

# 操作系统实验

上机准备：

- 熟悉 Cygwin 环境
- 编译源程序使用 `gcc 源程序名 -o 目标文件名`（缺省为 `a.out`）。
- 学习 Linux（Unix）的常用命令。
- 学习 vi 编辑器的使用。
- C 语言语法

# 1 进程管理（20 分）

## 1.1 实验目的

加深对于进程并发执行概念的理解。实践并发进程的创建和控制方法。观察和体验进程的动态特性。进一步理解进程生命期期间创建、变换、撤销状态变换的过程。掌握进程控制的方法，了解父子进程间的控制和协作关系。练习 Linux 系统中进程创建与控制有关的系统调用的编程和调试技术。

## 1.2 实验说明

### 1) 与进程创建、执行有关的系统调用说明

进程可以通过系统调用 `fork()` 创建子进程并和其子进程并发执行。子进程初始的执行映像是父进程的一个复本。子进程可以通过 `exec()` 系统调用族装入一个新的执行程序。父进程可以使用 `wait()` 或 `waitpid()` 系统调用等待子进程的结束并负责收集和清理子进程的退出状态。

`fork()` 系统调用语法:

```
#include <unistd.h>
```

```
pid_t fork(void);
```

`fork` 成功创建子进程后将返回子进程的进程号,不成功会返回-1.

`exec` 系统调用有一组 6 个函数,其中示例实验中引用了 `execve` 系统调用语法:

```
#include <unistd.h>
```

```
int execve(const char *path, const char *argv[], const char * envp[]);
```

`path` 要装入的新的执行文件的绝对路径名字符串.

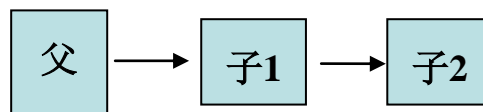
`argv[]` 要传递给新执行程序的完整的命令参数列表(可以为空).

`envp[]` 要传递给新执行程序的完整的环境变量参数列表(可以为空).

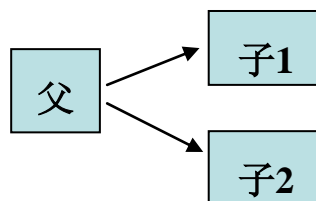
`Exec`执行成功后将用一个新的程序代替原进程，但进程号不变，它绝不会再返回到调用进程了。如果`exec`调用失败，它会返回-1。

`getpid()`---获取进程的pid

- 每个进程都执行自己独立的程序，打印自己的pid，每个父进程打印其子进程的pid;



- 每个进程都执行自己独立的程序，打印自己的pid，父进程打印其子进程的pid;



- 编写一个命令处理程序，能处理 `max(m,n)`，`min(m,n)` 和 `average(m,n,l)`这几个命令。（使用`exec`函数族）

## 2 处理器调度（20 分）

### 1.1 实验目的

熟悉使用各种单处理器调度的各种算法，加深对于处理机调度机制的理解。练习模拟算法的编程技巧，锻炼分析试验数据的能力。

### 1.2 实验说明

随机给出一个进程调度实例，如：

进程	到达时间	服务时间
A	0	3
B	2	6
C	4	4
D	6	5
E	8	2

模拟进程调度，给出按照算法先来先服务 FCFS、轮转 RR ( $q=1$ )、最短进程优先 SJF、最高响应比优先 HRN 进行调度各进程的完成时间、周转时间、带权周转时间。

## 3 存储管理上机作业（20 分）

### 3.1 实验目的

加深对于存储管理的了解，掌握虚拟存储器的实现原理；观察和了解重要的页面置换算法和置换过程。练习模拟算法的编程技巧，锻炼分析试验数据的能力。

### 3.2 实验说明

- 1.示例实验程序中模拟两种置换算法：LRU 算法和 FIFO 算法。
- 2.给定任意序列不同的页面引用序列和任意分配页面数目，显示两种算法的页置换过程。
- 3.能统计和报告不同置换算法情况下依次淘汰的页号、缺页次数（页错误数）和缺页率。

## 4 磁盘移臂调度算法实验（20 分）

### 4.1 实验目的

加深对于操作系统设备管理技术的了解，体验磁盘移臂调度算法的重要性；掌握几种重要的磁盘移臂调度算法，练习模拟算法的编程技巧，锻炼研究分析试验数据的能力。

### 4.2 实验说明

1. 示例实验程序中模拟两种磁盘移臂调度算法：SSTF 算法和 SCAN 算法
2. 能对两种算法给定任意序列不同的磁盘请求序列，显示响应磁盘请求的过程。
3. 能统计和报告不同算法情况下响应请求的顺序、移臂的总量。

## 5 文件管理作业（20 分）

### 5.1 实验目的

通过模拟文件的创建、删除操作，加深对于操作系统文件管理功能的了解，练习模拟算法的编程技巧，锻炼研究分析试验数据的能力。

### 5.2 实验说明

给出一个磁盘块序列：1、2、3、.....、500，初始状态所有块为空的，每块的大小为 2k。选择使用空闲表、空闲盘区链、位示图三种算法之一来管理空闲块。对于基于块的索引分配执行以下步骤：

- （1）随机生成 2k-10k 的文件 50 个，文件名为 1.txt、2.txt、.....、50.txt，按照上述算法存储到模拟磁盘中。
- （2）删除奇数.txt（1.txt、3.txt、.....、49.txt）文件
- （3）新创建 5 个文件（A.txt、B.txt、C.txt、D.txt、E.txt），大小为：7k、5k、2k、9k、3.5k，按照与（1）相同的算法存储到模拟磁盘中。
- （4）给出文件 A.txt、B.txt、C.txt、D.txt、E.txt 的盘块存储状态和所有空闲区块的状态。