

课堂问题

2013551 雷贺奥

问题一

if-else语句举例

```
int get_cont( int *p1, int *p2 )
{
    if ( p1 > p2 )
        return *p2;
    else
        return *p1;
}
```

p1和p2对应实参的存储地址分别为R[ebp]+8、R[ebp]+12，EBP指向当前栈帧底部，结果存放在EAX。

为何这里是“jbe”指令？

```
movl 8(%ebp), %eax    //R[edx] ← M[R[ebp]+8], 即 R[edx]=p1
movl 12(%ebp), %edx   //R[edx] ← M[R[ebp]+12], 即 R[edx]=p2
cmpl %edx, %eax       //比较 p1 和 p2, 即根据 p1-p2 结果置标志
jbe .L1               //若 p1 ≤ p2, 则转 L1 处执行
movl (%edx), %eax     //R[edx] ← M[R[edx]], 即 R[edx]=M[p2]
jmp .L2               //无条件跳转到 L2 执行

.L1:
movl (%eax), %eax     // R[edx] ← M[R[edx]], 即 R[edx]=M[p1]
.L2
```

75

jbe为小于等于0时JMP的指令。

前一条指令 `cmpl %edx,%eax` 将标志位设置为 $p_1 - p_2$ 的结果，

即当 $p_1 > p_2$ 时，进入if语句，jbe不跳转， $R[edx] = M[p_2]$ ，随后jmp 到.L2 退出此函数。

即当 $p_1 \leq p_2$ 时，进入else语句，jbe跳转至.L1， $R[edx] = M[p_1]$ ，随后执行到.L2，退出此函数

问题二

nemu 输出的 helloworld 和程序中输出的 helloworld 有什么区别？

- nemu中的程序是直接运行在裸机上,可以在 AM 的抽象下直接输出到设备(串口)；
- 而在程序设计课上写的 hello 程序位于操作系统之上,不能直接操作设备,只能通过操作系统提供的服务进行输出,输出的数据要经过很多层抽象才能到达设备层