**重庆交通大学信息科学与工程学院**

**综合性设计性实验报告**

**班 级： 计科1703**

**姓名 学号： 黄富灵 631707060308**

**实验项目名称： 实验八、多线程技术**

**实验项目性质： 综 合 性**

**实验所属课程： 《Java程序设计》**

**实验室(中心)：**

**指 导 教 师 ： 王桂平**

**实验完成时间： 2019年 4月 8 日**

教师评阅意见：

签名： 年 月 日

实验成绩：

**一、实验题目**

使用Java的多线程技术编写一个控制台(console)或GUI程序。

**二、实验要求**

1) 分组要求：分组完成，自由组队，2人一组。

2) 程序要求：不少于150行代码，且具有较好的可读性（注释、正确缩进等）。

3) 文档要求：详细阐述该程序的设计思想、采用的Java技术或方法。

**三、设计思想、采用的Java技术或方法**

调用一个线程的 start 方法时，它在一个线程中调用 run 方法里面的代码。如果我们不调用 start 方法，直接运行 run 方法。那么实际上是在主线程中运行 run 方法，而不是两个子线程中分别运行的。

**四、主要程序代码**

*/\**

*\* @Coding: utf-8 \n*

*\* @Despcription: \n*

*\* @Author: Suwen SP \n*

*\* @LastEditors: Suwen SP\n*

*\* @Date: 2019-04-08 18:37:05*

*\* @LastEditTime: 2019-04-08 18:40:11*

*\*/*

*//java 多线程实例*

*//实验目的*

*//继承Thread类*

*//覆盖run方法*

*//调用子类的时候调用线程对象*

*//调用start方法启动线程*

*//run方法里面的代码是启动多线程要执行的方法。*

class Demo extends Thread {

private String name;

public Demo(String *name*) {

this.name = name;

}

public void run() {

for (int i = 0; i < 5; i++) {

System.out.println(Thread.currentThread().getName() + "<----->" + name + "<----->" + i);

}

}

}

public class Thread\_test {

*// 三个线程在运行，main(主线程),Thread-1线程,Thread-0线程*

public static void main(String[] *args*) {

Demo d1 = new Demo("A");*// 创建之后线程就被创建了*

Demo d2 = new Demo("B");

*// start方法会做两件事，开启线程，运行run方法*

d1.start();

d2.start();

for (int i = 0; i < 10; i++) {

*// 主线程运行的方法就是main*

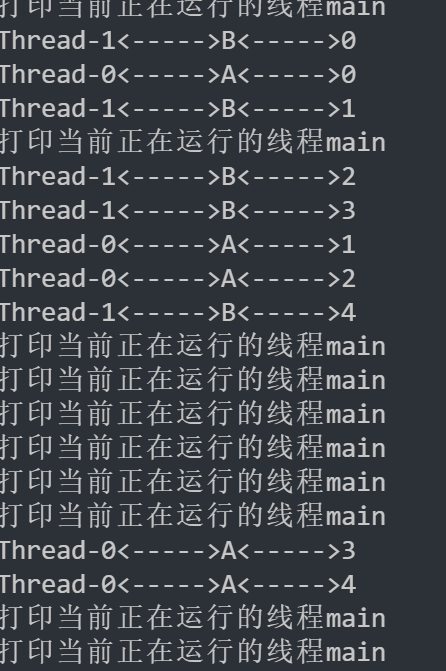
System.out.println("打印当前正在运行的线程" + Thread.currentThread().getName());

}

}

}

**五、测试结果及说明**



**六、实验体会**

在正常情况下，让程序来完成多个任务，只使用单个线程来完成比用多个线程完成所用的时间会更短。因为 JVM 在调度管理每个线程上肯定要花费一定资源和时间的。在一些条件下多线程是非常有用的，一是对用户响应要求比较高，又允许用户并发访问的场景；二是程序存在耗费时间的计算，整个系统都会等待这个操作，为了提高程序的响应，将耗费时间的计算通过线程来完成。