

## PART II Operating System Lab Exam Answer

### 一、选择题（共40分，每题2分）

- 1.( A ) 2.( B ) 3.( C ) 4.( D ) 5.( A )  
6.( A ) 7.( B ) 8.( C ) 9.( D ) 10.( B )  
11.( A ) 12.( B ) 13.( C ) 14.( D ) 15.( C )  
16.( A ) 17.( B ) 18.( C ) 19.( D ) 20.( D )

### 二、判断题（共 20 分，每题 2 分）

- 1.( T ) 2.( F ) 3.( T ) 4.( T ) 5.( F )  
6.( F ) 7.( F ) 8.( F ) 9.( F ) 10.( F )

### 三、读Linux内核代码（20分）

(1) 寻找父目录，5分

(2) 对如下的共15处位置，作出解释，每处1分

```
static inline void follow_dotdot(struct nameidata *nd)
{
```

```
    while(1) {
        struct vfsmount *parent; struct dentry *dentry;
        read_lock(&current->fs->lock); 对进程的文件管理模块加“读锁”，以保证互斥
        if (nd->dentry == current->fs->root && 本目录就是进程的根节点，而且
            nd->mnt == current->fs->rootmnt){ 本文件系统也是根文件系统
            read_unlock(&current->fs->lock); break; 那么，没有父目录了，只能维持
                                                    当前位置；解开“读锁”，退出循环
        }
        read_unlock(&current->fs->lock);
        spin_lock(&dcache_lock);           对 dentry 加自旋锁
        if (nd->dentry != nd->mnt->mnt_root) { 当前目录不是某文件系统的根
            dentry = dget(nd->dentry->d_parent); 找到父目录
            spin_unlock(&dcache_lock);         解开 dentry 的自旋锁
            dput(nd->dentry);                  释放 nd 中的 dentry 数据结构
            nd->dentry = dentry;               更新 nd 中的 dentry 指针
            break;                            退出循环
        }
        parent=nd->mnt->mnt_parent;           找到当前文件系统的“父”文件系统
        if (parent == nd->mnt) {              如果“父”文件系统就是自己
            spin_unlock(&dcache_lock);
            break;                            只能维持当前位置，退出循环
        }
        mntget(parent);
        dentry=dget(nd->mnt->mnt_mountpoint); 当前文件系统在“父”文件系统的安装点
        spin_unlock(&dcache_lock);
```

```
        dput(nd->dentry);  
        nd->dentry = dentry;  
        mntput(nd->mnt);  
        nd->mnt = parent;  
    }  
}
```

#### 四、编程序，显示缺页中断次数（20分）

##### 1. 叙述缺页中断调用机理

缺页中断矢量设置（3分）

区分是页面异常（地址非法，访问权限等）还是缺页（3分）

Linux的页面管理，虚拟地址段管理，换页等（4分）

##### 2. 给出显示的方案（10分）