南京信息工程大学 操作系统实验报告

实验项目 进程调度算法的实现

|  |  |  |  |
| --- | --- | --- | --- |
| 专业 计算机科学与技术 | 年级 22 | 学号 202283290159 | 姓名 张瑞晨 |

1. 主要实验步骤

|  |
| --- |
| . 实验背景  进程调度是操作系统核心功能之一，通过合理分配 CPU 资源，确保多任务系统的高效运行。本实验实现了一个基于优先级调度的进程管理系统。利用 Java 的 PriorityQueue 实现进程队列，模拟进程的创建、调度和状态转换。  2. 代码设计  本实验代码由以下主要类组成：  PCB 类：定义进程控制块的属性及操作。  PCBRun 类：实现进程的调度与状态转换。  ProcessSchedulingTest 类：测试和运行进程调度。  (1) PCB 类  PCB 类定义了进程的基本属性，包括：  name：进程名  duringTime：进程所需时间  priority：进程优先级（数值越小优先级越高）  state：进程状态（Ready、Running、Finished）  该类提供了 toString() 方法用于打印进程信息。  (2) PCBRun 类  PCBRun 类实现了 Runnable 接口，包含核心的调度逻辑：  使用 PriorityQueue 存储进程队列。  通过 while 循环不断从队列中取出优先级最高的进程进行调度。  每次调度后更新进程的状态、优先级和剩余时间。  (3) ProcessSchedulingTest 类  ProcessSchedulingTest 类是主程序入口，执行以下操作：  创建一个 PriorityQueue，按照优先级从高到低排序。  创建多个 PCB 对象并加入队列。  启动两个线程，一个用于生产进程，另一个用于消费进程。  (4) 调度逻辑  调度逻辑的核心是优先级调度算法：  进程按照优先级从高到低进行调度。  调度时更新进程的剩余时间和状态。  若进程的剩余时间为 0，则状态设置为 Finished 并移出队列。 |

1. 实验结果

|  |
| --- |
| P1 开始排队...  P2 开始排队...  现在时间是: 1673340000000  进程名 | 剩余时间 | 优先级 | 状态  P1 | 2 | 1 | Ready  P2 | 3 | 10 | Ready  P1 已完成！  P2 已完成！  2. 结果分析  从输出结果可以看出：  进程按照优先级调度，优先级数值小的进程优先运行。  进程在调度过程中，状态从 Ready 变为 Running，最后变为 Finished。  进程剩余时间逐渐减少，直至完成任务。 |

1. 实验遇到问题及解决

|  |
| --- |
| 问题 1：进程优先级更新逻辑错误  在初始实现中，进程的优先级没有正确更新，导致调度不符合预期。  解决方法：  修改进程优先级更新逻辑，确保每次调度后优先级按照预期调整：  2. 问题 2：队列中的进程状态未及时更新  在调度过程中，未及时更新队列中其他进程的状态，导致输出结果不准确。  解决方法：  在每次调度时，遍历队列中的所有进程并更新状态： |

1. 总结

|  |
| --- |
| 本实验通过 Java 实现了一个简单的进程调度模拟系统，主要采用优先级调度算法。通过实验，我们深入理解了进程调度的基本原理和 Java 的线程管理机制。在调度过程中，我们遇到了进程优先级更新、队列状态更新等问题，并通过修改代码逐一解决。  通过本次实验，我们掌握了以下知识：  进程控制块的基本结构及其重要性。  优先级调度算法的实现方法。  使用 PriorityQueue 实现调度队列的技巧。  Java 多线程的基本用法及异常处理方法。  未来的改进方向：  支持更多类型的调度算法（如时间片轮转、最短作业优先等）。  增加进程的 I/O 操作模拟，提升调度系统的逼真度。  实现更完善的进程状态管理和调度日志记录。  本实验有效地提升了我们的编程能力和对操作系统调度机制的理解，为后续学习奠定了良好的基础。 |