

**实 验（实训）报 告**

**项 目 名 称**  实验二 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**  2024.10.30

**班 级**  22软件3班

**学 号**  220110900735

**姓 名**  徐峥扬

**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. 熟悉gcc。 2. 培养动手实践能力。 3. 学会问题总结与互联网检索。 4. 练习使用gcc写代码并运行。 5. 进一步了解加深操作系统的进程概念。   **【基本原理】**  GCC是一个开源的编译器集合，支持多种编程语言（如C、C++、Objective-C、Fortran、Ada和Go等），能够将源代码编译成可执行文件。熟悉GCC意味着了解它的基本功能、命令行选项、编译流程（预处理、编译、汇编、链接）以及如何使用它来构建和调试程序。使用GCC进行编译，并运行生成的可执行文件，可以直观地看到代码的执行结果和编译器的输出信息。这个过程有助于理解代码的编译过程、调试技巧以及优化方法。  **【实施环境】**  **虚拟机软件：VMware Workstation Pro**  **操作系统：Linux** |
| **二、实验（实训）内容：** |
| **【实验（实训）过程】（步骤、记录、数据、程序等）**   1. 安装gcc   sudo apt update  sudo apt install build-essential  验证安装gcc –version     1. 完成hello.c，fork.c, process.c, communication.c   Hello.c      Fork.c      Process.c      Communication.c       1. 用C语言实现对N个进程采用动态优先权算法的调度           输出结果：    **【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：** |