

《现代交换原理》实验报告

实验名称	摘挂机检测实验
班 级	2015211306
学 号	2015211301
姓 名	魏 晓
指导教师	丁 玉 荣

实验 2 摘挂机检测试验

一、实验目的

摘挂机检测试验用来考查对摘挂机检测原理的掌握情况

二、实验内容和实验步骤（简写）

用户在挂机状态时扫描输出为0,用户在摘机状态扫描输出是1.

那么:

- 摘机: `(!linestate200[i]) && linestate[i]`
- 挂机: `(!linestate[i]) && linestate200[i]`

还要保存当前状态到上一个 200 秒数组,然后判断是摘机还是挂机,还是不变.如果摘机或者挂机.那么改变当前的状态.upwx->

phonestate = ehandup,如果是挂机,同理.然后改变链表的状态.就可以完成摘挂机检测

```
1. LINEMAX : 最大线路数 ;
2. int linestate200[LINEMAX],linestate[LINEMAX] : 线路从 0 开始编号 ; 状态 : 1 : 有电
   流, 0 无电流 ;
3. enum UporOn {ehandup,ehandon} : 为摘挂机区别符: ehandup 表示摘机, ehandon 表示挂
   机 ;
4. struct UpOnnode //摘挂机队列节点结构
5. {
6.     UporOn phonestate; //摘挂机区别符 ;
7.     int linenum; //线路号 (从 0 开始) ;
8.     struct UpOnnode* next; //指向下一节点的指针 ;
9. };
```

三、源代码

```
1. #include "bconstant.h"
2. extern "C" _declspec(dllexport) void scanfor200(int linestate200[LINEMAX
], int linestate[LINEMAX], UpOnnode * head1, UpOnnode * end1)
3. {
4.     int up=0,down=0;
5.     for(int i = 0;i<LINEMAX;i++){
6.         struct UpOnnode * upwx = new struct UpOnnode;
7.         up = (!linestate200[i]) && linestate[i];
8.         down = (!linestate[i]) && linestate200[i];
9.         linestate200[i] = linestate[i];
10.        if(up || down){
11.            if(up)
12.                upwx->phonestate = ehandup;
13.            else
14.                upwx->phonestate = ehandon;
15.            upwx->linenum = i;
16.            upwx->next = 0;
17.            end1->next =upwx;
18.            end1 = upwx;
19.        }
20.    }
21.    return;
22. }
23. extern "C" _declspec(dllexport) void freenode(UpOnnode * node)
24. {
25.     delete node;
26. }
```

四、实验结果

在程序执行以后,能够实现以下功能:

1. 检测用户机当前的状态
2. 根据当前的状态生成相对应的摘挂队列节点

五、实验心得

心得:

1. 一定要在实验课前预习实验,对相应的实验原理有一个深度的认识,对于实验的目标和当前的条件有一个准确的判断
2. 注意实验老师在课前对实验的讲解,通常能够避开实验中大大小小的陷阱,成功完成实验
3. 对于现代交换原理的理解基础上,通常还要考虑具体的业务逻辑
4. 在写代码的过程中,首先应该写一个大致的框架,不需要具体到细节,然后再一步步细化,这样能更轻松地完成
5. 对与实验中的名称一定要有一个规律的命名,否则后期容易混淆

体会:

经过这次摘挂机检测实验,是我对通信过程中的具体过程有了一个更具体的理解,虽然代码写起来比较有难度.但只要读懂了实验原理,其实也是不难写出来的