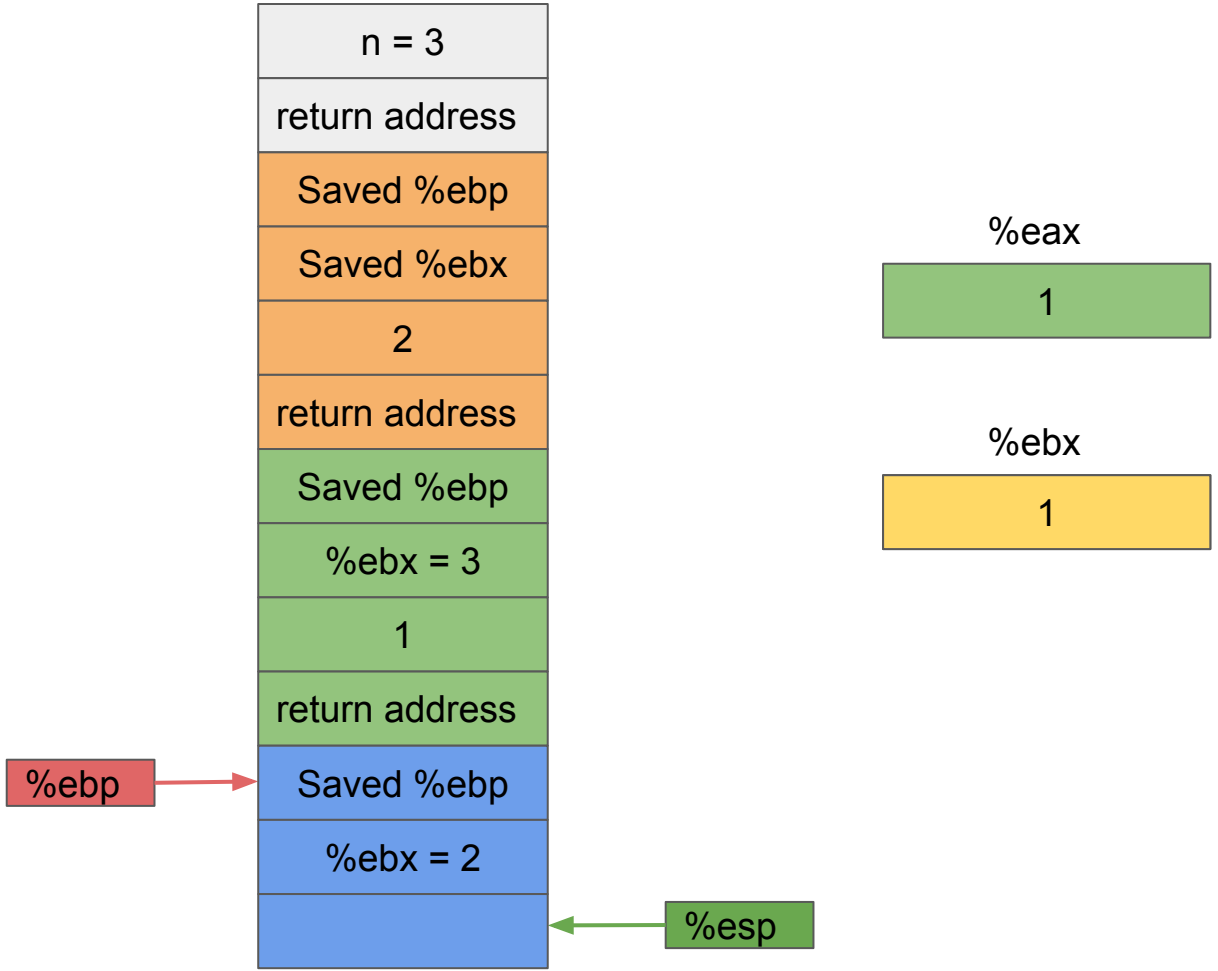
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



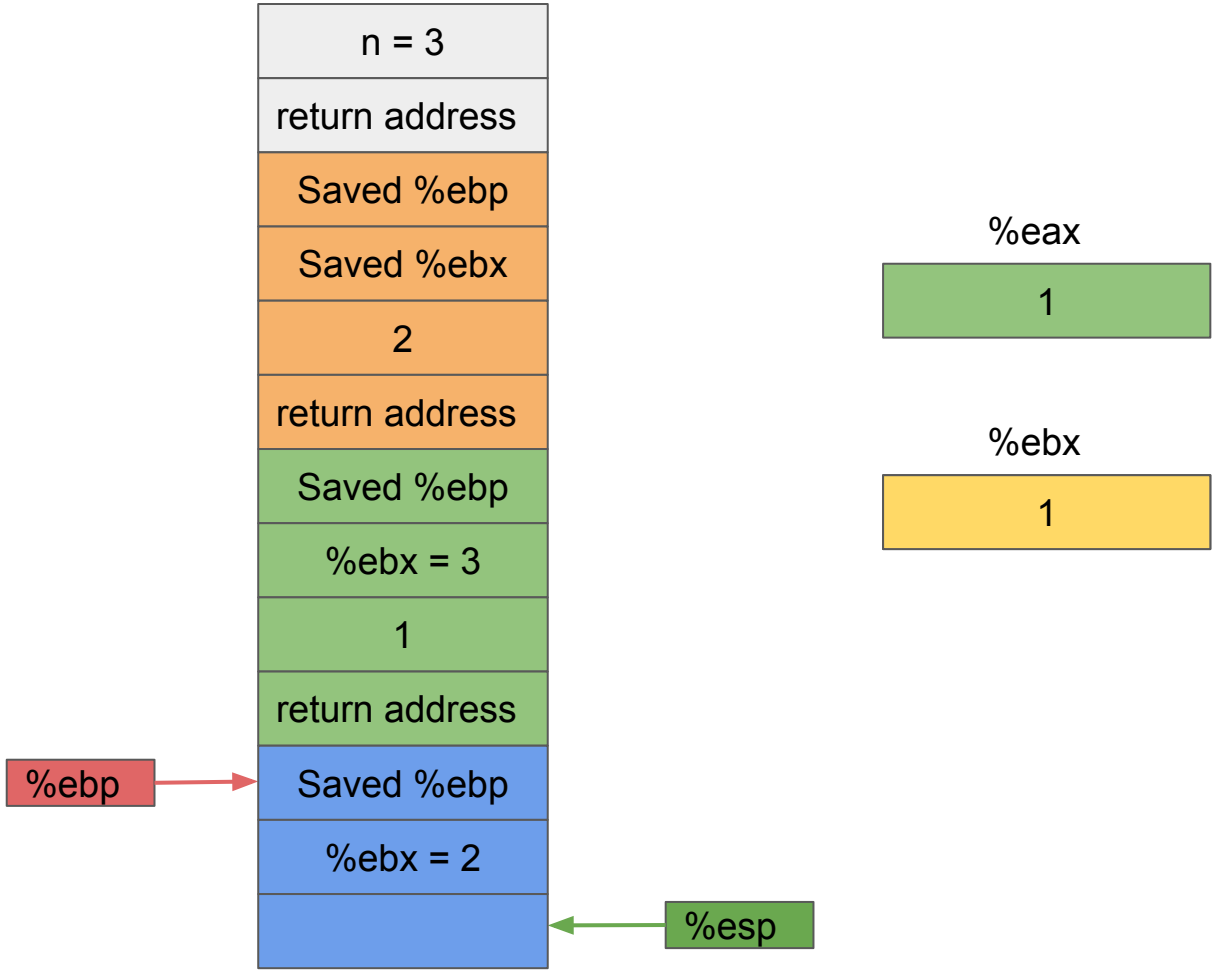
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



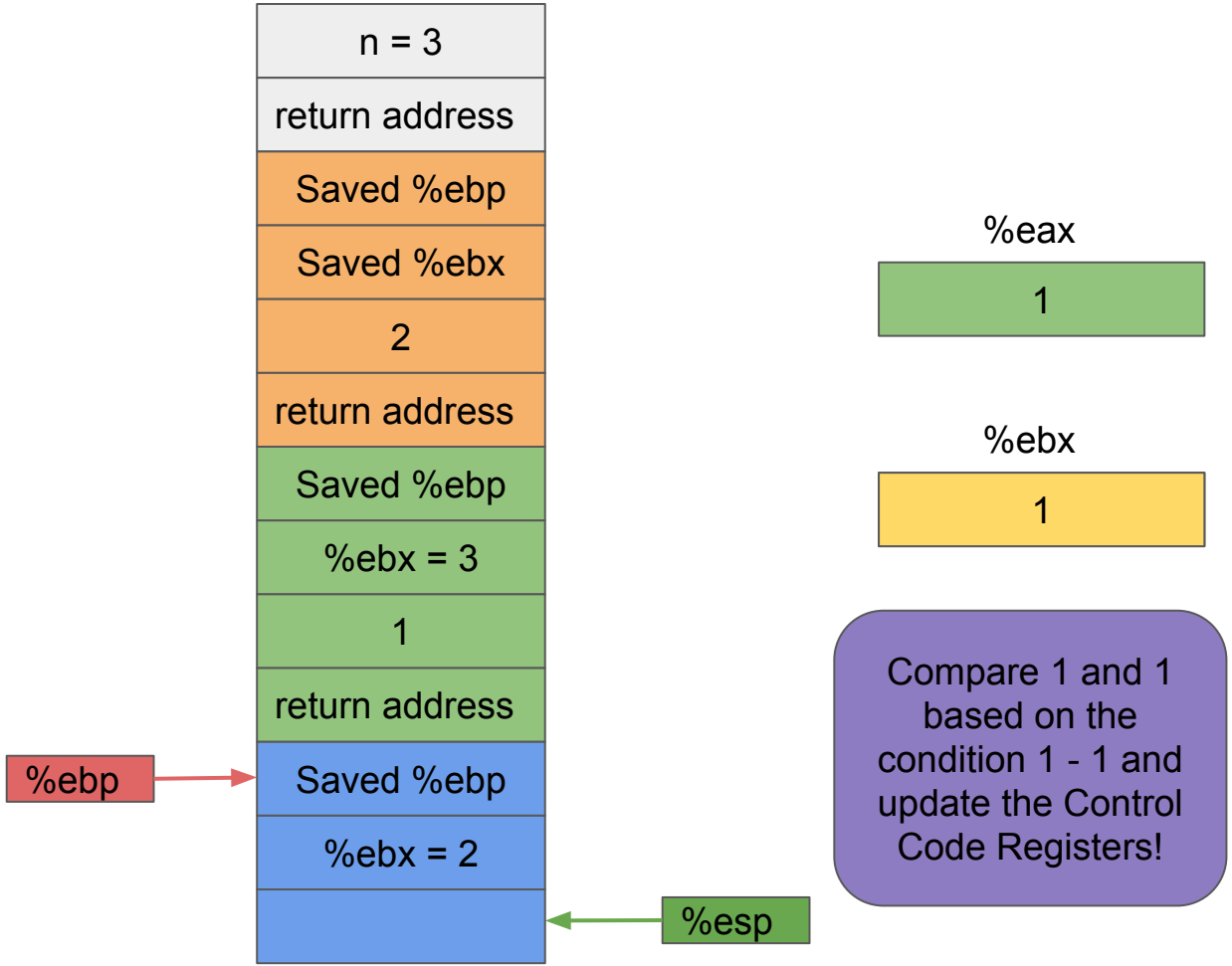
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



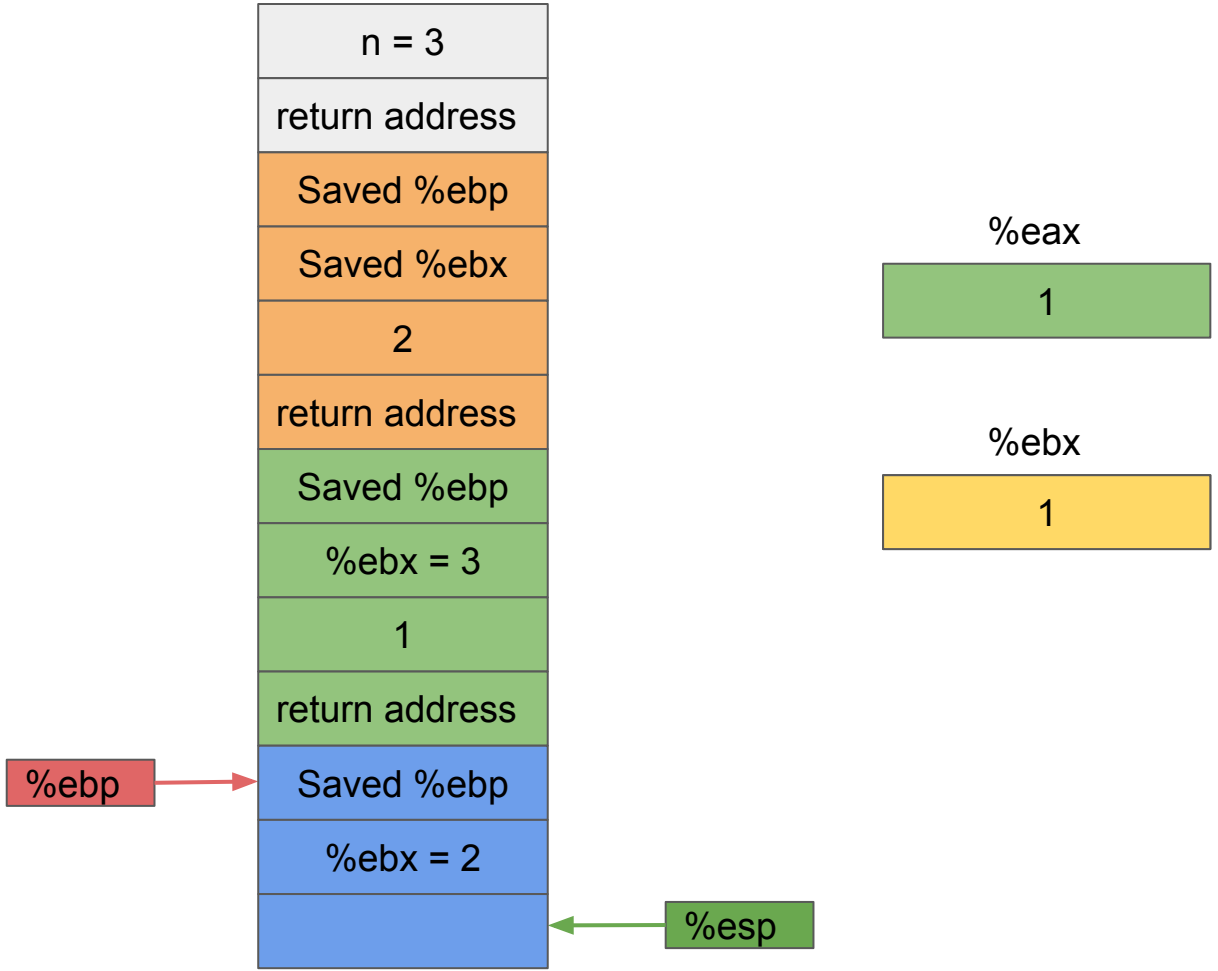
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



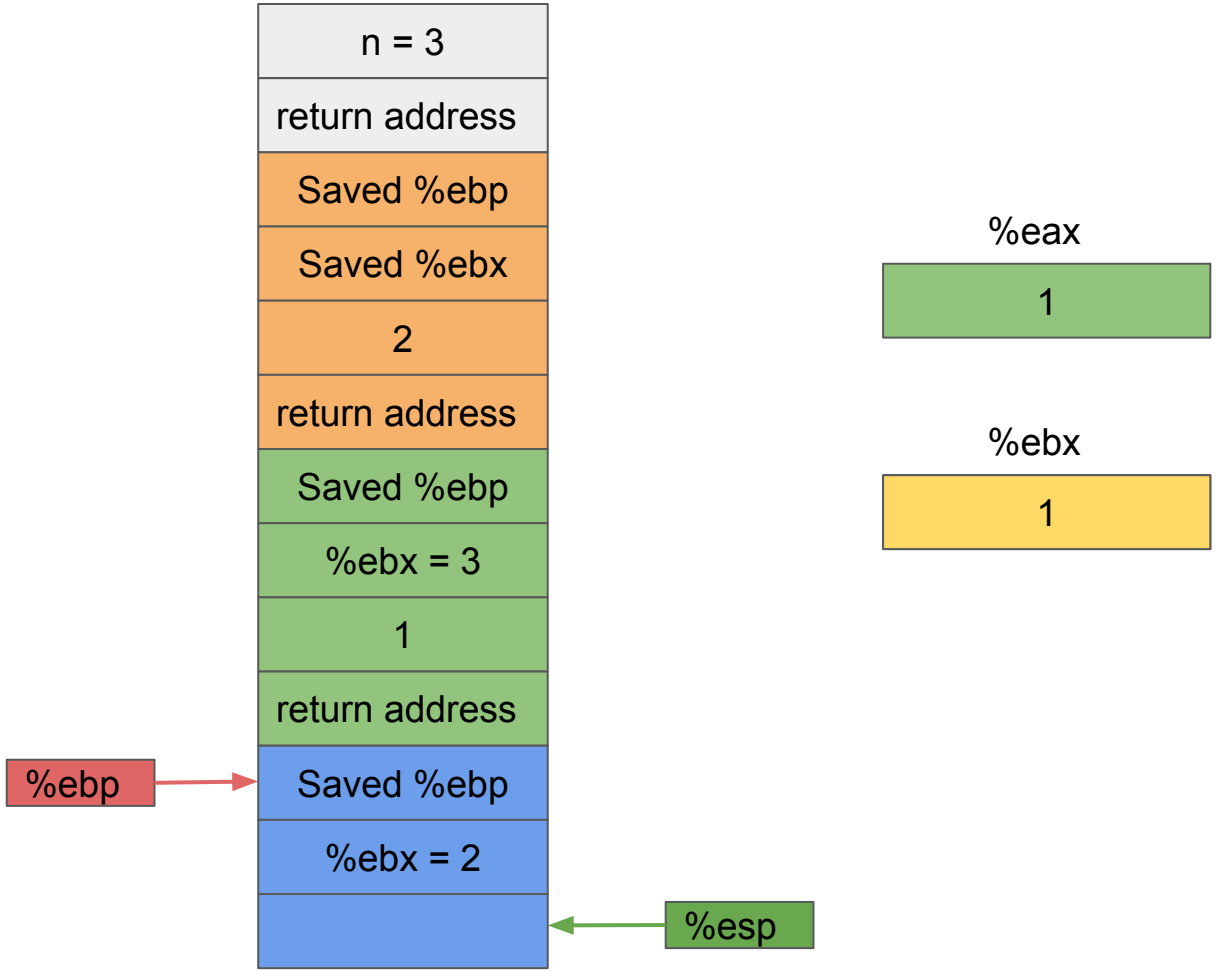
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



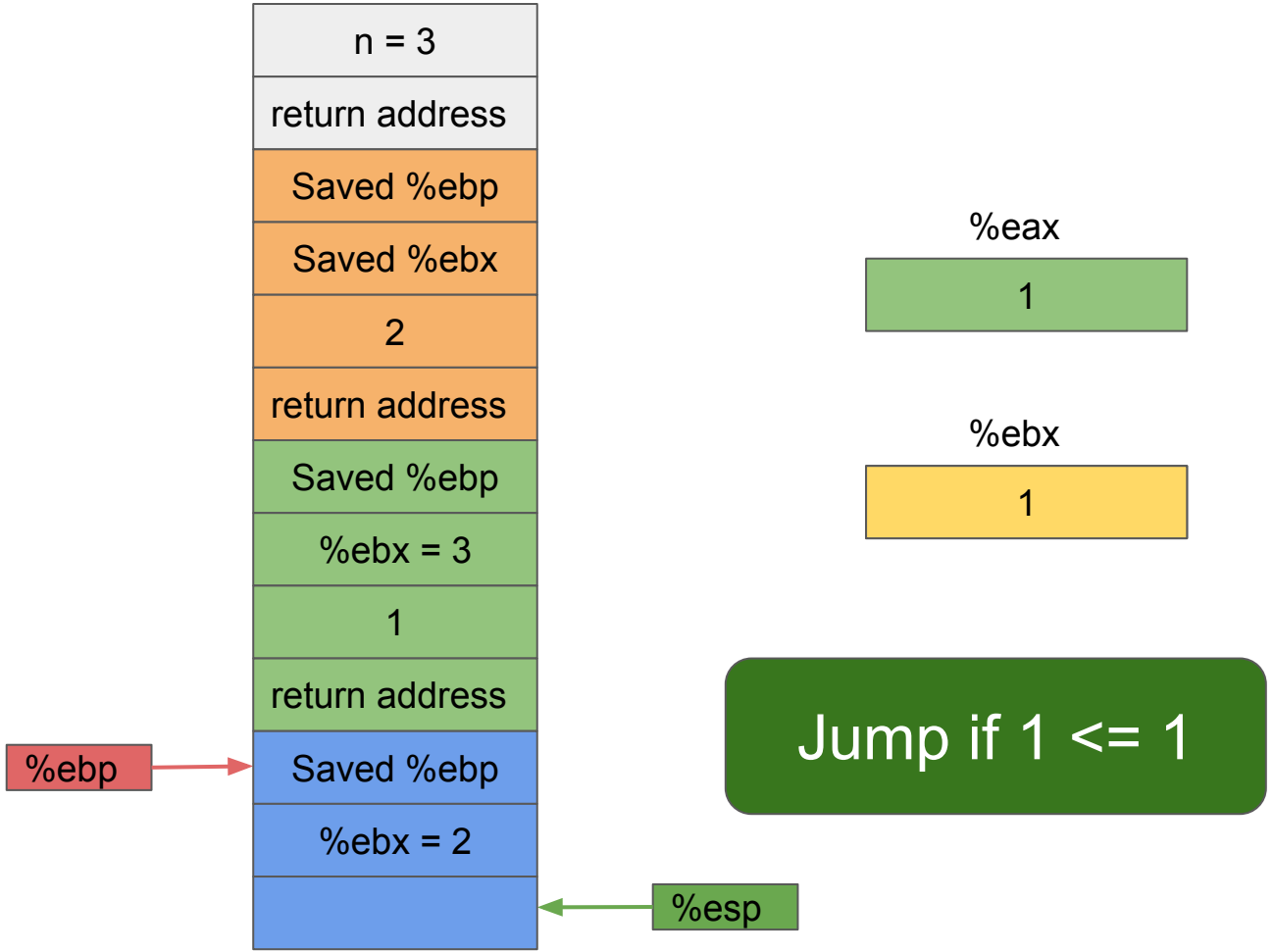
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



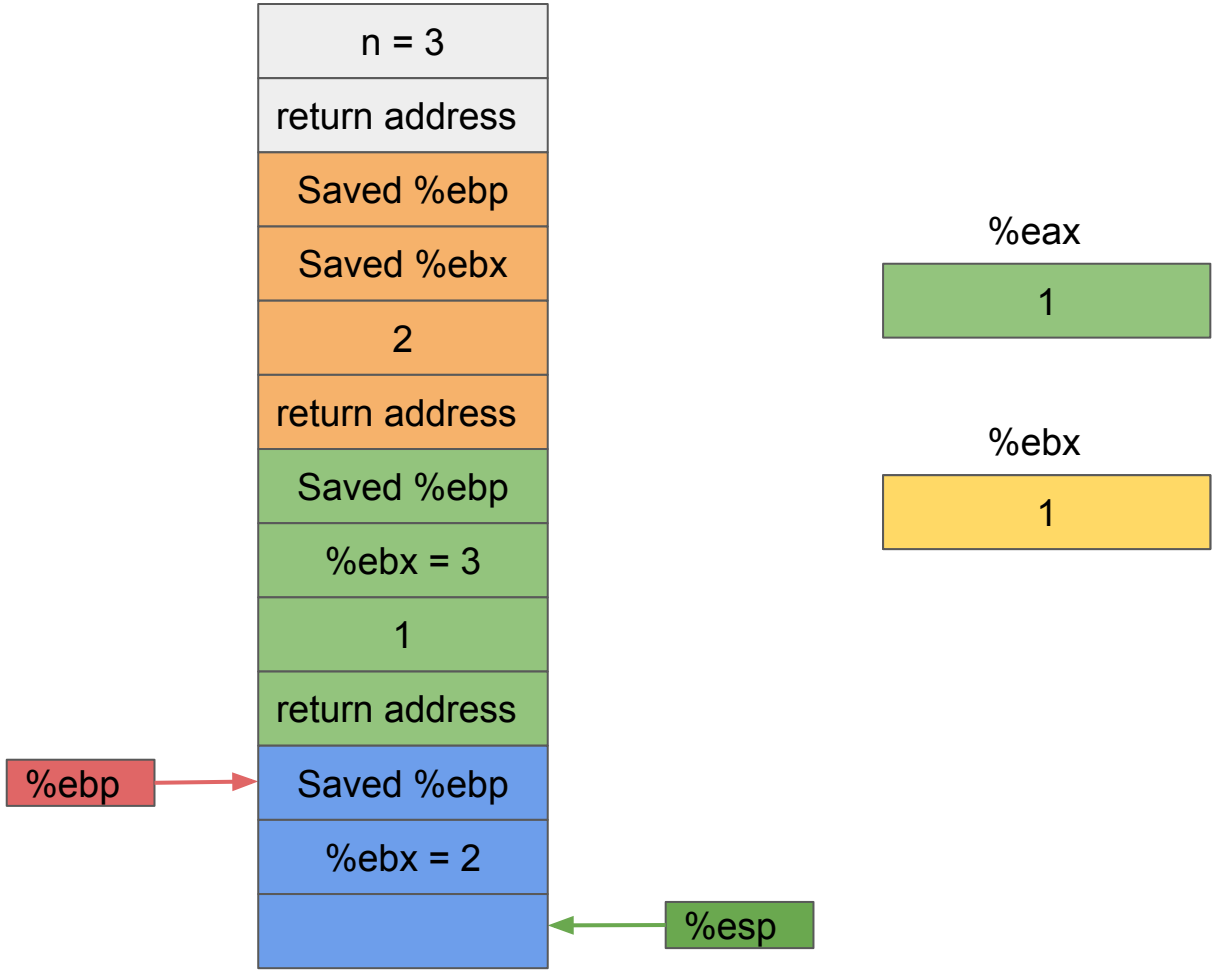
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



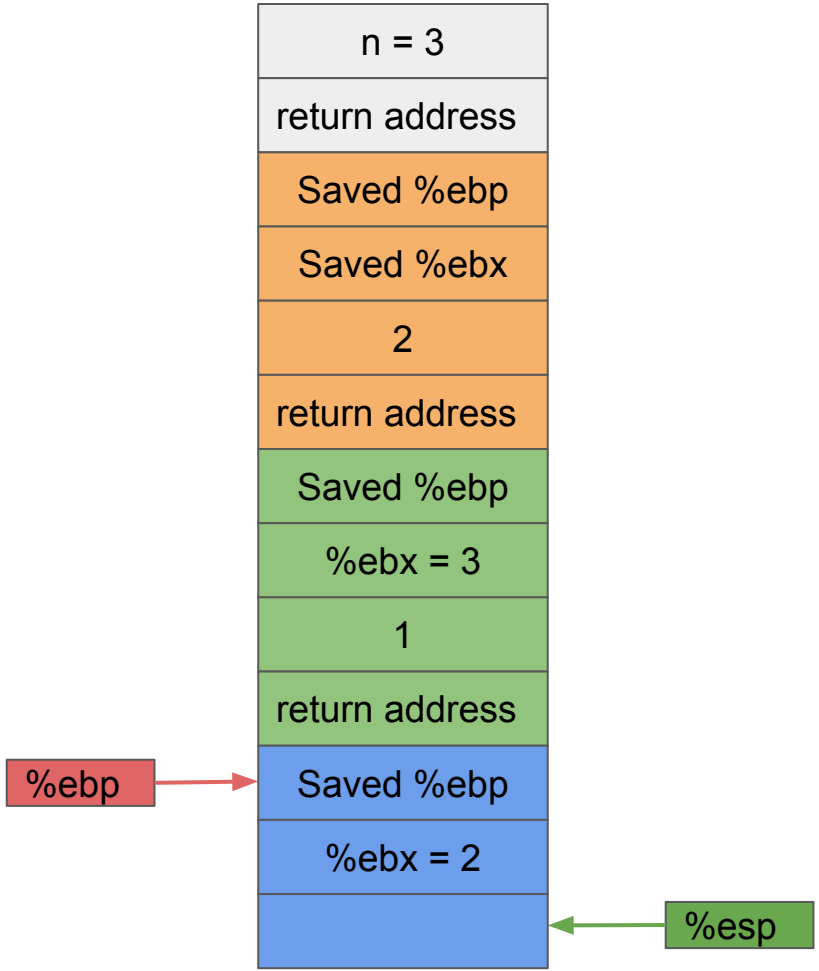
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```

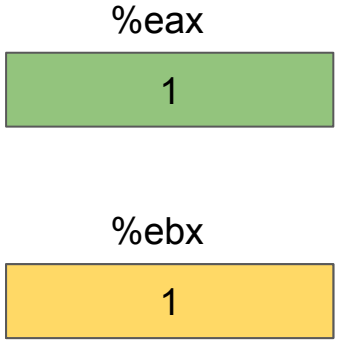


```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```

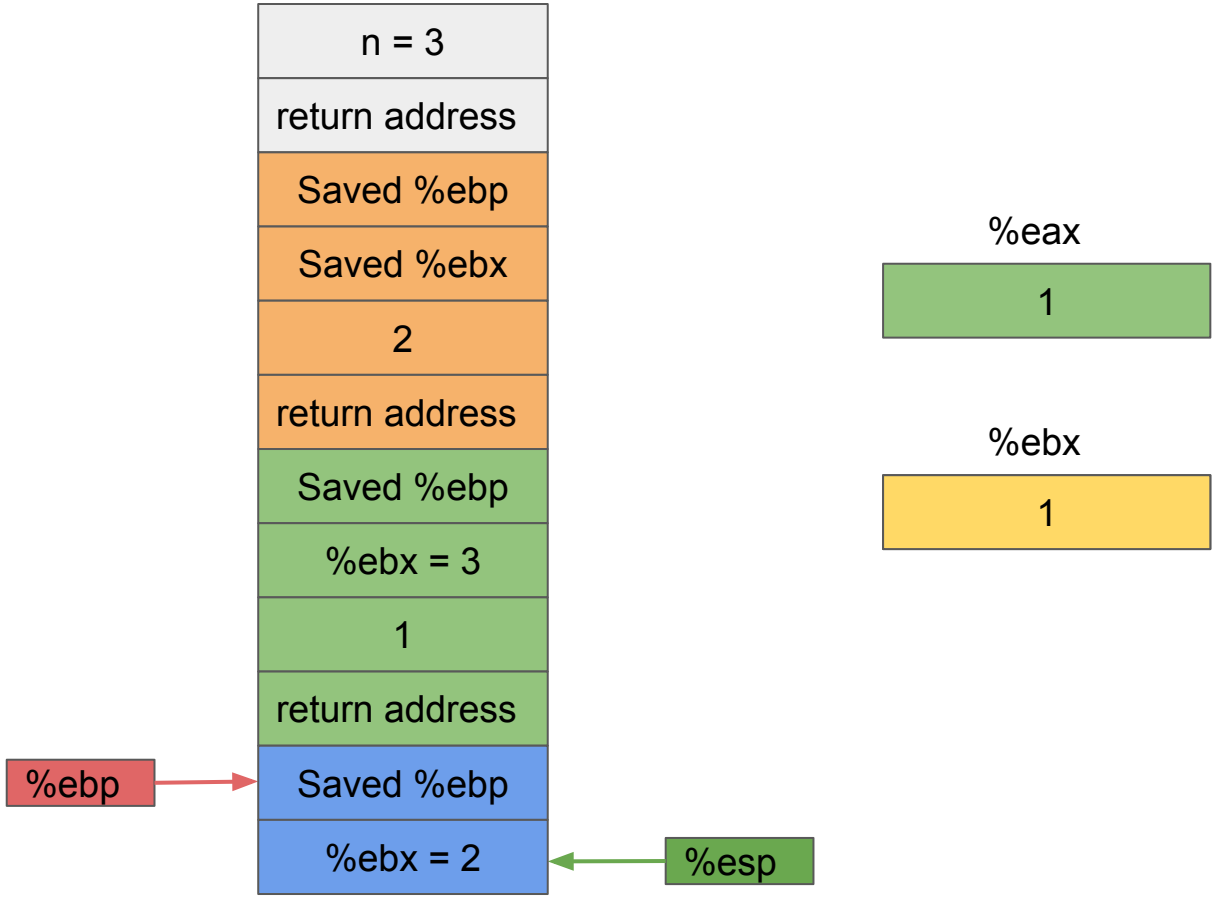


```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```

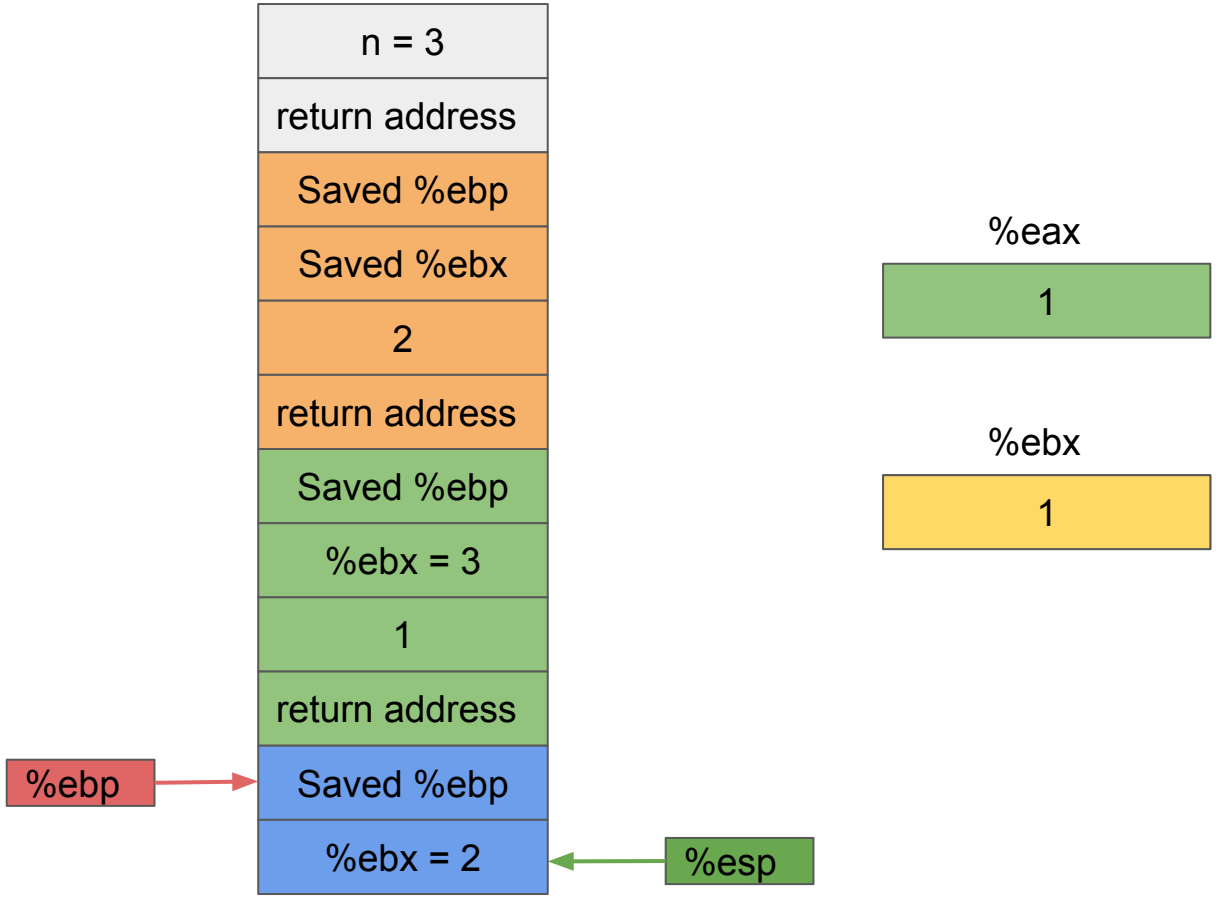




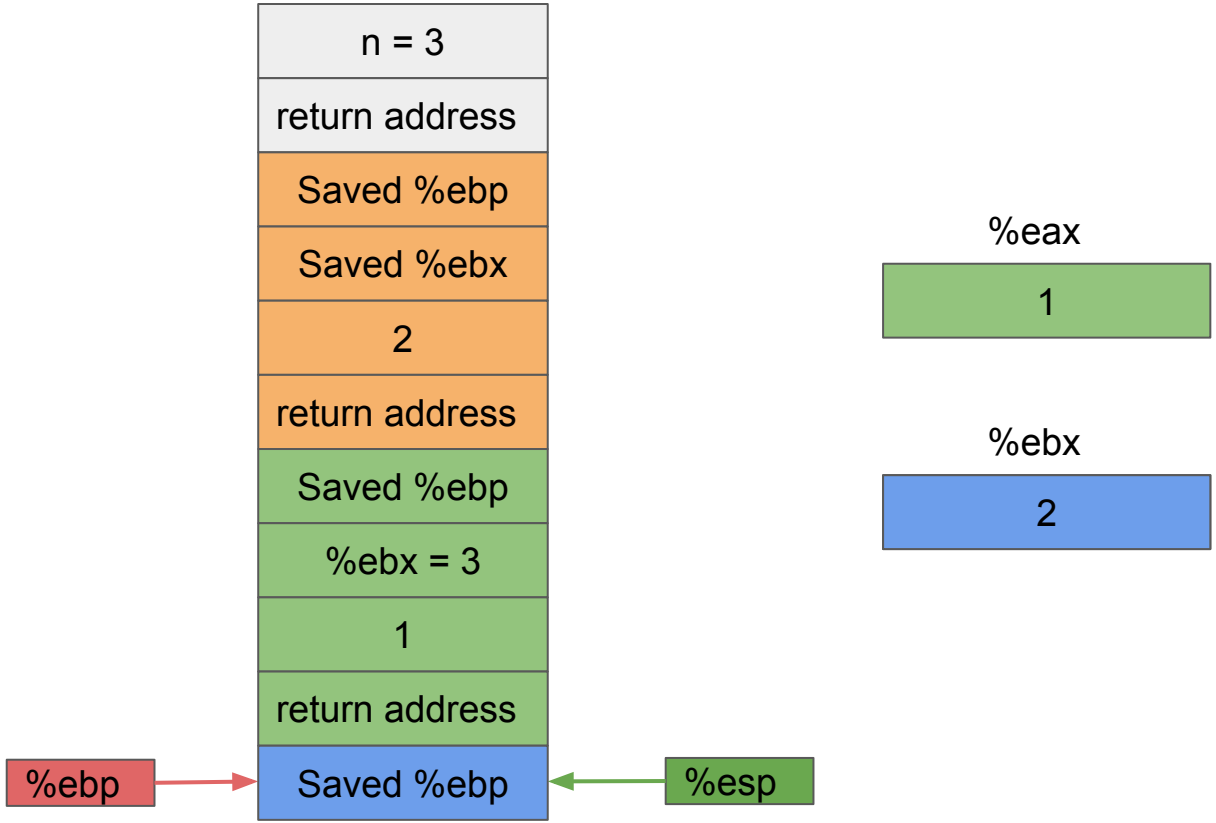
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



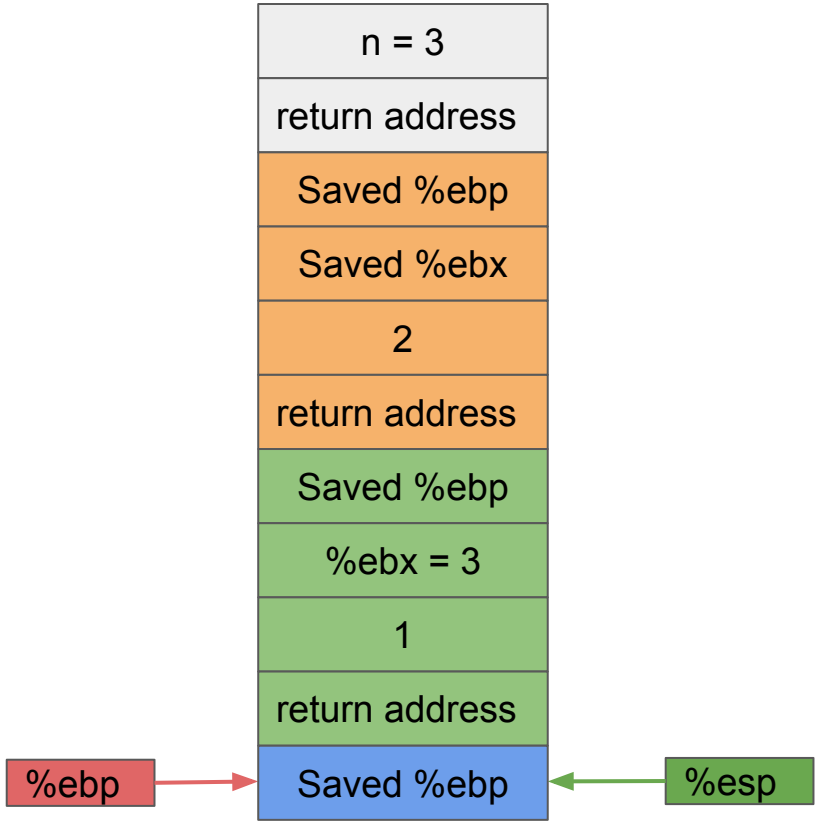
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```

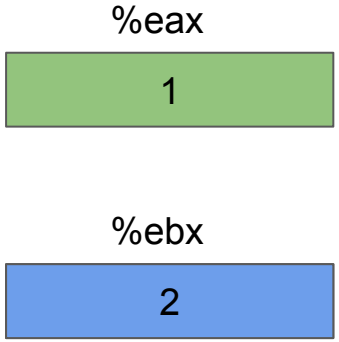


```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```

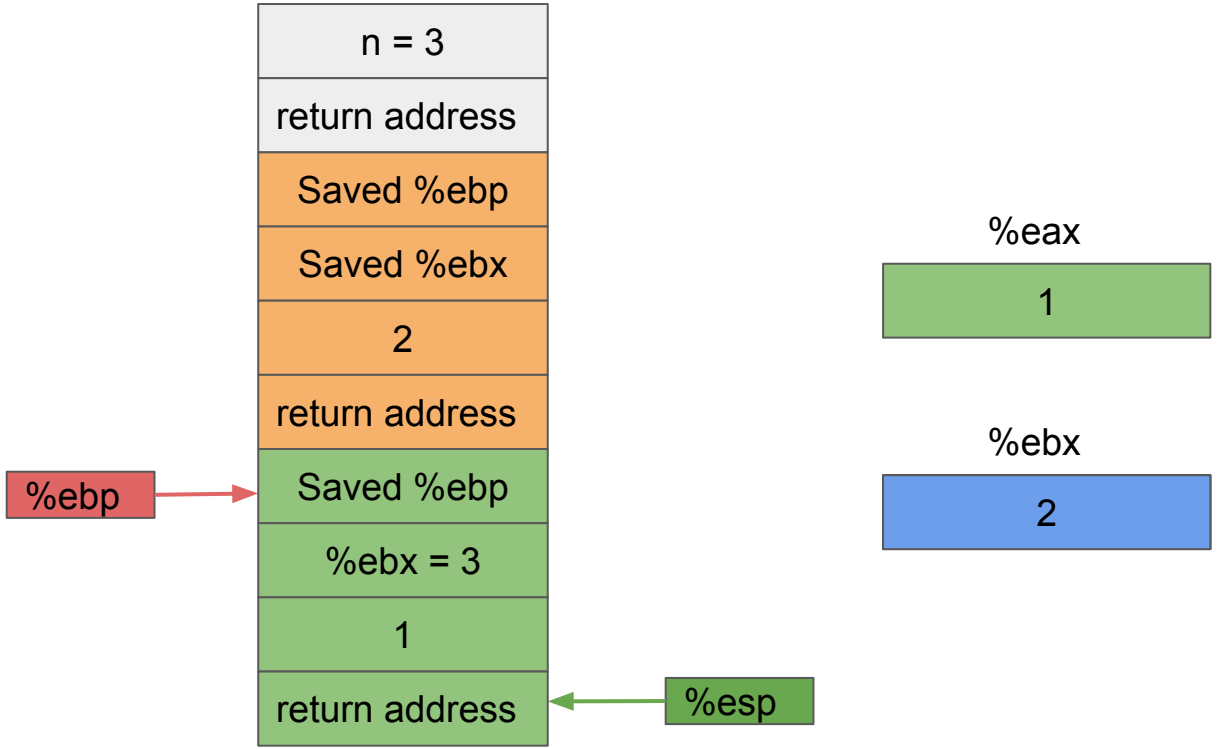


```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```

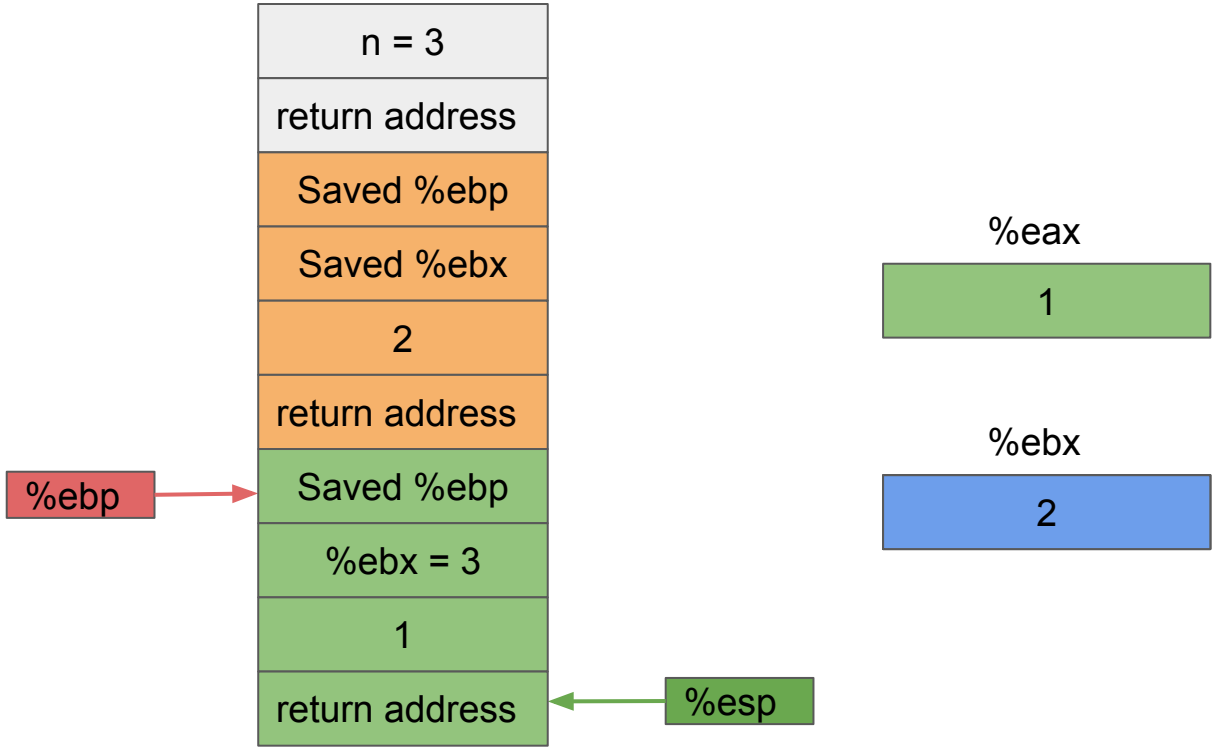




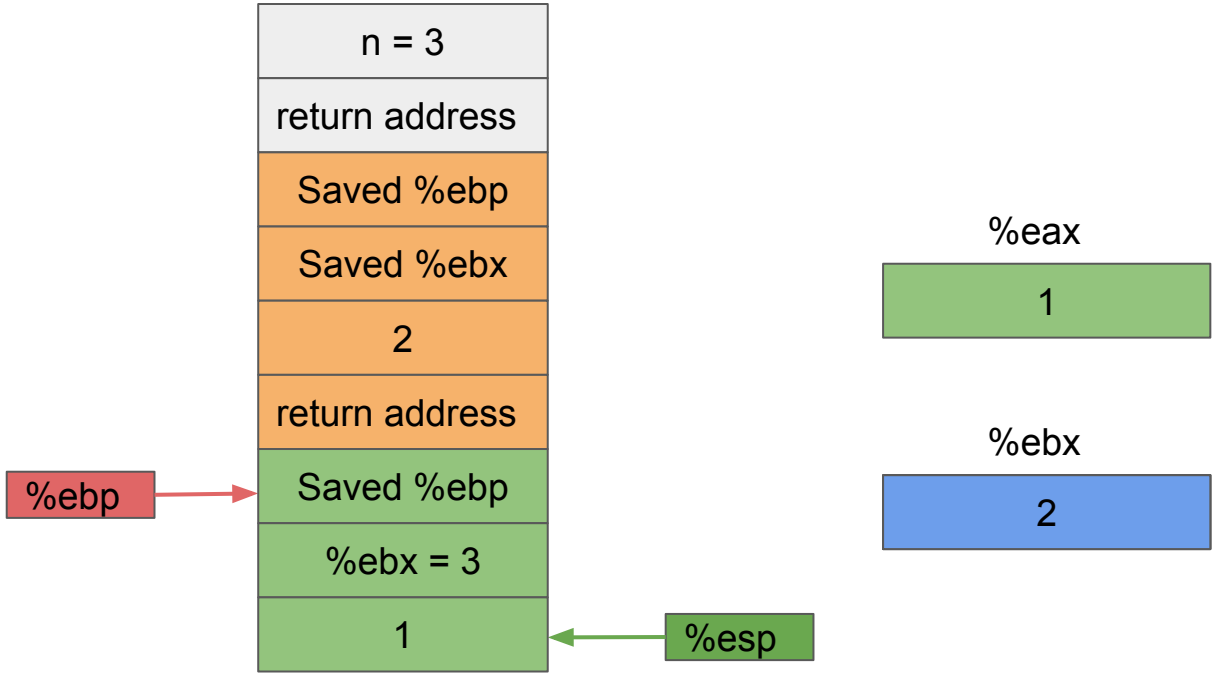
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



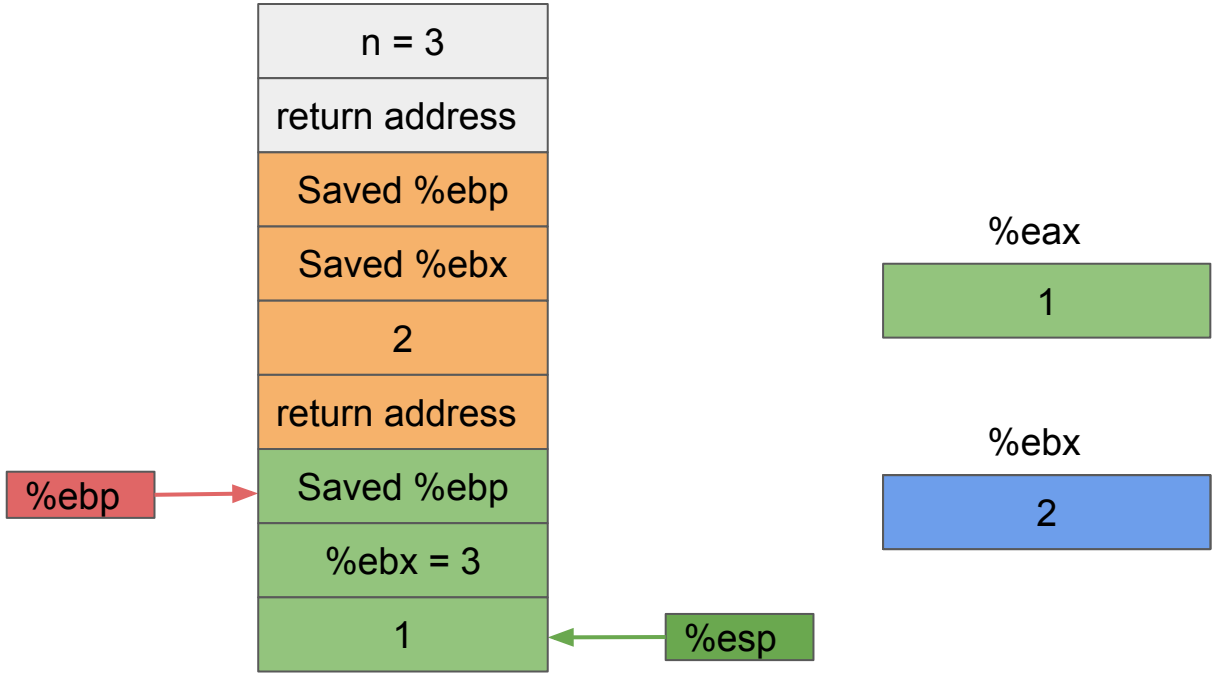
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



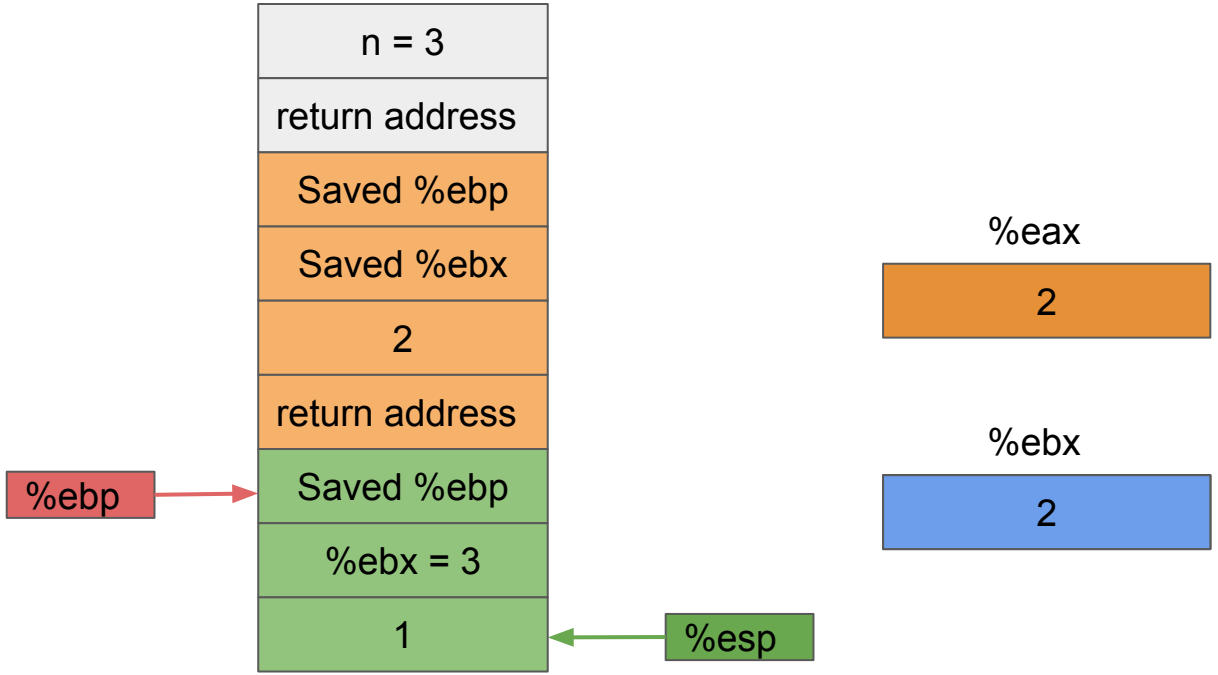
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



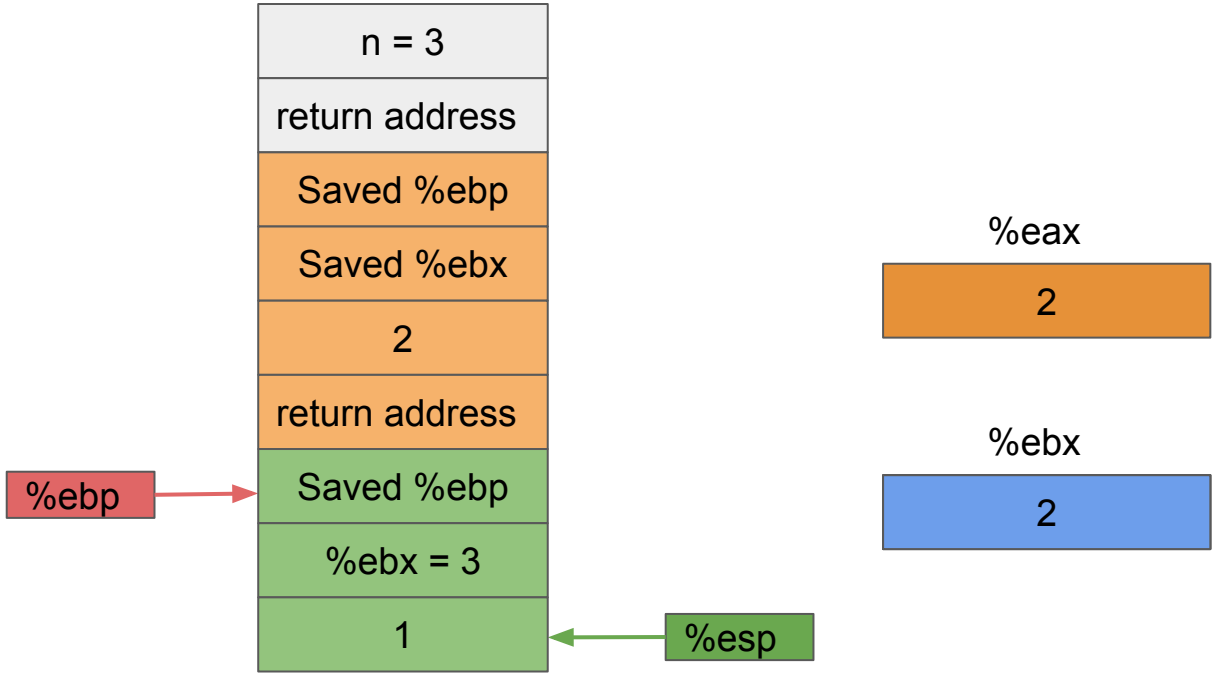
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



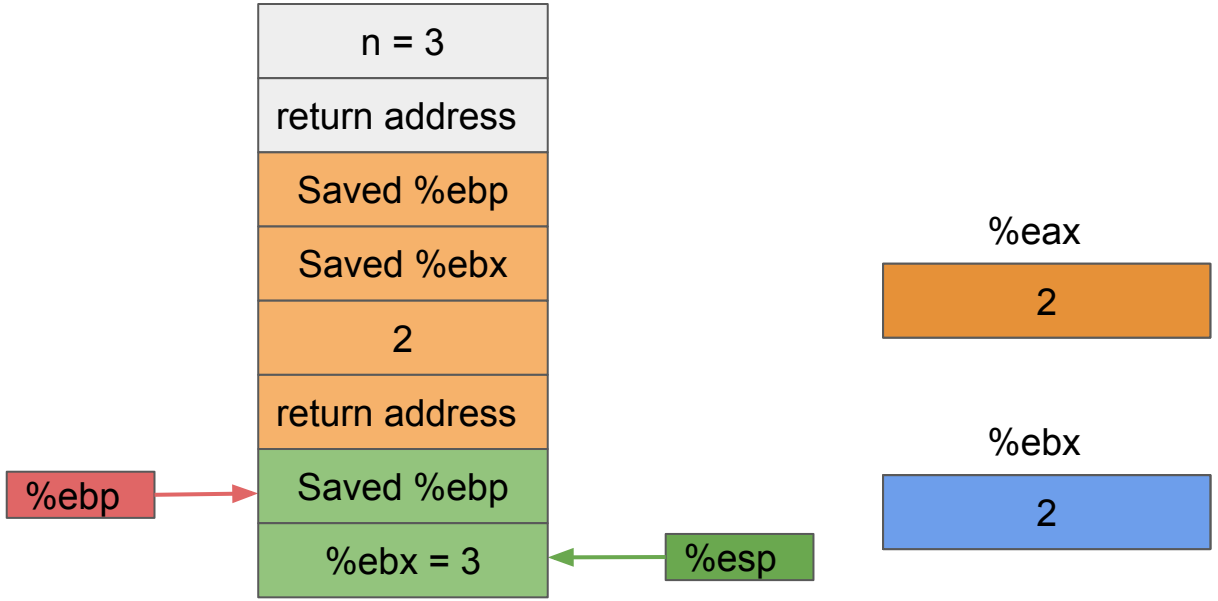
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



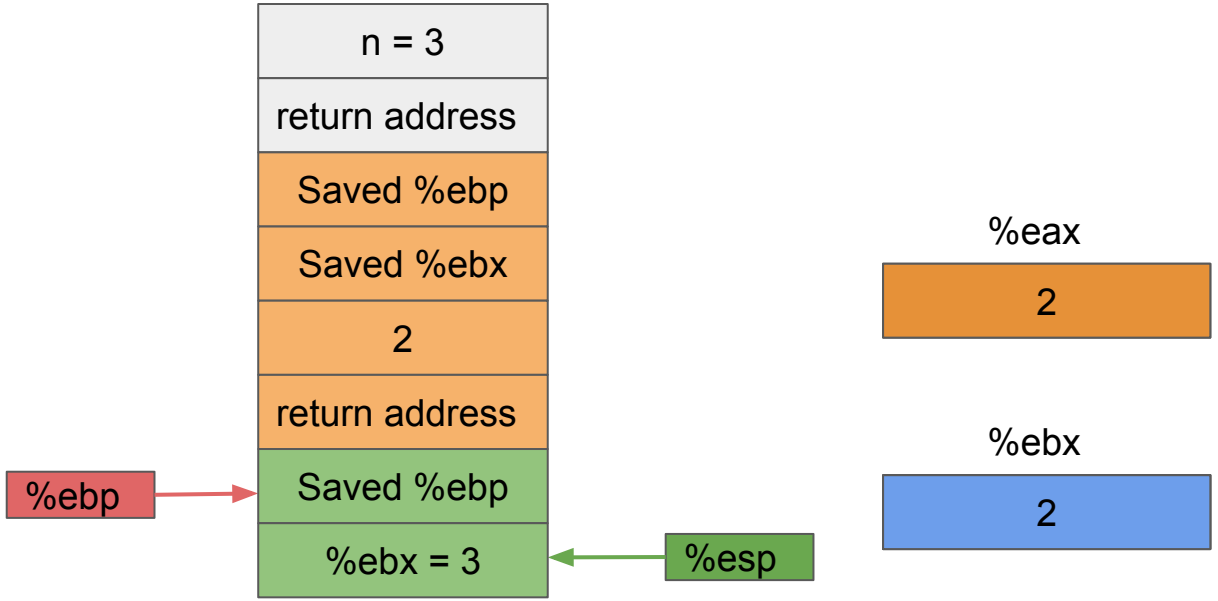
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



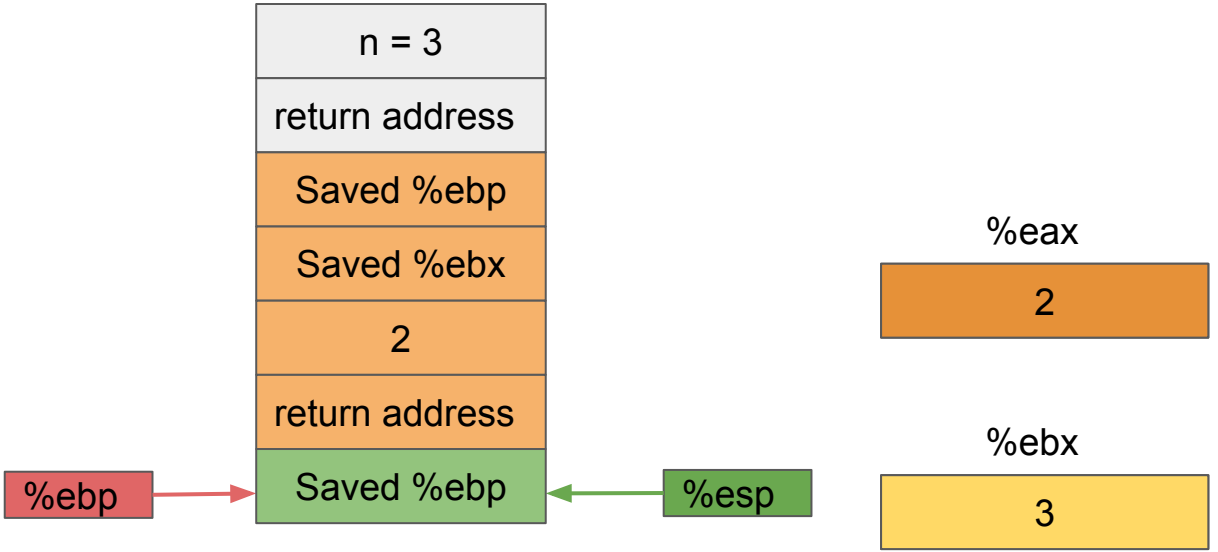
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



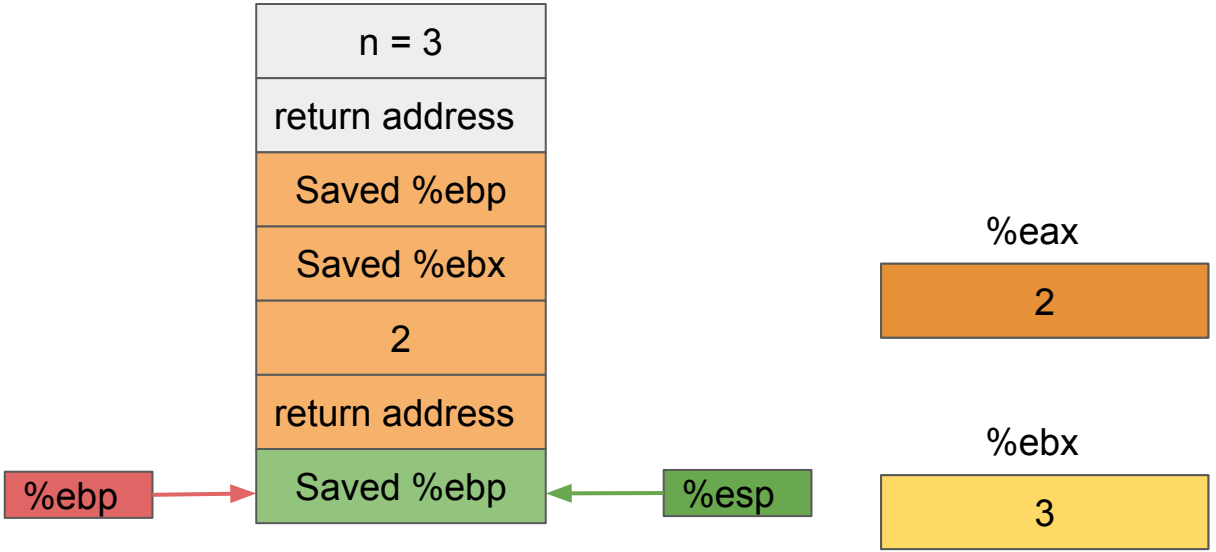
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



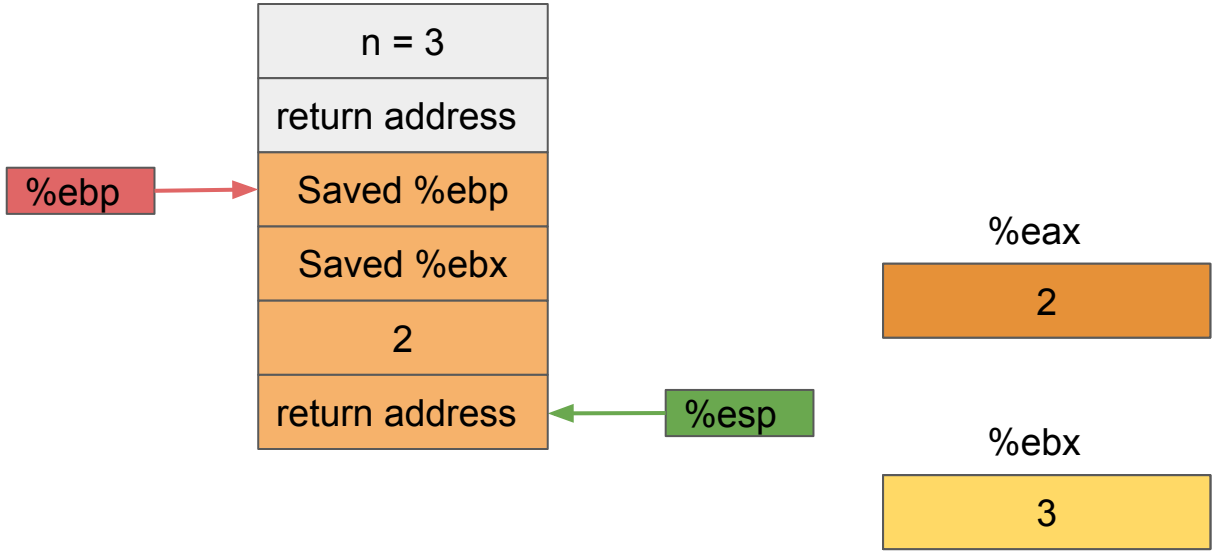
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



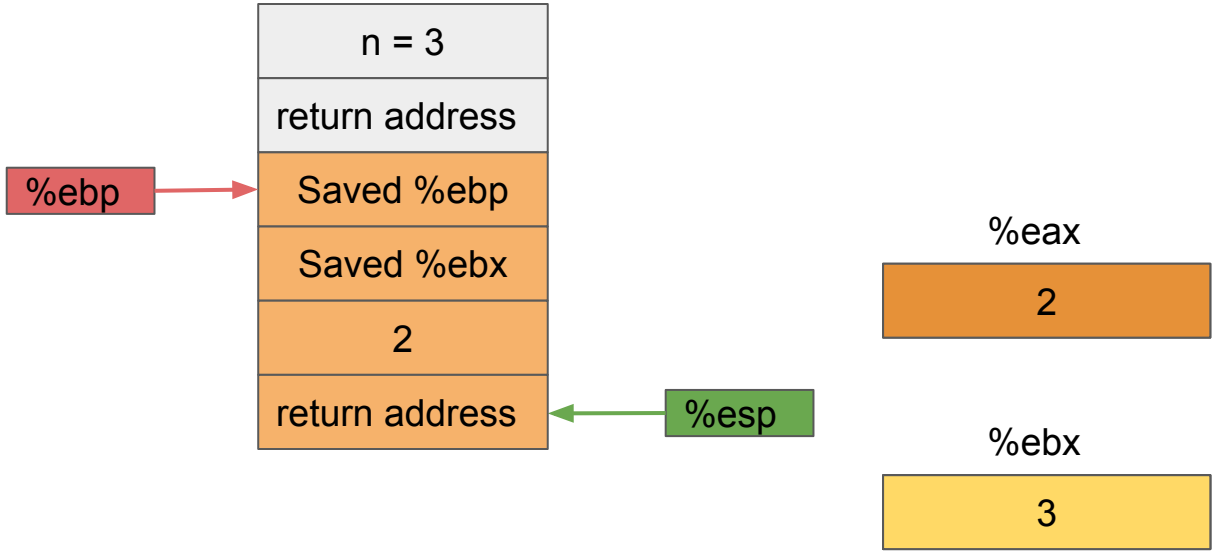
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



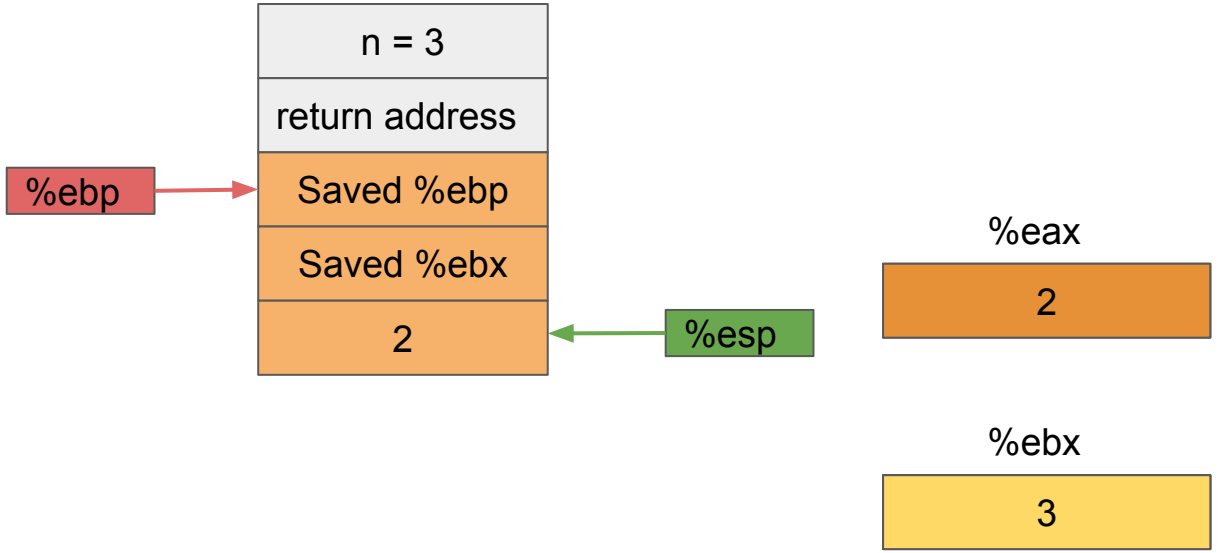
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



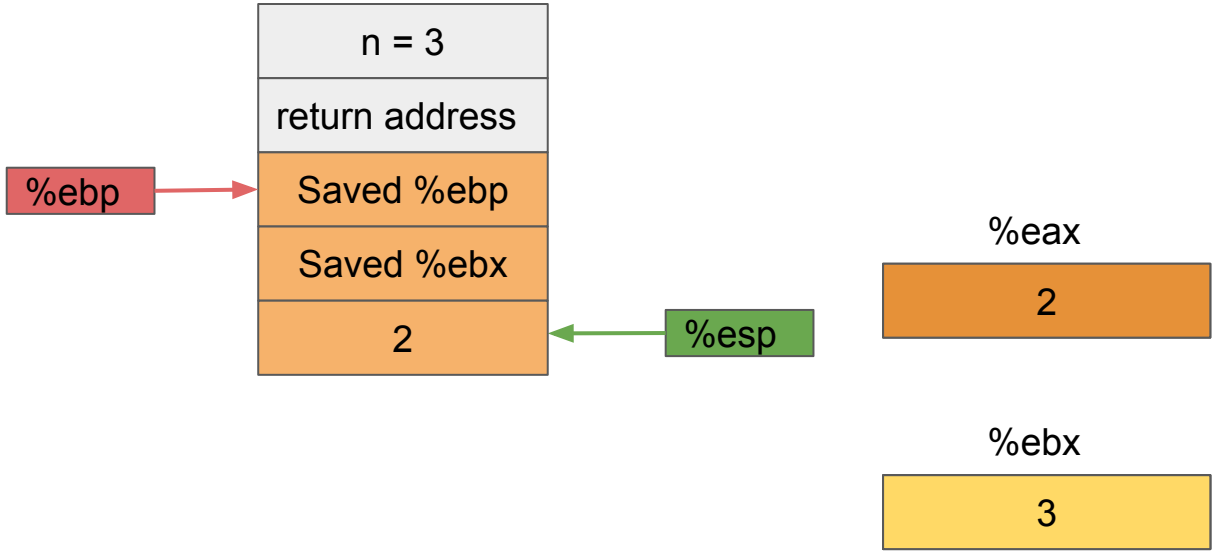
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



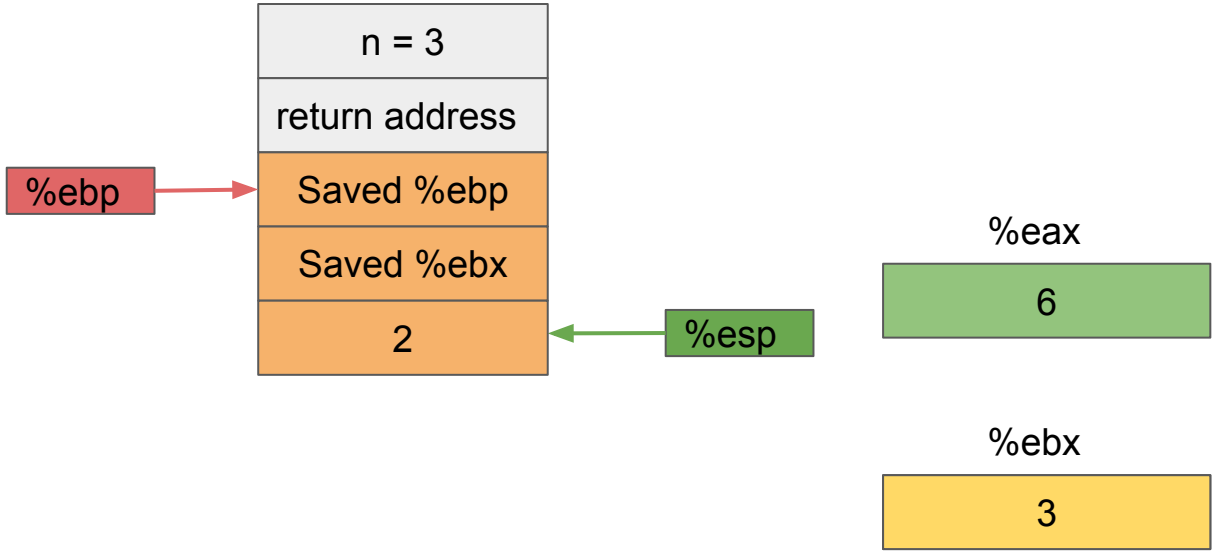
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



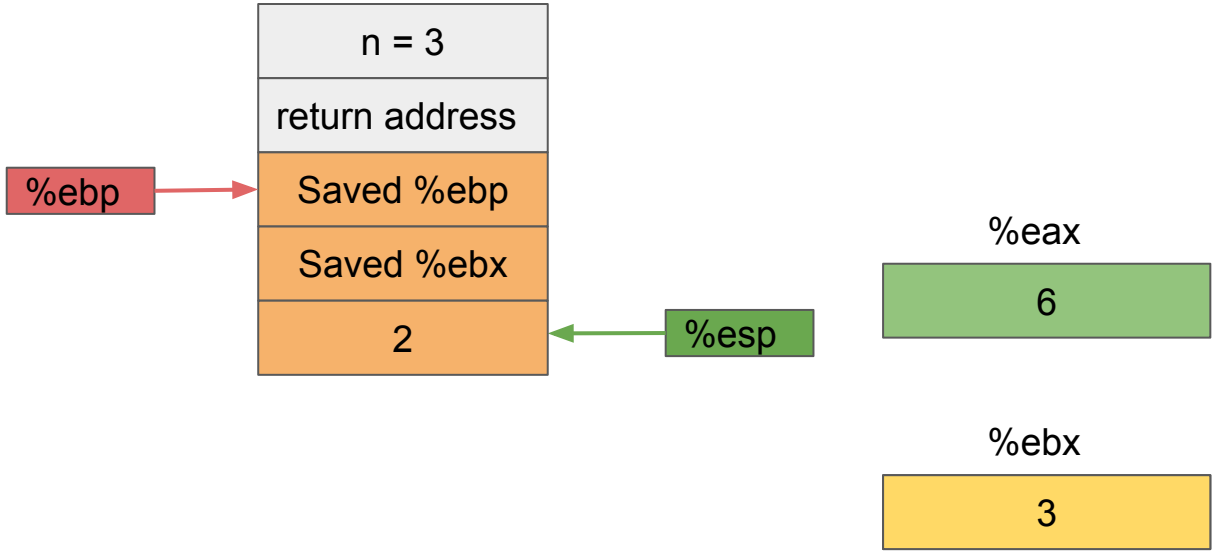
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



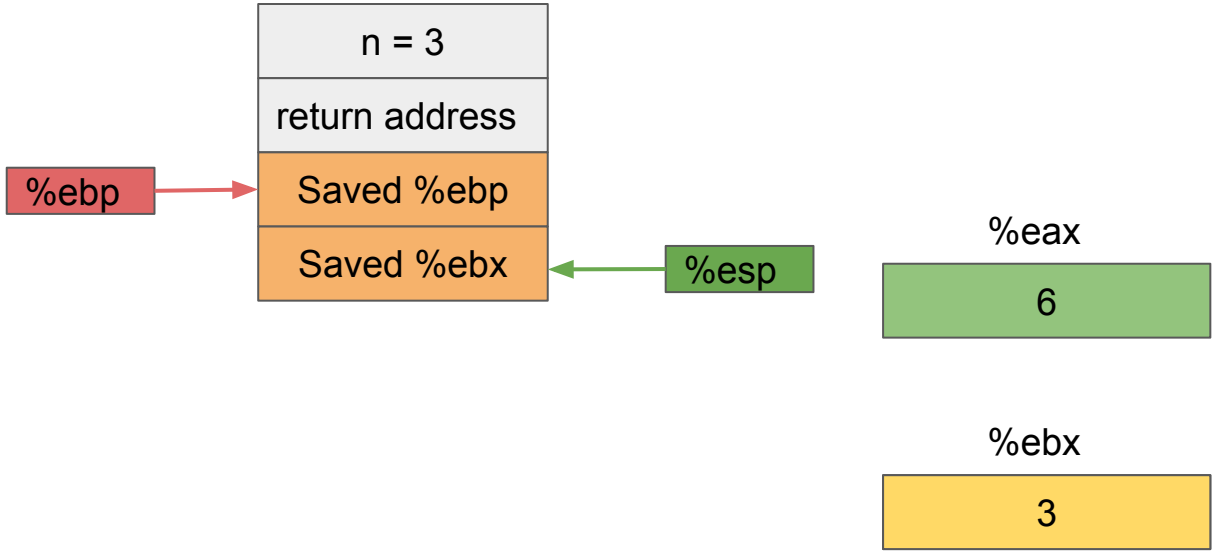
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



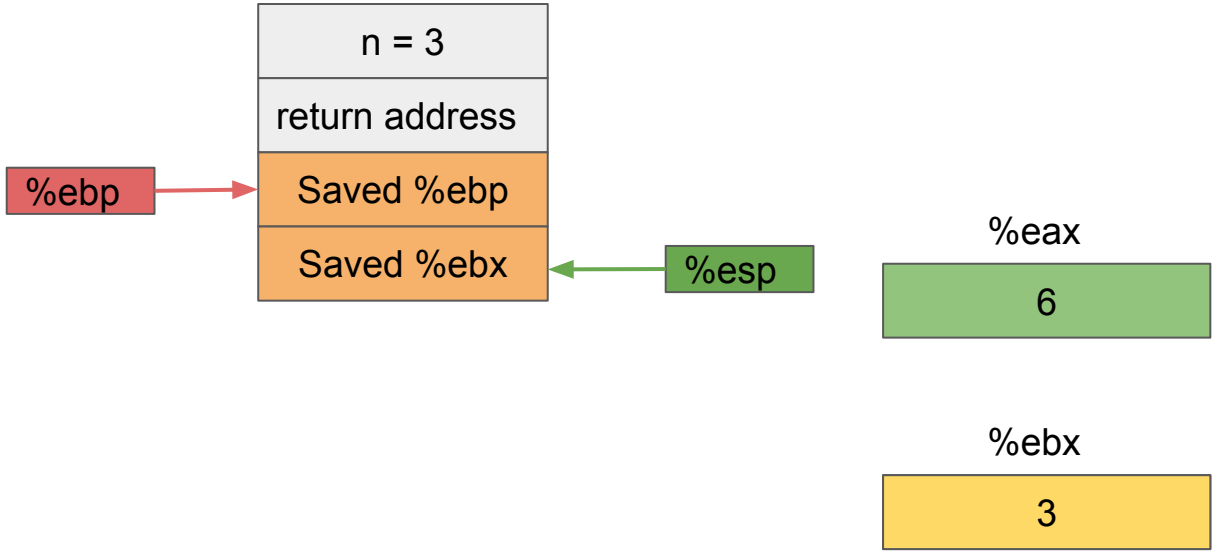
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



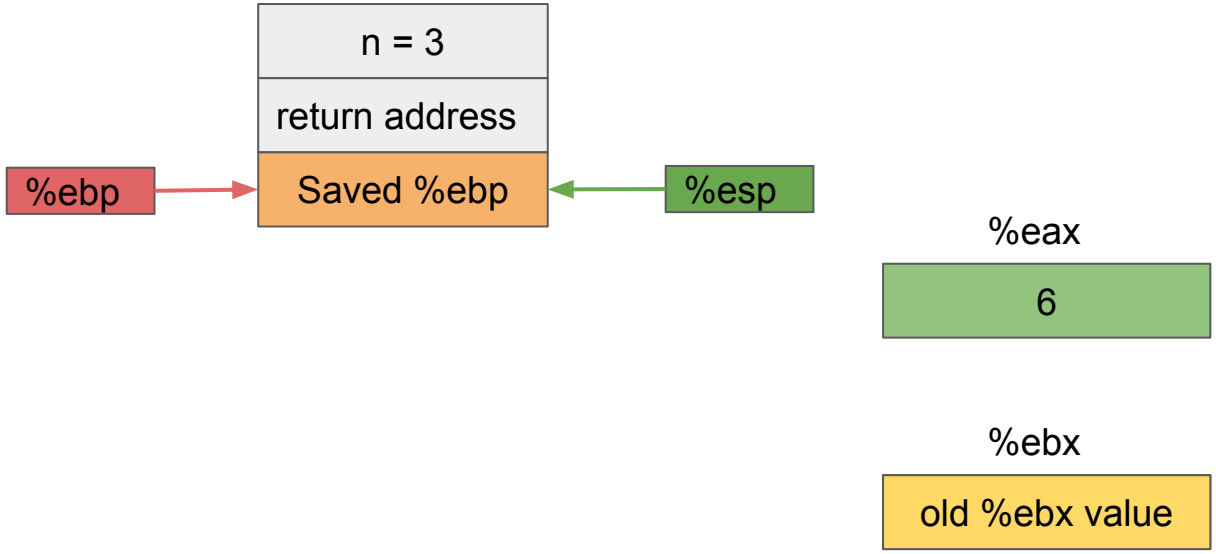
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



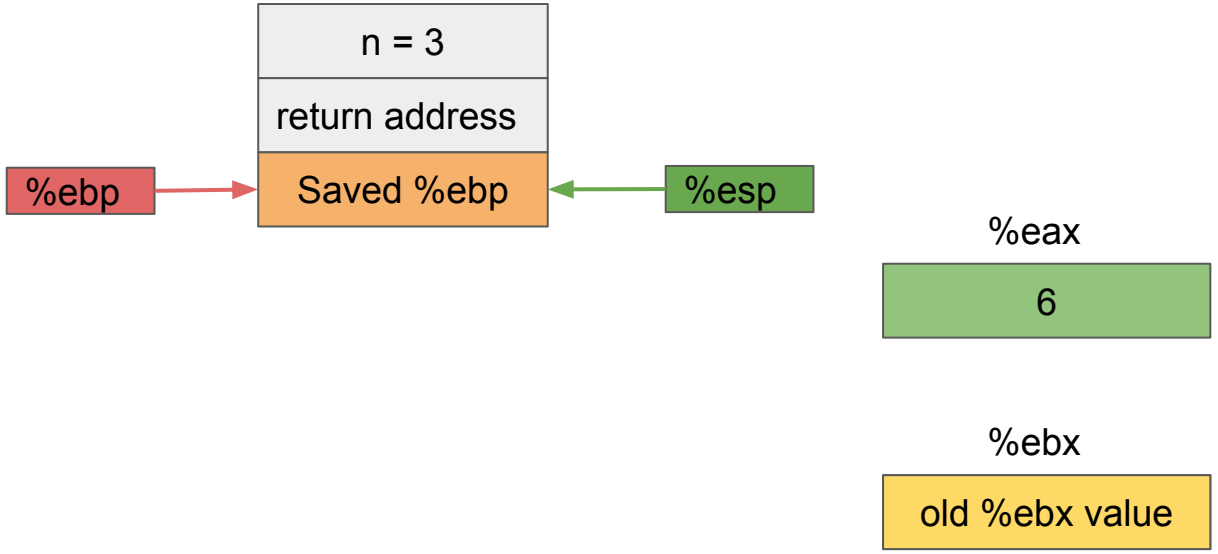
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



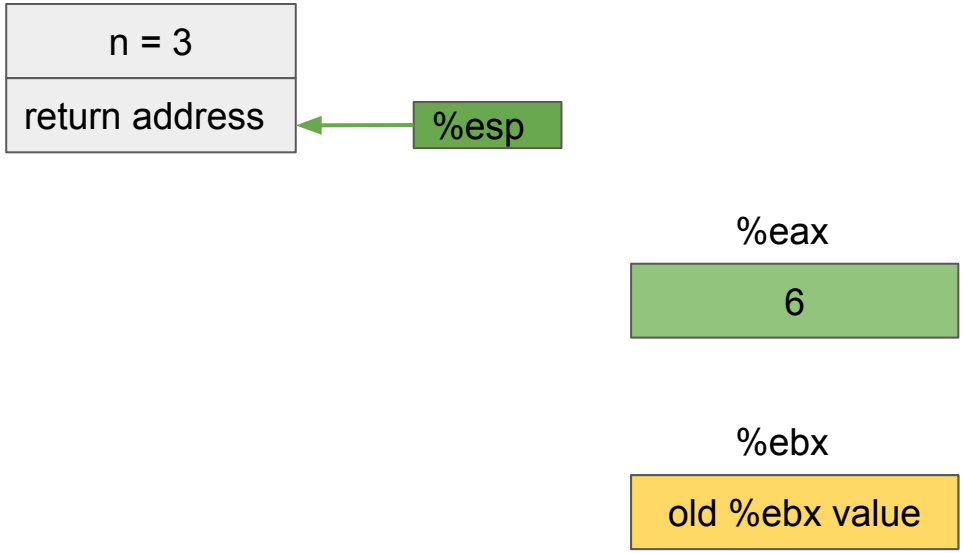
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



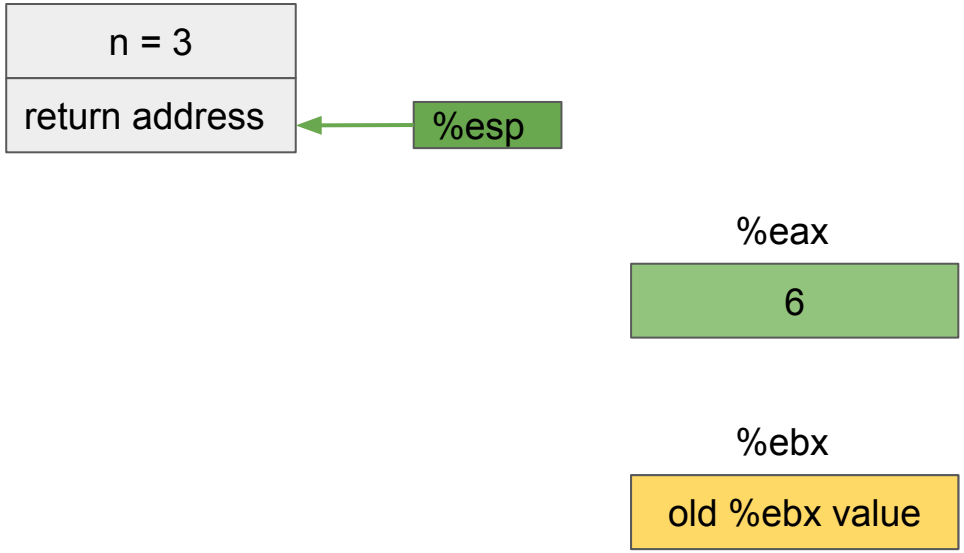
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



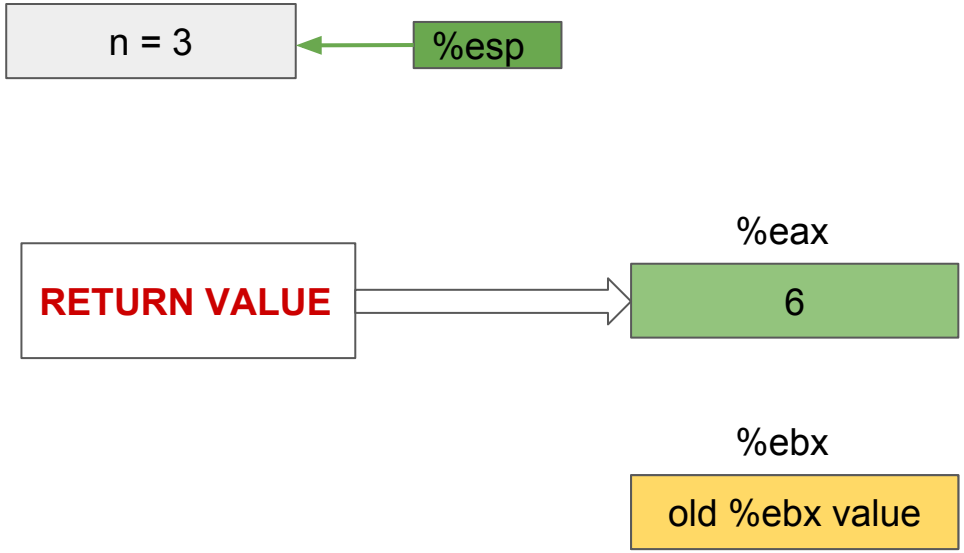
```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```



```
rfact:
    pushl %ebp
    movl %esp, %ebp
    pushl %ebx
    subl $4, %esp
    movl 8(%ebp), %ebx
    movl $1, %eax
    cmpl $1, %ebx
    jle .L53
    leal -1(%ebx), %eax
    movl %eax, (%esp)
    call rfact
    imull %ebx, %eax
.L53:
    addl $4, %esp
    popl %ebx
    popl %ebp
    ret
```