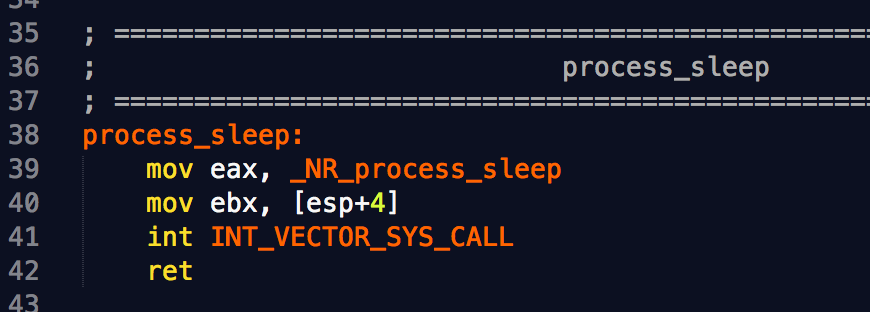
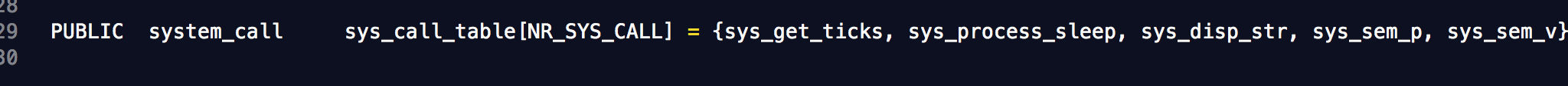
**第四次实验**

首先，根据要求添加系统调用sys\_process\_sleep，sys\_disp\_str，sys\_sem\_p，sys\_sem\_v四个系统调用。所以仿照系统调用sys\_get\_ticks，在syscall.asm文件中添加四个系统调用编号，并添加导出符号和调用代码（还有三个省略）。





和get\_ticks不同的是，新增的系统调用全部带参数，所以加上一句“mov ebx, [esp+4]”用来传递参数，然后在global.c中系统调用表中添加四项。同样在proto.h中也要添加申明（图略）。

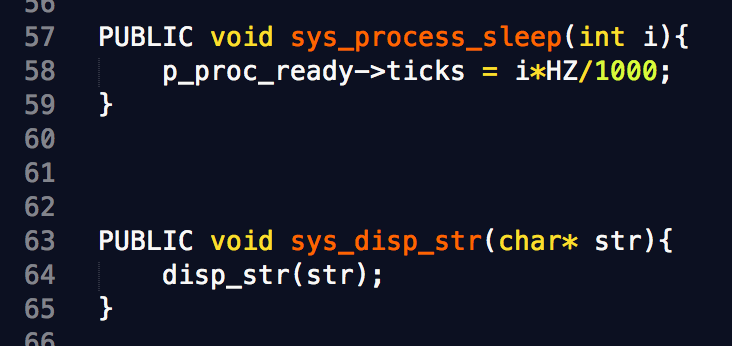


要处理进程调度问题，先在proc.h文件中的p\_proc结构中加上一个isWait标识变量。由于进程增多为五个，所以在proc.h中修改以下内容。



在main.c中增加方法TestD，TestE，供新增的进程使用（相应在proc.h中也增加申明）。

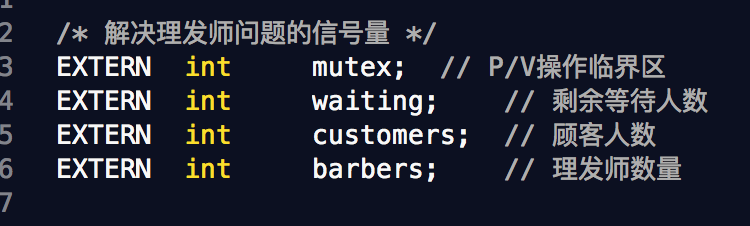
在proc.c中完成系统调用函数sys\_process\_sleep和sys\_disp\_str，其中分别给进程分配沉睡时间，以及调用内核函数disp\_str打印字符串。



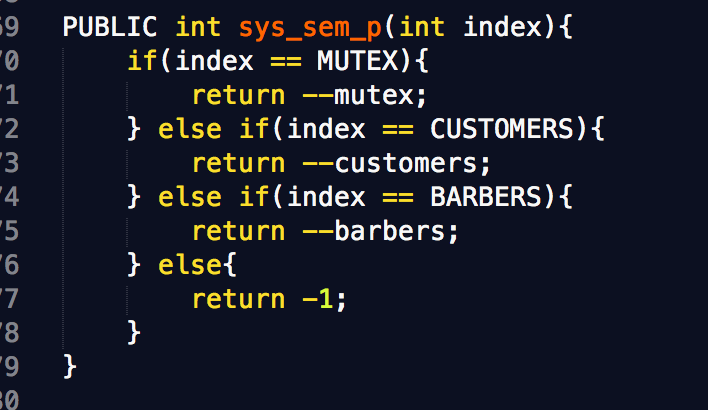
在const.h中添加几个信号量序号，可以用来统一进行多个信号量的PV操作。



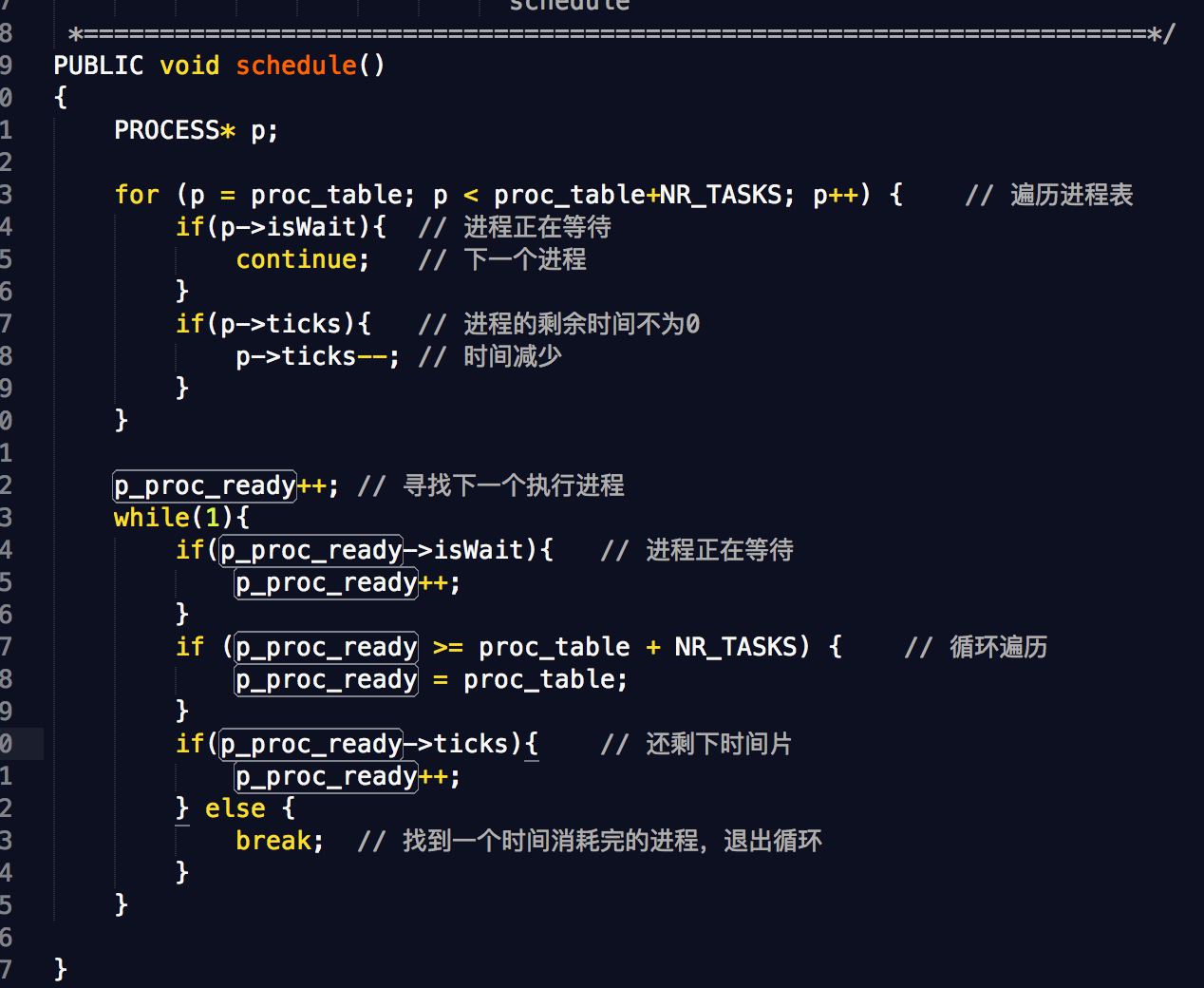
在global.h中增加四个信号量变量，作为处理理发师问题的PV操作中的信号量



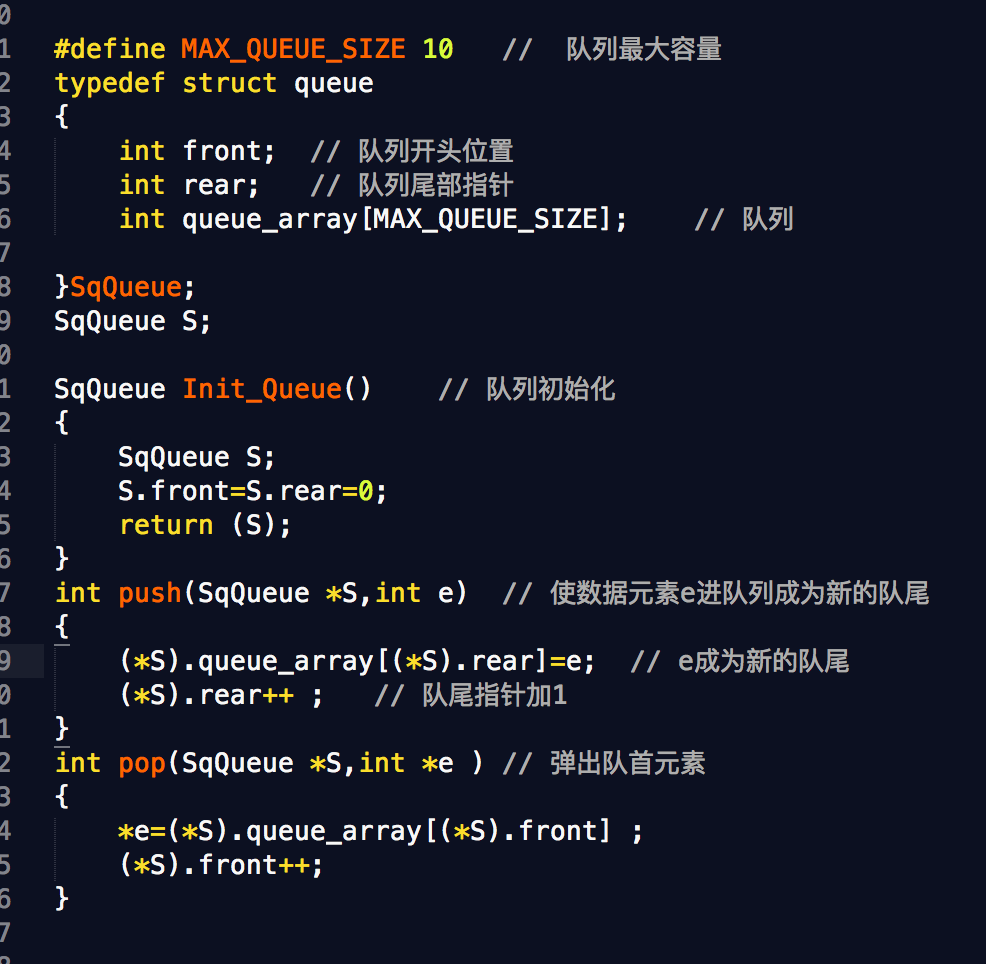
在系统调用中最简单的P/V，配合常量序号统一管理各变量信号量，供更具体的PV函数调用。



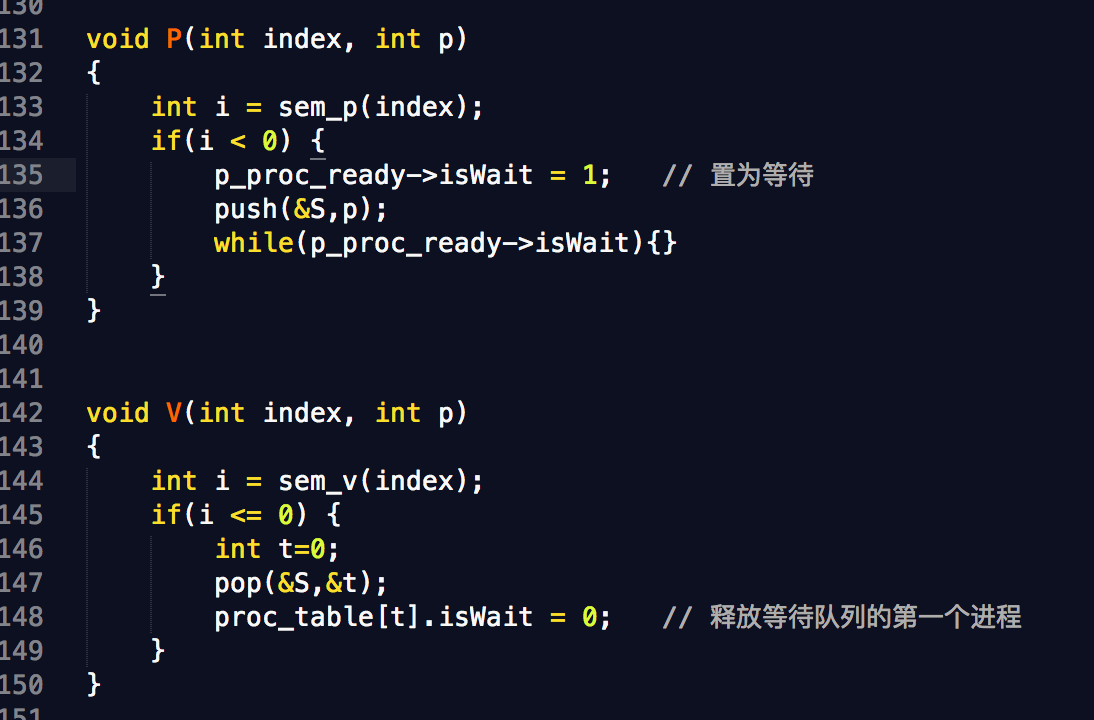
在proc.c中修改调度函数，使用时间片调度，而非优先级，如下：



接下来编写PV操作部分代码，首先需要定义一个等待队列，供在P操作中受阻的进程等待。还有初始化，入队和出队函数。



利用等待队列，编写实际使用的P，V函数



理发师进程

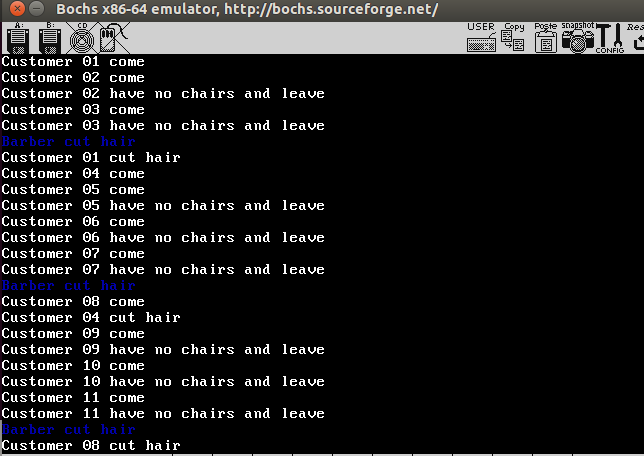


顾客进程

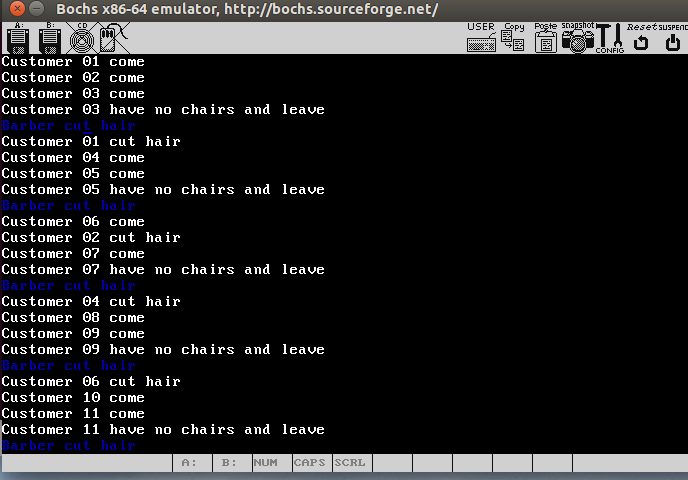


结果截图：

（1）一张椅子



（2）两张椅子



（3）三张椅子

