验收日期： 2023年5月 10 日（星期 三 ） 时间：15:20

组长：倪玮昊

手机： 13965540430 Email：160622205@qq.com

|  |  |  |  |
| --- | --- | --- | --- |
| 贡献率/  组内成绩排序 | 学号 | 姓名 | 工作内容 |
| 14% | 2020211346 | 倪玮昊 | Kernel和shell开发和操作指令设计,GCC模块,日志系统,全部代码整合拼接 |
| 14% | 2020211345 | 缪奇志 | 进程管理模块,中断系统 |
| 14% | 2020211347 | 钱锡锐 | 内存模块 |
| 14.5% | 2020211349 | 施天勤 | 文件系统,磁盘调度 |
| 14.5% | 2020211350 | 石亚行 | 文件系统,编辑器系统 |
| 14% | 2020211 | 葛北鱼 | 设备管理模块,PCB设计 |
| 15% | 2020211 | 何正豪 | UI设计与实现,包括shell,设备运行状态,文件框,内存占用情况 |
| 提交  资料  清单 | 1、课程设计报告（分组表） 2、课程设计报告  3、源程序 4、可运行程序 5、测试用例及报告  6、软件使用说明 | | |

软件功能

|  |  |
| --- | --- |
| 进程控制 | 进程状态为ready, running, waiting, terminated，进程创建时首先向内存申请，成功后创建新进程，进程状态设置为ready加入到ready队列，同时加入到pcblist当中。当进程被调度会转为running状态，当需要使用设备时转为waiting，加入waiting队列。当一个进程执行完毕时转为terminated状态，并释放相关的资源。可以使用kill指令结束进程释放资源。 |
| 进程调度 | 采用时间片轮转调度的算法，当时间片使用完之后调度进程，按照优先级，优先从优先级最高的ready队列里找到进程，把之前正在运行的状态设置为ready加入ready队列，把选中的进程置为currentrunning，设置进程状态为running。每执行一个指令程序计数器加一。 |
| 内存管理 | 采用两类算法对对内存的分配及释放，即页式分配与连续分配，采用连续分配时，实现了 FF 算法。采用页式分配时，实现了虚拟内存机制并采用“Demand Paging”算法，页面置换算法可以选择LRU以及FIFO。整个类也会统计缺页率等信息 |
| 中断 | 当进程需要申请使用设备资源时产生中断，首先把进程转为waiting，加入到waiting队列，申请设备分配资源，分配成功后设备加载进程信息，当使用设备完毕后判断进程是否结束，后续还有指令则置为ready加入ready队列，否则置为terminated，释放资源。 |
| 软件界面 | 桌面显示 时间显示、窗口拖动和缩小  设备管理器窗口 :设备使用状态监控  文件编辑器窗口：文件的编辑和保存  文件管理器窗口：文件以及文件夹的显示、增删改和运行  任务管理器窗口：进程监控以及控制终止进程  Shell窗口：发送和接收指令 |
| 文件系统 | 构建文件树 ls，列出目录下文件和文件夹 cd，改变工作目录 mkdir，创建文件夹 mkf，创建文件 rm，删除文件 vi，编辑文件 三个策略进行磁盘分配 整理磁盘碎片 展示磁盘块状态 根据请求，计算在不同策略下，磁头移动的距离 |
| 设备管理 | 设备管理：实现设备安装时间、使用时间、空闲时间记录，设备与当前使用设备进程号的绑定，使用设备相关操作，当前设备信息统计（总设备、设备种类、空闲设备、设备使用情况、设备使用时间、本次设备使用时间等），设备添加与删除，设备状态更新，返回刚刚结束任务的设备（即刚刚由使用状态进入空闲状态的设备）等功能 |
| 模块间协调 | Kernel唤醒controller模块控制UI,UI和shell通过管道通信,给kernel传输数据(即指令),指令对各个模块进行操作  文件系统负责有关文件路径和文件内容的信息,把数据传给进程模块执行文件内容,文件内容中的内容会通过文件系统进行磁头寻道,对内存进行访问,对硬件模块进行调度硬件和进行中断,或占用cpu.还可以通过fork指令创建子进程 |