**深 圳 大 学 实 验 报 告**

**课程名称： Java程序设计**

**实验项目名称： 必实验4 线程应用**

**学院： 计算机与软件学院**

**专业： 计算机科学与技术（创新班）**

**指导教师： 姚俊梅**

**报告人：何泽锋 学号： 2022150221 班级：高性能特色班**

**实验时间： 2023年11月20日（周一）~2022年12月2日（周六）**

**实验报告提交时间：**

**教务部制**

|  |
| --- |
| **实验目的与要求：**  **实验目的：** 掌握Java程序设计中的线程同步等技术。  **实验要求：**  (1). 运行以下三个程序（每个程序运行10次），并对输出结果给出分析。在报告中附上程序截图和详细的文字说明。（15分）  **程序1:**  **程序2:**  **程序3:**  (2). 编写Java应用程序实现如下功能：第一个线程输出数字1,2,..,12，第二个线程输出英文单词数字和月份One January, Two February, …, Twelve December，输出的顺序和格式为1OneJanuary2TwoFebruary...12TwelveDecember，即每1个数字紧跟着2个英文单词的方式。要求线程间实现通信。要求采用实现Runnable接口和Thread类的构造方法的方式创建线程，而不是通过Thread类的子类的方式。在报告中附上程序截图、运行结果截图和详细的文字说明。（15分）  (3). 编写Java应用程序实现如下功能：创建工作线程，模拟银行现金账户取款操作。多个线程同时执行取款操作时，如果不使用同步处理，会造成账户余额混乱，要求使用syncrhonized关键字同步代码块，以保证多个线程同时执行取款操作时，银行现金账户取款的有效和一致。要求采用实现Runnable接口和Thread类的构造方法的方式创建线程，而不是通过Thread类的子类的方式。在报告中附上程序截图、运行结果截图和详细的文字说明。（25分）  (4). 有一座东西向的桥，只能容纳一个人，桥的东边有20个人（记为E1,E2,…,E20）和桥的西边有20个人（记为W1,W2,…,W20），编写Java应用程序让这些人到达对岸，每个人用一个线程表示，桥为共享资源，在过桥的过程中输出谁正在过桥（不同人之间用逗号隔开）。运行10次，分别统计东边和西边的20人先到达对岸的次数。要求采用实现Runnable接口和Thread类的构造方法的方式创建线程，而不是通过Thread类的子类的方式。在报告中附上程序截图、运行结果截图和详细的文字说明。（25分）  报告写作。要求：主要思路有明确的说明，重点代码有详细的注释，行文逻辑清晰可读性强，报告整体写作较为专业。（20分）  **说明：**  （1）本次实验课作业满分为100分，占总成绩的比例（待定）。  （2）报告正文：请在指定位置填写，本次实验**需要单独提交源程序文件**。  （3）个人信息：WORD文件名中的“姓名”、“学号”，请改为你的姓名和学号；实验报告的首页，请准确填写“学院”、“专业”、“报告人”、“学号”、“班级”、“实验报告提交时间”等信息。  （4）提交方式：截至时间前，请在Blackboard平台中提交。  （5）发现抄袭（包括复制&粘贴整句话、整张图），**抄袭者和被抄袭者的成绩记零分（含抄袭往届同学的作业）。**  （6）延迟提交酌情扣分，期末考试阶段补交无效。 |
|  |

|  |
| --- |
| (1). 运行以下三个程序（要求每个程序运行10次），并对输出结果给出分析。在报告中附上程序截图和详细的文字说明。（15分）  **程序1:**  ①按照题目要求将程序运行10次，运行的结果如下  ②首先观察结果可以看见，thread1，thread2，thread3的运行顺序是不确定的，并不是按照start()方法的顺序进行的，再者可以看到，直接输出字符的进程会比输出数值的进程要快。  ③原因分析：当程序有多个进程的时候，线程之间是通过轮询机制来使用CPU资源，当轮到该进程并且进程已经准备完成才会运行进程。多次测试结果都可以看见PrintChar的进程会比PrintNum的进程要快，对代码进行适当更改并测试后发现，这是因为PrintNum的输出内容是“ ”+i，而PrintChar是输出一个字符，前者需要将int型的i转为char并于“ ”放在一起输出，而后者可以直接输出字符，因此前者进程分配的时间较后。  **程序2:**  ①代码的运行结果规律与程序1基本相同，此处不再进行过多解释，结果图如下：  ②此处代码并没有直接使用Thread以及对应的start()方法，而是采用Executor来启动线程，这主要是使用了线程池。  ③使用线程池的优点，可以降低CPU资源的消耗，可以重复利用已创建的线程。提高相应速率以及可管理性，通过线程池进行统一分配、监控等等。  **程序3:**  ①按照题目要求输入代码并运行10次，运行的结果如下图所示：  ②可以看到进程有多种输出结果，这主要是因为多个进程竞争的结果，观察代码，在exector时使用了for循环开了100个线程任务，并且后续使用shutdown方法关闭线程，但需要注意的是，shutdown只是不再接受新的线程，若原来已经加载的线程会继续运行  ③输出0可能是因为先运行了输出相关的进程而还未进行account的deposit方法。而输出1则刚好相反。对于输出2的结果，可以看见每次有2都会是输出多次，原因可能为for循环的100个进程都被加载了，并且是同时运行所以结果都为2.  (2). 编写Java应用程序实现如下功能：第一个线程输出数字1,2,..,12，第二个线程输出英文单词数字和月份One January, Two February, …, Twelve December，输出的顺序和格式为1OneJanuary2TwoFebruary...12TwelveDecember，即每1个数字紧跟着2个英文单词的方式。要求线程间实现通信。要求采用实现Runnable接口和Thread类的构造方法的方式创建线程，而不是通过Thread类的子类的方式。在报告中附上程序截图、运行结果截图和详细的文字说明。（15分）  ①按照题目要求，需要定义两个线程分别命名为num和month，对应输出数字和月份信息，定义成员变量，包括num用来遍历数字输出，num1遍历月份输出，str字符串组用来存放月份。采用synchronized实现线程间的通信。  ②但线程名称为“num”时，若num小于12，即未输出完12个月份，则采用try-catch方法输出当前的数值，并先唤醒所有线程，再让当前线程wait。  ③线程month的逻辑同上，只是输出时需要通过num1遍历输出str字符组。  ④进程最后再次唤醒所有进程  ⑤定义Use类，将Runnable作为接口，通过有参构造输入其成员变量的值。重构run方法，调用Thread.currentThread().getName().equals()方法判断该name是否符合Month类的进程名称，若是则运行该进程  ⑥Main函数定义m通过有参构造赋值，再Thread两个实参，将m传入，然后通过setName方法设置线程名称。最后通过start方法启用线程  ⑦运行结果如下，可以看见数字与月份一一对应，说明两个线程实现了沟通，wait方法有效。  (3). 编写Java应用程序实现如下功能：创建工作线程，模拟银行现金账户取款操作。多个线程同时执行取款操作时，如果不使用同步处理，会造成账户余额混乱，要求使用syncrhonized关键字同步代码块，以保证多个线程同时执行取款操作时，银行现金账户取款的有效和一致。要求采用实现Runnable接口和Thread类的构造方法的方式创建线程，而不是通过Thread类的子类的方式。在报告中附上程序截图、运行结果截图和详细的文字说明。（25分）  ①首先定义Bank类来控制不同线程运行结果，含有一个私有成员money，初始值为10000，表示银行账户中有10000元。此处只定义一种线程“withdrew”用来实现取款操作，可以根据实际需求添加其他功能。  ②取款线程具体内容：如果当前账户余额大于0则一直进行取钱操作，操作成功一次就唤醒其他线程并挂起当前线程，此处为了区分每次取钱造作，设置了两种取钱数值，可以通过参数b定义输入的数值大小。  ③定义WithdrawTread，将Runnable作为接口，进行有参构造时注意要将bank定义为输入参数，才能对同一个账户进行不同线程操作，其中s是线程名，b为取钱大小。重构run方法，如果名字符合线程名则调用Withdraw方法。  ④Main中定义一个bb作为账户，有两个线程，都是取钱操作，一个取100，一个取200，再设置线程的名称，最后通过start方法调用线程。  ④程序运行结果如下，可以发现两个线程是交叉运行的，多次运行后可以发现，首个线程是不确定的。    (4). 有一座东西向的桥，只能容纳一个人，桥的东边有20个人（记为E1,E2,…,E20）和桥的西边有20个人（记为W1,W2,…,W20），编写Java应用程序让这些人到达对岸，每个人用一个线程表示，桥为共享资源，在过桥的过程中输出谁正在过桥（不同人之间用逗号隔开）。运行10次，分别统计东边和西边的20人先到达对岸的次数。要求采用实现Runnable接口和Thread类的构造方法的方式创建线程，而不是通过Thread类的子类的方式。在报告中附上程序截图、运行结果截图和详细的文字说明。（25分）  ①首先定义Bridge类，拥有多个成员变量，用来打标记或记录输出个数，主要变量是两个字符串组L和R，存放对应的20个人。  ②采用synchronized，每次都遍历20个人，如果当前走的人是这20个中的一个（分东西两侧），则输出其标号，并且统计当前走过的人数，若一侧先走完所有人，则将对应的check标记置1。线程的沟通方法同上任务3。  ③此处是输出哪一侧先全部到达。  ④定义对应的BridgeThread，将Runnable作为接口，运行时遍历20次是否有人走过，通过if判断东西两侧，调用对应的Walk方法。  ⑤主函数大部分与之前相似，需要注意的是进程命名和开始采用了for循环。  ⑥程序运行10次结果如下，可以看到每次走过的顺序都不相同，可以看出线程的不确定性，并且可以看到两侧谁先全部到达，统计结果为西：东=6：4。但运行次数足够多时，其结果应当趋近于1：1。  ++++++++++++++++++++++++++++++++++++++++++++++++++++++  **其他（例如感想、建议等等）。**  本次实验学习了Java多进程的相关知识，通过代码可以更加清晰的了解其内部逻辑。从任务1理解了多线程的运行是分时的，并不是根据代码的顺序进行运行，而是由CPU分配，再者对于不同的线程其所用时间也是不一样的，因此其运行顺序也会受到影响。在第二个样例，了解了线程池的概念，通过Executor进行定义，线程池的优点包括可以降低CPU资源的消耗，可以重复利用已创建的线程。提高相应速率以及可管理性，通过线程池进行统一分配、监控等等。通过任务2、3、4可以更加清晰的了解到如何自定义线程分类，并调用输出。 |

深圳大学学生实验报告用纸

|  |
| --- |
| 指导教师批阅意见：  成绩评定：  指导教师签字：  2023年 月 日 |
| 备注： |

注：1、报告内的项目或内容设置，可根据实际情况加以调整和补充。

2、教师批改学生实验报告时间应在学生提交实验报告时间后10日内。