

**信息科学与技术学院**

School of Information Science and Technology

**学生实验报告**

**（2021 —2022 学年第 1 学期）**

|  |  |  |  |
| --- | --- | --- | --- |
| **学生姓名** |  | **学号** |  |
|  |  |  |  |
| **专业名称** |  | **所在班级** |  |
|  |  |  |  |
| **课程名称** | 网络编程实践 | | |
|  |  |  |  |
| **教师姓名** | 贾中云 | **实验地点** | 勤园13-204 |

信息科学与技术学院教学部印制

实验报告要求

1、实验需独立完成，实验结果真实可靠；

2、实验报告中实验内容写于表格中，表格报告页面不够可“插入”一行

3、实验报告格式不可更改

4、实验报告内容包括程序及结果

5、实验结果可采用粘贴图片的方式，同时须对结果进行讨论

6、每个实验的思考题须回答，每个实验完成后须给出实验心得

**学 生 实 验 报 告**

|  |  |  |  |
| --- | --- | --- | --- |
| 实验名称 | 实验2线程的使用 | | |
| 实验地点 | 勤园13-204 | 实验日期 |  |
| 实验仪器  （软件、硬件  环境） | 装有Java JDK开发套件及Eclips的计算机 | | |
| 实验目的（要求） | 1、理解线程的概念，定义和运行线程；  2、掌握线程的编程方法（Thread类和Runnable接口）；  3、掌握多线程的编程。 | | |
| 实验主要步骤  （硬件类为：  原理、主要步骤、电路原理图等）  （软件类为：  数据结构、算法、主要步骤、界面等） | 撰写提示：  （实验内容和步骤应简明扼要，可以将图、代码等内容图片方式粘贴于此）  **1、用Thread类继承的方法创建线程：**  以下程序代码是用创建一个Thread子类的方法实现线程创建的程序代码。  import java. io.\*;  import java.security.\*;  import javax.xml.bind.\*;  public class DigestThread extends Thread {  private String filename;  public DigestThread(String filename) {  this.filename = filename;  }  @Override  public void run() {  try {  FileInputStream in = new FileInputStream(filename);  MessageDigest sha = MessageDigest.getInstance("SHA-256");  DigestInputStream din = new DigestInputStream(in, sha);  while (din.read() != -1);  din.close();byte[] digest = sha.digest();  StringBuilder result = new StringBuilder(filename);  result.append(": ");  result.append(DatatypeConverter.printHexBinary( digest)); | | |
| （本栏页面不够可附页） | | |
| 实验主要步骤  （硬件类为：  原理、主要步骤、电路原理图等）  （软件类为：  数据结构、算法、主要步骤、界面等） | System.out.println(result);  } catch (IOException ex) {  System.err.println(ex);  } catch (NoSuchAlgorithmException ex) {  System.err.println(ex);  }  }  public static void main(String[] args) {  for(String filename : args) {  Thread t = new DigestThread(filename);  t.start();  }  }  }  修改、调试程序，使程序可运行。然后在目标位置建立4个.txt文件，分别对这4个文件建立线程并输出结果。  运行结果（图）并讨论：    思考题：  （1）程序中的@Override语句表示什么意思？这里为什么要用到这个语句？  重写，因为是父类的方法  （2）程序中哪句语句表示开始建立并运行线程？  Start()  （3）修改程序，使得每个线程的SHA数据在输出时，格式变成每输出2个HEX数据接1个空格。 | | |
| 实验主要步骤  （硬件类为：  原理、主要步骤、电路原理图等）  （软件类为：  数据结构、算法、主要步骤、界面等） | （3）修改后的程序：  运行结果（图）并讨论： | | |
| 实验主要步骤  （硬件类为：  原理、主要步骤、电路原理图等）  （软件类为：  数据结构、算法、主要步骤、界面等） | **2、以下为用Runnable方法实现线程创建的代码：**  import java.io.\*;  import java.security.\*;  import javax.xml.bind.\*;  public class DigestRunnable implements Runnable {  private String filename;  public DigestRunnable(String filename) {  this.filename = filename;  }  @Override  public void run() {  try {  FileInputStream in = new FileInputStream(filename);  MessageDigest sha = MessageDigest.getInstance("SHA-256");  DigestInputStream din = new DigestInputStream(in, sha);  while (din.read() != -1);  din.close();  byte[] digest = sha.digest();  StringBuilder result = new StringBuilder(filename);  result.append(": ");  result.append(DatatypeConverter.printHexBinary(digest));  System.out.println(result);  } catch (IOException ex) {  System.err.println(ex);  } catch (NoSuchAlgorithmException ex) {  System.err.println(ex);  }  }  public static void main(String[] args) {  for(String filename : args) {  DigestRunnable dr = new DigestRunnable(filename);  Thread t = new Thread(dr);  t.start();  }  } } | | |
| 实验主要步骤  （硬件类为：  原理、主要步骤、电路原理图等）  （软件类为：  数据结构、算法、主要步骤、界面等） | 修改、调试程序，使程序可运行。同样在这个程序中建立4个读取文件内容并进行加密的线程，运行程序并输出结果。比较这个方法与Thread方法在程序上的不同之处，并给出其结果。  运行结果及讨论：    **3、返回结果：**  以下两个程序一起用于返回结果。  程序A：建立返回结果的线程ReturnDigest  import java.io.\*;  import java.security.\*;  public class ReturnDigest extends Thread {  private String filename;  private byte[] digest;  public ReturnDigest(String filename) {  this.filename = filename;  }  @Override  public void run() {  try {  FileInputStream in = new FileInputStream(filename);  MessageDigest sha = MessageDigest.getInstance("SHA-256");  DigestInputStream din = new DigestInputStream(in, sha);  while(din.read() != -1); //读取整个文件  din.close();  digest = sha.digest();  } catch (IOException ex) {  System.err.println(ex); | | |
| 实验主要步骤  （硬件类为：  原理、主要步骤、电路原理图等）  （软件类为：  数据结构、算法、主要步骤、界面等） | } catch (NoSuchAlgorithmException ex) {  System.err.println(ex);  }  }  public byte[] getDigest() {  return digest;  }  }  程序B：程序运行方式：  **import** javax.xml.bind.\*;  **public** **class** ReturnDigestUserInterface {  **public** **static** **void** main(String[] args) {  **for** (String filename : args) {  // 计算摘要  ReturnDigest dr = **new** ReturnDigest(filename);  dr.start();  //现在显示结果  StringBuilder result = **new** StringBuilder(filename);  result.append(": ");  **byte**[] digest = dr.getDigest();  result.append(DatatypeConverter.*printHexBinary*(digest));  System.***out***.println(result);  }  }  }  （1）调试并运行这两个程序，使得程序能对建立的.txt文件数据结果输出。  （2）修改线程运行方式，使得建立的线程能直接输出.txt文件的内容。  （3）修改程序的运行方式，使ReturnDigestUserInterface能以交互的方式运行。即由程序提示.txt文件的输入，当最后一个文件输入结束时输出结果。  运行结果（多运行几次）及讨论： | | |
| 实验主要步骤  （硬件类为：  原理、主要步骤、电路原理图等）  （软件类为：  数据结构、算法、主要步骤、界面等） | 出现这样结果的原因是什么？程序中的竞态条件是什么？如何修改程序才能正常显示结果？  修改后的程序：  运行结果（图）及讨论： | | |
| 实验结果  或结论 | 1、这三个程序建立的线程分别是干什么的？  2、这三个程序都用到java.security类包和javax.xml.bind类包。这两个类包分别在程序中什么地方用到？  3、程序中的异常Exception类在Java的哪个类包中？  4、程序中“SHA-256”的作用是什么？实现这个运算的原理是什么？ | | |
| 实验心得  （重点填写内容） | 撰写提示：  （简述实验涉及的知识点、实验过程中遇到哪些难题、如何分析与处理难题、对实验结果的思考、学习建议） | | |
| 实验指导教师评语及评分 |  | | |

注：学生做每个实验都必须填写实验报告。实验报告可以是纸质的，也可以是电子形式的。凡电子文本形式的实验报告须在学期末刻录进该学期教学资料光盘并存个人教学档案。

实验指导教师签名\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_年\_\_\_\_ 月\_\_\_\_ 日