**深 圳 大 学 实 验 报 告**

**课程名称：Java程序设计**

**实验项目名称： 选实验4 常用集合类使用**

**学院： 计算机与软件学院**

**专业： 计算机科学与技术**

**指导教师：潘微科**

**报告人： 刘俊楠 学号：2017303010 班级： 01**

**实验时间：2021年11月12日（周五）~2021年11月17日（周三）**

**实验报告提交时间： 2021.11.15**

**教务部制**

|  |
| --- |
| **实验目的与要求：**  **实验目的：**掌握常用的集合类，能够较为熟练的查阅Java提供的常见的类，并进行程序设计。  **实验要求：**  (1). 编写Java应用程序，根据用户输入的5个时间（每行一个时间），计算相邻两个时间之间的间隔，共4个结果。注：用户输入的时间格式和程序输出的时间格式均为为“××××年××月××日××时××分××秒”。在报告中附上程序截图、完整的运行结果截图和简要文字说明。（40分）  (2). 编写Java应用程序，实现稀疏矩阵的加法和乘法运算，其中稀疏矩阵是指矩阵中的大部分元素的值为0。用户在命令行输入矩阵时矩阵的大小可能有错，因此需要使用异常处理。在报告中附上程序截图、完整的运行结果截图和简要文字说明。（40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）本次实验课作业截至时间2021年11月17日（周三）21:59。  （3）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**（源程序单独打包在Blackboard中上传，不要包含外部导入的包）。  （4）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。  （5）提交方式：截至时间前，请在Blackboard平台中提交。  （6）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）。**  （7）延迟提交，不得分；如有特殊情况，请于截至日期之后的48小时内发邮件到panweike@szu.edu.cn，并在邮件中注明课程名称、作业名称、姓名、学号等信息，以及特殊情况的说明，我收到后会及时回复。  （8）期末考试阶段补交无效。 |

|  |
| --- |
| (1). 编写Java应用程序，根据**用户输入的5个时间（每行一个时间）**，计算相邻两个时间之间的间隔，共4个结果。注：用户输入的**时间格式和程序输出的时间格式均为**“××××年××月××日××时××分××秒”。在报告中附上程序截图、完整的运行结果截图和简要文字说明。（40分）  1、简要文字说明   * 分别引入字符串转换date异常包ParseException,格式化时间包SimpleDateFormat,以及Date包、scanner包。  * 创建一个Date数组，并声明scanner对象，获取输入值，利用SimpleDateFormat以及其parse的方法将输入的字符串按照pattern的格式转换成date类型数值赋值给time[i]，并且使用try catch捕捉异常。  * 创建for循环五次，利用date对象的getTime方法计算出时间戳的毫秒数，然后做差并取绝对值计算差值，随后通过一系列除余计算算出两日期所间隔的天、时、分、秒。   2、程序截图  3、结果截图   * 第三第四时间差因为在2100-2200一百年之间，有24个闰年，所以多了24天   (2). 编写Java应用程序，实现稀疏矩阵的加法和乘法运算，其中稀疏矩阵是指矩阵中的大部分元素的值为0。用户在命令行输入矩阵时矩阵的大小可能有错，因此需要使用异常处理。在报告中附上程序截图、完整的运行结果截图和简要文字说明。（40分）  1、简要文字说明   * 本题我一开始思路并不是很清晰，思考良久决定用HashMap做. * 下图为Main函数部分，我们首先创建对应二维HashMap变量，m1\_temp用于装载输入的第一个矩阵，m2\_temp用于装载输入的第二个矩阵，m2\_2\_temp用于装载输入的第二个矩阵的转置矩阵，m\_add\_result用于装载矩阵加法的结果矩阵，m\_substract\_result用于装载矩阵乘法的结果矩阵。 * 为了检查错误，我们设置了一个不会中断的while循环，提示用户输入矩阵大小以及非零元素个数，检查矩阵1行列是否等于矩阵2的行列的值，并算出稀疏密度检查是否小于等于0.05，反之，则抛出异常，此处计算0.05时由于double误差问题，我使用了bigdecimal来修正，使之保留两位小数，不会出错。  * 在上图中，由于是二维HashMap，我们需要对其每行进行初始化，否则会出现错误。  * 我们创建了一个中间HashMap，用于存储矩阵创建过程中需要的HashMap<Integer,Integer>，并且对用户输入进行判断，若不为零则将行列与值put进二维HashMap中，最后我们声明并定义Matrix类，利用他们的构造方法、print方法、以及add、mul方法实现矩阵的加、乘与输出。  * 定义一个Matrix类，用来装载矩阵并进行后续运算，其中包括HashMap<Integer,HashMap<Integer,Integer>>两个HashMap组成的嵌套变量matrix，用来转载稀疏矩阵中非零值与其对应行与列值，row用来记录矩阵行数，colomn用来记录矩阵列数，并定义有参构造方法。  * 添加矩阵相加相关方法add,参数为另一个matrix类  * 方法主要思路是先初始化一个二层HashMap装载结果，Iterator迭代器用于寻找下一个矩阵内非零键值对，Map内部接口Entry方法用于表示HashMap中的一个实体，HashMap.get()方法用于显示对应键的视图，entryset方法用于找到对应键值对集合。 * 因此this.matrix.get(i).entrySet().iterator()表示HashMap对应键视图下键值对集合的迭代器，iterator的hasNext方法表示是否有下一个键值对，代码中我通过iterator.next()方法强制类型转换为Map.Entry，并且利用Map.Entry的getkey()与getvalue()方法实现了寻找相应行列对应的非零值的功能，此代码中getkey()代表寻找对应列，getvalue()代表寻找相应值。  * 在上图代码中，逻辑是判断第i行HashMap是否有键值对，若有，则本类中变量matrix与m1的列比较，若matrix的列值大于m1的列值，则m1对应行和列对应的值赋值给结果矩阵，并且对m1的Map.Entry 类型值移动到下一位；若matrix的列值小于m1的列值，则把matrix对应值赋值给结果矩阵，并把matrix的Map.Entry类型指针引动到下一位；倘若两者行列都相等，则两两相加后赋值给结果矩阵，并两两移动Map.Entry类型值到下一位，如果有一方没有下一位了，就跳出循环。 * 跳出循环后，可能出现一方还有值，另一方已经没有非零值了，所以考虑两种情况对结果矩阵赋值，最后返回结果矩阵（二层HashMap）即可。  * 对于上图的矩阵乘法方法，主要思路与加法略微不同，其参数进来的是行列值颠倒的矩阵，这有利于后续运算，结果矩阵初始化与add方法相同，此次我们使用了双重循环，并且对matrix的行与m1的列分别求迭代器，二重循环是因为乘法需要计算一整行乘以一整列，计算中我们需要分别对matrix对应行上非零元素的列与m1对应列上非零元素的行进行比较，大于或小于的情况与add方法类似，等于的情况发生时我们就利用已经创建好的int result来累加结果，最后将result填入结果矩阵（二层HashMap）。  * Print方法主要用来输出二层HashMap对应的稀疏矩阵，我们首先利用Iterator迭代器定义外层HashMap键中所有键值对，然后在循环中定义另一个内层HashMap中所有的键值对，利用Map.Entry求取对应值，flag标记非零元素出现的位置，在flag之前与之后全部填充0，即可完成稀疏矩阵的print。   2、代码展示  3、结果展示  4、测试样例  10 10 95  10 10 95  0 0 0 0 0 0 0 0 1 2  0 0 0 0 0 0 0 0 0 1  0 1 0 0 0 0 0 0 0 0  0 0 0 0 0 1 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 1 2  0 0 0 0 0 0 0 0 0 1  0 1 0 0 0 0 0 0 0 0  0 0 0 0 0 1 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0  15 15 215  15 15 215  0 0 0 0 0 0 0 0 0 0 0 0 0 3 2  0 0 0 0 0 0 0 0 0 0 0 0 0 3 2  0 0 0 0 0 0 0 0 0 0 0 0 0 3 2  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 2 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 2 0 0 0 0 0 0 0 0 0 0 0 0 0  3 2 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 3 2  0 0 0 0 0 0 0 0 0 0 0 0 0 3 2  0 0 0 0 0 0 0 0 0 0 0 0 0 3 2  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 2 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  0 2 0 0 0 0 0 0 0 0 0 0 0 0 0  3 2 0 0 0 0 0 0 0 0 0 0 0 0 0  ++++++++++++++++++++++++++++++++++++++++++++++++++++++  **其他（例如感想、建议等等）。**  1、虽然这次实验只有两题，但是我感到非常的吃力，并且在完成该实验之后，我也感觉到我进步了很多，首先首先我学会了如何使用Date类与Calendar类，并且知道了他们的异同，然后我还明白了HashMap的二维使用方法，这些都让我感受到java的魅力，也更加深刻的了解到java与c++更多的不同。 |

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：  2021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。