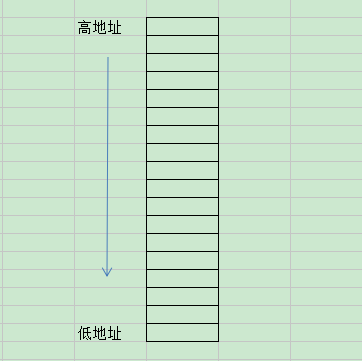
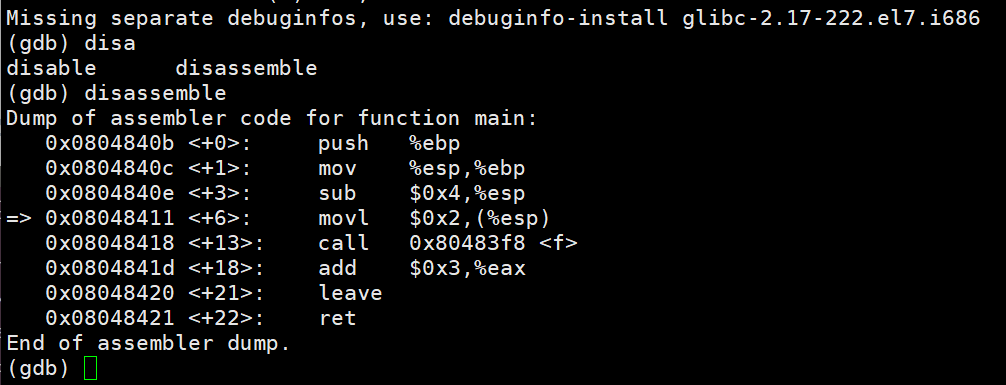
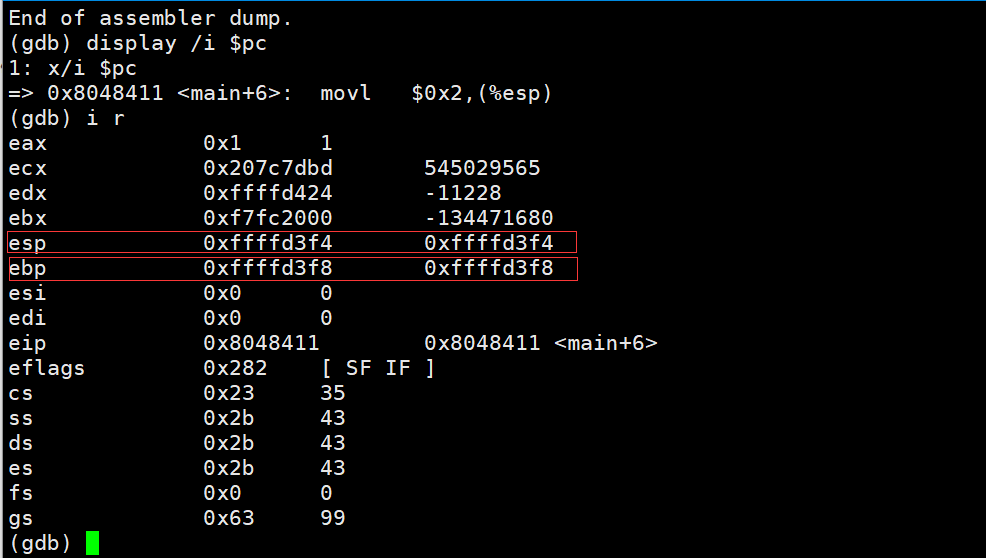


栈是从高地址向低地址延伸的。每个函数的每次调用，都有它自己独立的一个栈帧，这个栈帧中维持着所需要的各种信息。寄存器ebp指向当前的栈帧的底部（高地址），寄存器esp指向当前的栈帧的顶部（地址地）。

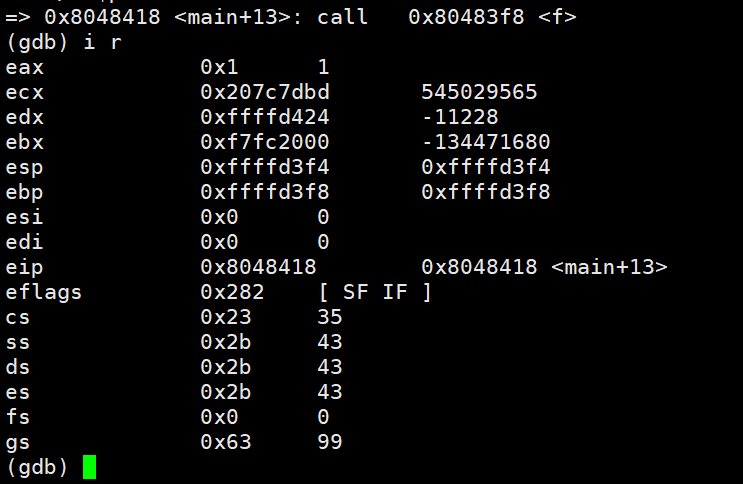


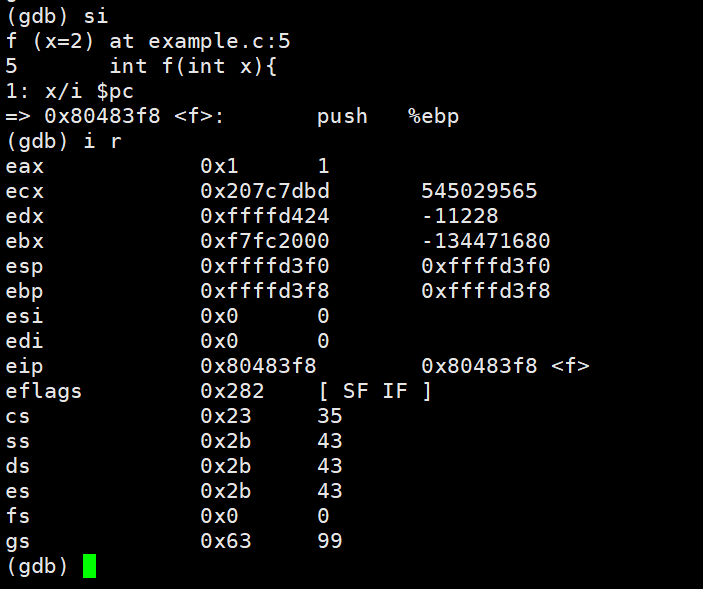


执行的第一步：

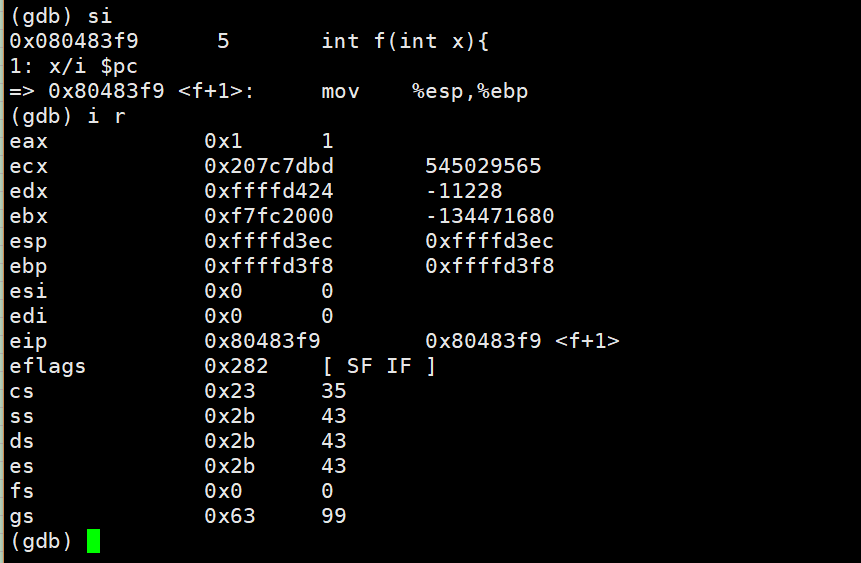


执行第二步：push 0x2

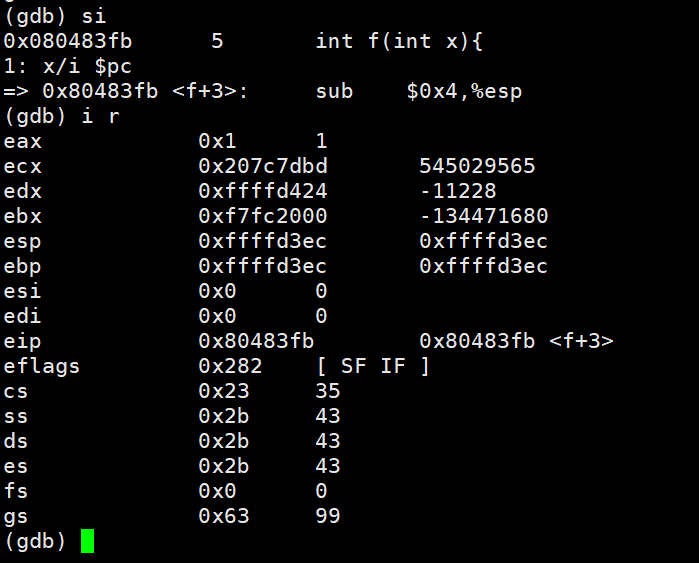


call将下一条指令的所在地址（即当时程序计数器PC的内容）入栈：

·push %ebp：

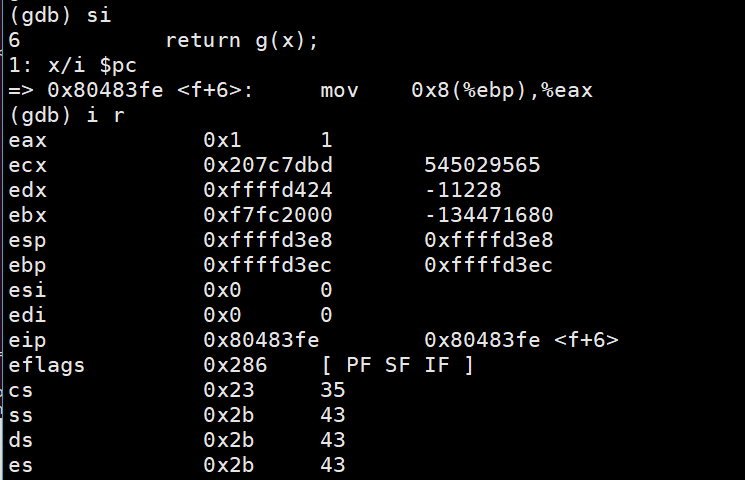


·mov %esp,%ebp：

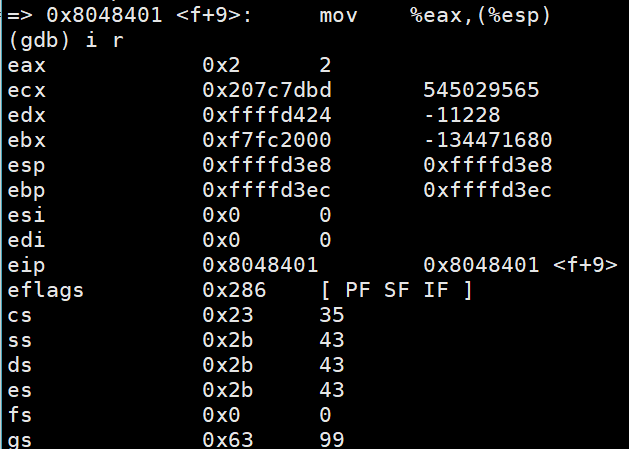


先为传参做准备：实参的计算在%eax中进行：

sub $0x4,%esp



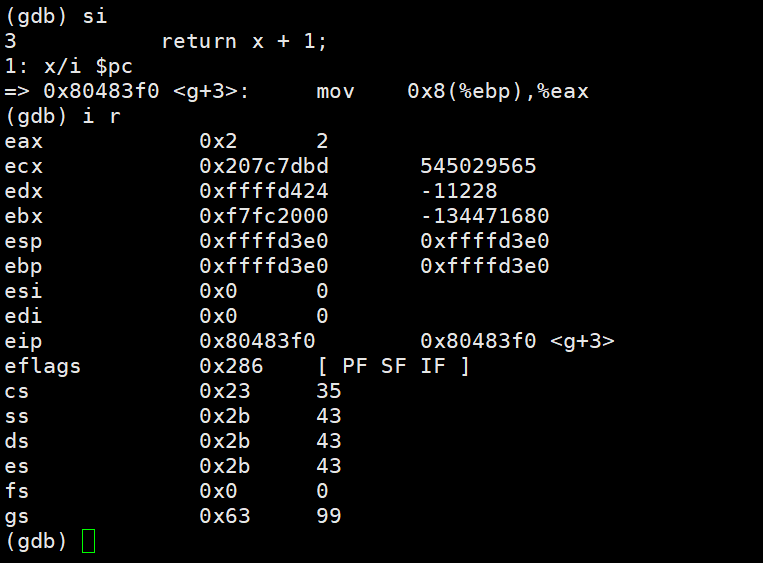
mov %eax,(%esp)



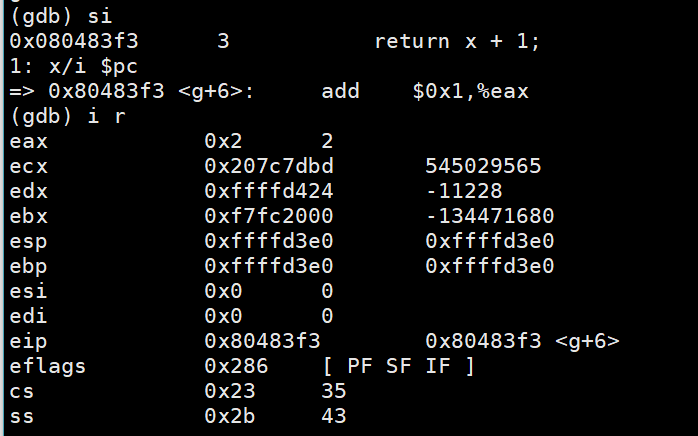
·call指令将下一条指令的地址入栈：

·push %ebp：

·mov %esp,%ebp：



add $0x1,%eax



·pop %ebp——将栈顶弹到%ebp中：

