

**实 验（实训）报 告**

**项 目 名 称**  （实验名称）

**所属课程名称**  操作系统

**项 目 类 型**  进程管理

**实验(实训)日期**  2024.11.6

**班 级**  软件1班

**学 号**  220110900843

**姓 名**  周弋航

**指导教师**  陈伟锋

浙江财经大学教务处制

|  |
| --- |
| **一、实验（实训）概述：** |
| **【目的及要求】**  1.在ubuntu上使用vim+gcc进行简单编程，了解进程  2.进程调度模拟，用c语言实现对n个进程采用动态优先权算法的调度  **【基本原理】**  1.vim操作：  创建文件：touch 文件名  进入编辑：vi文件名  按esc退出编辑  保存文件输入：wq  2.使用gcc编译执行  编译：gcc 文件名  执行：./a.out  3.动态优先权算法的调度原理：若数值越大优先权越高，每运行一个时间单位优先权-n，若数值越小优先权越高，每运行一个时间单位优先权+n  【实施环境】  windows 11；VMmare;linux;Dev-c++; |
| **二、实验（实训）内容：** |
| 安装gcc与使用gcc 编译文件  Vim hello.c  Gcc hello.c -o hello  ./hello  执行hello.c    执行fork.c    编写程序模拟动态优先权算法的调度        1 首先定义了Process结构体来表示一个进程，包含进程 ID（pid）、优先权（priority）和执行时间（burst\_time）等属性。同时重载了<<运算符以便于输出进程信息。  2 comparePriority函数是一个比较函数，用于根据进程的优先权对进程进行排序，使得优先权高的进程排在前面。  3 dynamicPriorityScheduling函数实现了动态优先权调度算法的核心逻辑：  它将传入的进程向量复制到就绪队列中，并在循环中不断处理队列中的进程。  每次从就绪队列中取出优先权最高的进程进行执行。  根据时间片（time\_quantum）和进程的剩余执行时间来决定是完成该进程还是将其重新放回就绪队列（抢占）。  在每个时间片结束后，通过生成一个随机数来简单地随机调整当前执行进程的优先权值。  4 在main函数部分：  首先设置了随机数生成器的种子，以便每次运行程序时能生成不同的随机数序列。  创建了指定数量的随机进程，并将它们添加到进程向量中。  输出初始进程信息，然后调用dynamicPriorityScheduling函数进行调度模拟，最后输出最终的执行顺序。  **【结论与讨论】（结果、分析）** |
|  |
| **评语：**  **成绩： 指导教师签名：**  **批阅日期：** |