

**实 验（实训）报 告**

**项 目 名 称**  进程管理

**所属课程名称**  操作系统

**项 目 类 型**  验证/设计型

**实验(实训)日期**  2024.11.13

**班 级**  22软件2班

**学 号**  220110900643

**姓 名**  张乐怡

**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. 学习gcc的安装 2. 运用gcc进行代码编写与运行 3. 学会进程调度模拟   **【基本原理】**  GCC是由GNU项目开发的编程语言编译器，支持多种编程语言，包括C、C++、Objective-C、Fortran、Java、Ada和Go等，以及这些语言的库。通过GCC，开发者可以高效地编译和优化代码。  动态优先权调度算法是操作系统进程调度的关键策略之一。该算法的核心在于根据进程的实际运行情况实时调整其优先级，以此达到系统资源的最优分配。在动态优先权调度中，每个进程被赋予一个优先级，优先级的数值越小表示优先级越高。一个进程的优先级可以基于多种因素动态调整，如等待时间、资源需求量、用户输入等。调度器的决策过程往往基于当前所有进程的优先级来进行，选择优先级最高的进程分配CPU时间。当多个进程优先级相同，则可以采用其他调度策略，如时间片轮转或随机选择。  **【实施环境】**  VMware 17.5.2  Ubuntu 22.04.4 |
| **二、实验（实训）内容：** |
| **【实验（实训）过程】（步骤、记录、数据、程序等）**  **实验五：**  ·安装gcc  安装命令为sudo apt install gcc  ·完成hello.c  1731485218915  ·完成fork.c  1731485623877  ·完成proceses.c  1731485651818  ·完成communication.c  1731486084836  **实验六：**  ·用c语言实现对n个进程采用动态优先权算法的调度  （代码详见补充文档，以下展示流程及结果）  1731487425595  结果：  1731487530655  1731487555703  1731487580318  1731487640334  1731487671855  1731487693316  1731487721518  1731487744026  **【结论与讨论】（结果、分析）**  在Linux系统中，安装GCC编译器大大简化了C语言程序的编译和调试过程。利用C语言实现对N个进程采用动态优先权算法的调度，可以通过定义进程控制块PCB结构，包括进程ID、优先级、CPU时间等关键字段，并设定优先数变化规则来管理就绪、阻塞和运行队列。实验结果表明，动态优先权调度算法能够根据进程的优先级合理调度进程，提高系统效率。然而，由于该算法是非抢占式的，存在进程饥饿问题，可能需要结合其他调度算法进行补充。总的来说，本次实验加深了对进程概念和进程调度过程的理解。 |
| **三、指导教师评语及成绩：** |
| **评语：**  **成绩： 指导教师签名：**  **批阅日期：** |