

**实 验（实训）报 告**

**项 目 名 称**  进程控制

**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** 2024/11/11

**班 级**  22软件2班

**学 号**  220110900240

**姓 名**  应罗毅

**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】**  安装 gcc ,完成 hello.c、fork.c、proceses.c、communication.c。  完成进程调度模拟，实现对 N 个进程采用动态优先权算法的调度。    **【基本原理】**  1.编译和运行程序：  在命令行中，导航到每个.c 文件所在的目录，并使用 gcc 命令编译它们，例如  gcc hello.c -o hello。  运行编译生成的可执行文件，例如./hello。  2.进程调度模拟：  动态优先权算法根据进程的等待时间和服务时间来调整其优先级。  定义一个进程控制块（PCB）来存储每个进程的信息，实现用调度器来管理 PCB。  **【实施环境】**  **Linux Ubuntu** |
| **二、实验（实训）内容：** |
| **【实验（实训）过程】（步骤、记录、数据、程序等）**  1.通过包管理器apt来安装GCC：实际上安装的是"build-essential"    2.完成hello.c,并加载到linux中,通过命令执行    结果:    3.进行动态优先权算法      **【结论与讨论】（结果、分析）**  **实验结果表明，GCC编译器在Linux环境下成功安装并配置，使得C语言程序的编译和运行成为可能。通过fork()函数的运用，我创建了子进程，并通过对父进程与子进程行为的对比分析，加深了对进程创建机制的理解。在对N个进程进行动态优先权调度算法的实践中，我通过动态调整进程优先级，提升了系统的响应速度和资源利用率。动态优先权算法的实现过程较为复杂，需要综合考虑进程的CPU占用率、等待时间等多个因素。合理的优先级调整策略对于提升系统效率具有关键作用。这次实验不仅加深了我对理论知识的理解，也提升了我的实践操作能力，对于未来学习操作系统的高级功能打下了坚实的基础。** |
| **三、指导教师评语及成绩：** |
| **评语：**  **成绩： 指导教师签名：**  **批阅日期：** |