

**实 验（实训）报 告**

**项 目 名 称**  进程控制与通信、进程调度

**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**  2024/11/20

**班 级**  软工一班

**学 号**  220110900728

**姓 名**  吴辰

**指导教师**  陈伟锋

浙江财经大学教务处制

|  |
| --- |
| **一、实验（实训）概述：（简单复述实验要求，1页以内，红字部分应删除）** |
| **【目的及要求】**  **实验五：**  安装gcc  完成hello.c  完成fork.c  完成processes.c  完成communication.c  **实验六：**  目的：  通过实验，掌握操作系统中进程调度的基本概念和机制。  了解动态优先级调度算法的原理和实现方式。  通过模拟进程调度，掌握优先级变化规则对进程调度过程的影响。  要求：  使用C语言编写程序，模拟N个进程的动态优先级调度。  设计合理的进程控制块（PCB）结构，包括优先级、CPU时间、总时间、开始阻塞时间、阻塞时间等信息。  程序能够根据进程状态和优先级进行动态调度，并在每个时间片内输出调度信息和进程状态变化。  **【基本原理】**  **实验五：**  安装gcc  要求: 使用包管理工具（如apt-get）安装GCC。  完成hello.c  要求: 编写一个打印"Hello, World!"的简单程序。  完成fork.c  要求: 编写一个使用fork()函数的程序，父进程和子进程各自打印不同的消息。  完成processes.c  要求: 编写一个程序，展示如何创建、等待和终止多个进程。  完成communication.c  要求: 编写一个程序，演示进程间的通信方法，如管道、共享内存等。  **实验六：**  进程调度：  进程调度是操作系统中的核心功能之一，用于管理系统中多个进程的执行顺序和资源分配。调度算法决定了哪个进程在什么时候运行，以实现高效的资源利用和合理的响应时间。  动态优先级调度算法：  该算法根据进程的优先级进行调度，优先级是一个可变化的数值，通常根据进程的执行情况进行调整。  **【实施环境】**  **操作系统：Windows**  **Linux VM Pro 16** |
| **二、实验（实训）内容：（记录实验内容，要求逻辑清晰、页面整洁、必要的图文并茂，核心的数据及程序代码，不能完全复制所有代码，完整代码若有必要，可以增加附录，尽量控制在6页以内，红字部分应删除）** |
| **【实验（实训）过程】（步骤、记录、数据、程序等）‘** 实验五 **安装gcc**    **编译hello.c**          **编译fork.c**      **编译processes.c**    **编译communication.c**   实验六     **【结论与讨论】（结果、分析）** |
|  |
| **三、指导教师评语及成绩：** |
| **评语：**  **成绩： 指导教师签名：**  **批阅日期：** |