



| 日期 | 完成内容 |
|--------|--|
| 12月23日 | <p>从敏捷开发、代码易读性、开发复杂程度上考虑，决定使用 C 语言的结构体，而非 c++ 的类。</p> <ol style="list-style-type: none"> 1) 定义 PCB 结构体、PCB 队列、内存以及各种变量。 2) 编写多进程并行代码，预留 虚拟存储器管理即进程调度算法 接口, 同时编写进程 PV 原语。 <ul style="list-style-type: none"> -通过设定模拟计时器，来实现多进程并行 3) 模拟计算机分配内存的管理程序 <p>设置变量 <code>memory_allocation_method</code> 选择两种分配方式：固定分区方式、可变分区方式</p> <ul style="list-style-type: none"> -固定分区方式选用 分区大小相等方式。 -判断内存不够：外部碎片？进程太多？ -内存分配方式采用链表思想：节省检索时间、提高分配效率 <p>如 <code>m[0-4096]</code> 均分配给操作系统，只需定义 <code>m[0]=4095</code>，<code>m[0+1]=操作系统进程 PID</code>，<code>m[m[0]]=-1</code></p> |
| 12月24日 | <ol style="list-style-type: none"> 1) 测试代码：分别 进程管理模块 和 存储器管理模块，以及接口测试。 2) 测试发现内存分配有 bug，即分配空间存在-1 的情况，复刻出现错误的操作，重点针对 <code>memAllocation(PCB* p)</code>、<code>Control()</code> <i>//处理器调度程序</i> 进行调试 3) 开始设计开发第三个模块虚拟存储器管理，目光聚焦于 <code>memClear(PCB* p)</code>，变量 <code>pageFaults</code>，<code>memAllocation(PCB* p)</code>。 <ul style="list-style-type: none"> -缺页数 关注 进程分配内存函数，因为它分配多少页，等于它缺多少页 -淘汰页面号 关注 memClear 函数 -<code>pageFaults</code> 记录缺页总次数 4) 在 <code>main</code> 函数上预留文件管理模块接口 |

| | |
|--------|--|
| 12月25日 | <div data-bbox="391 206 1356 694"><p>1) 设计文件管理模块</p><p>-创建以下全局变量和函数</p><pre>// 磁盘调度算法 extern int Track_alo;// 磁盘选择算法的标记: 1 为 T_FCFS, 2 为 T_SSTF, 3 为 T_SCAN extern int serviceOrder[MAX_TRACKS];// 存储磁盘调度算法中的服务顺序 extern int totalTracks;// 记录当前 serviceOrder 数组中已经存储的磁道数量 const char* readFile();// 读取文件 void T_FCFS(int requests[], int numRequests); void T_SSTF(int requests[], int numRequests, int currentTrack); void T_SCAN(int requests[], int numRequests, int currentTrack, int direction); void fileManage();</pre></div> <div data-bbox="391 723 1356 884"><p>2) 集成各个模块, 进行测试</p><p>经测试发现, scan 选择 0 方式时, 输出故障, 需要针对 scan 算法进行修改, 经调试发现, 是 scan 算法的跳出循环条件有问题。</p></div> <div data-bbox="391 896 1259 1227"><pre>*生产者消费者模拟 正在从磁盘文件中读取预生产的字符... ***** *请选择磁盘调度算法: * 1: 先来先服务(T_FCF) * 2: 最短寻道优先(T_SSTF) * 3: 电梯算法(T_SCAN) ***** 3 ***** *按SCAN磁盘调度算法重新排序! ***** *选择移动方向: *0: 低轨道 *1: 高轨道 ***** 0 *SCAN Service Order: *Total Movement: 0 *****</pre></div> <div data-bbox="391 1240 1353 1630"><pre>正在从磁盘文件中读取预生产的字符... ***** *请选择磁盘调度算法: * 1: 先来先服务(T_FCF) * 2: 最短寻道优先(T_SSTF) * 3: 电梯算法(T_SCAN) ***** 3 ***** *按SCAN磁盘调度算法重新排序! ***** *选择移动方向: *0: 低轨道 *1: 高轨道 ***** 0 *当前从第0条磁道开始移动 *原磁道请求服务顺序: 10 23 11 15 4 16 9 5 2 19 *SCAN Service Order: 2 4 5 9 10 11 15 16 19 23 *Total Movement: 23 *****</pre></div> <div data-bbox="391 1659 1356 2004"><p>3) 进行系统测试, 优化代码, 提高代码易读性, 提高代码复用性、运行效率</p><p>-将 绝对路径 改成 相对路径</p><p>-替换冗余变量等</p><p>-重构函数, 提高代码复用性</p><p>-优化进程调度算法、处理器处理算法</p></div> |
|--------|--|

| | |
|--------------|--|
| 12 月 26 日 | |
|--------------|--|

| 日期 | 完成内容 |
|-----------|------|
| 12 月 27 日 | |
| 12 月 30 日 | |
| 12 月 31 日 | |
| 12 月 2 日 | |