

廈門大學



信息学院软件工程系

《JAVA 程序设计》实验报告

实验八

姓名：陈澄

学号：32420212202930

学院：信息学院

专业：软件工程专业

完成时间：2023.04.23

一、实验目的及要求

实验目的：

- 熟悉集合类

实验要求：

- 按照题目要求写代码，并在下次上课前将相关文件提交到 ftp

二、实验题目及实现过程

题目 1：编写一个程序，读取一个名字序列，将他们保存在一个 set 中，然后删除重复的名字。应允许用户输入名字。

（一）实验环境（集成开发环境、jdk 版本、字符编码等）

集成开发环境：IntelliJ IDEA

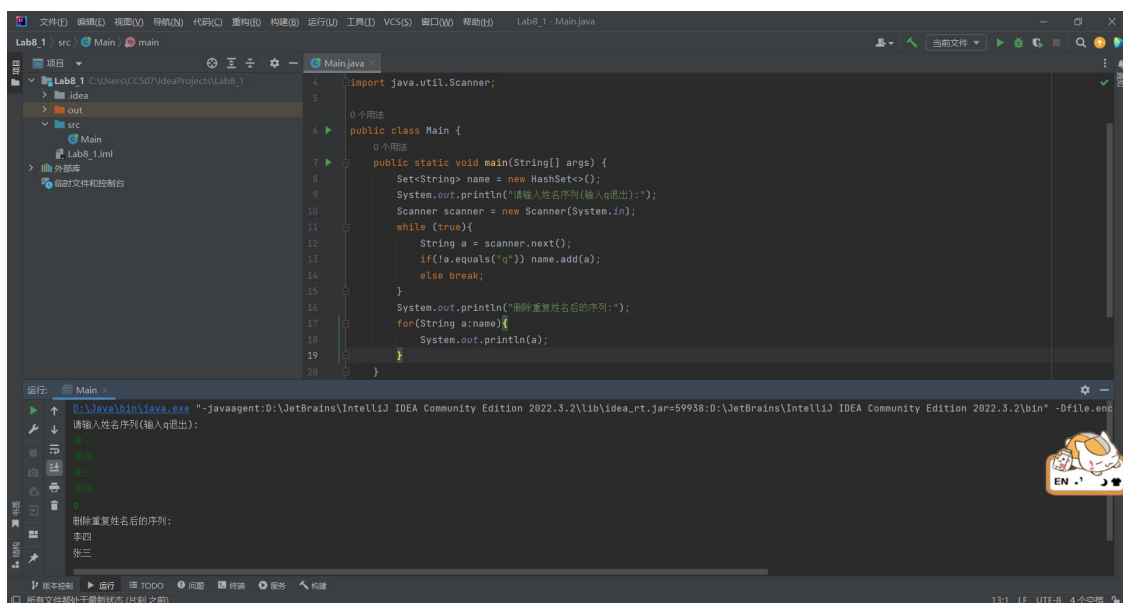
jdk 版本：17.0.5

字符编码：ASCII

（二）实现过程（**本部分为主要评分依据**，请描述解题思路，比如总共设计几个类，各个类的用途、成员、主要方法等及其之间调用关系等）

1. 建立一个名为 name 的 HashSet 用于储存姓名
2. 通过 while 读取用户输入，输入 q 时退出，若读取的输入非 q 则保存在 name 中。由于 HashSet 中不允许重复，在这个过程中重复的名字被剔除。
3. 通过 for 循环输出 name 中每一个字符串即可。

（三）过程截图（**本部分为主要评分依据**，一张全屏截图（必须）、若干运行结果展示图（可选），主要代码（可选））



题目 2：编写一个程序，它确定并输出用户输入的某个句子中重复的单词数。
不区分字母的大小写。忽略标点符号。

（一）实验环境（集成开发环境、jdk 版本、字符编码等）

集成开发环境：IntelliJ IDEA

jdk 版本：17.0.5

字符编码：ASCII

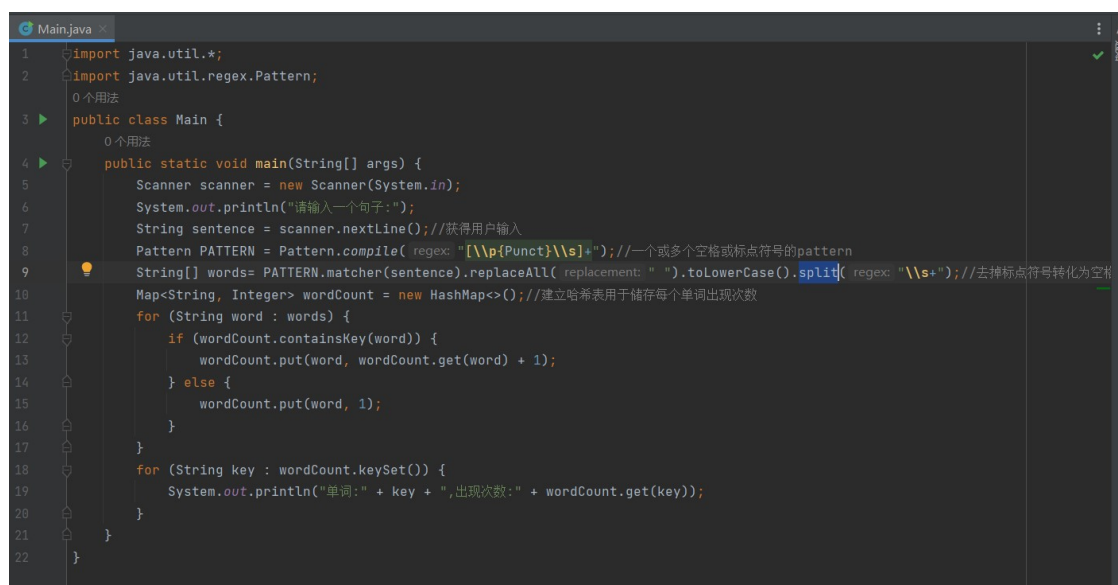
（二）实现过程（**本部分为主要评分依据**，请描述解题思路，比如总共设计几个类，各个类的用途、成员、主要方法等及其之间调用关系等）

1. 创建字符串 `sentence` 获取用户一整行的输入。
2. 创建 `Pattern` 型变量 `pattern`，格式为 `[p{Punct}\s]+`，代表一个或多个的标点符号或空格。
3. 通过 `match` 方法匹配上述 `pattern` 并通过 `replaceAll` 方法替换为空格，再通过 `toLowerCase` 方法将其变为小写，再通过 `split` 去掉空格并转化为字符串数组 `words`。
4. 创建两种参数分别为 `String,Integer` 类型的 `HashMap`（命名为 `wordCount`）来储存每个单词出现的次数。

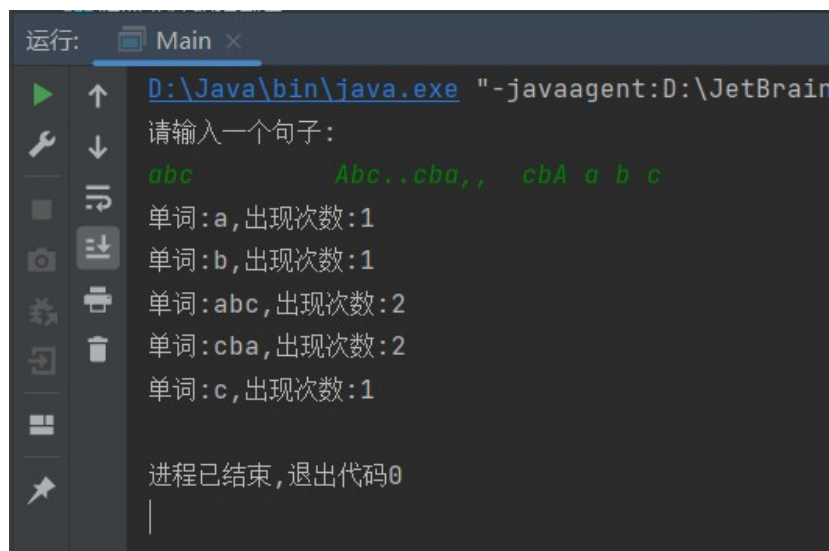
5. 对于 word 中的每个单词若在 wordCount 中出现, 则通过 get 方法取得其 Integer 值并用 put 方法使其+1。若未出现则将其出现次数置 1。

6. 输出: 通过 keyset 方法获得其字符串组, 并循环输出。

(三) 过程截图 (本部分为主要评分依据, 一张全屏截图 (必须)、若干运行结果展示图 (可选), 主要代码 (可选))



```
1 import java.util.*;
2 import java.util.regex.Pattern;
3
4 public class Main {
5     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7         System.out.println("请输入一个句子:");
8         String sentence = scanner.nextLine(); // 获得用户输入
9         Pattern PATTERN = Pattern.compile( regex: "[\\p{Punct}\\s]+" ); // 一个或多个空格或标点符号的pattern
10        String[] words= PATTERN.matcher(sentence).replaceAll( replacement: " ").toLowerCase().split( regex: "\\s+" ); // 去掉标点符号转化为空格
11        Map<String, Integer> wordCount = new HashMap<>(); // 建立哈希表用于存储每个单词出现次数
12        for (String word : words) {
13            if (wordCount.containsKey(word)) {
14                wordCount.put(word, wordCount.get(word) + 1);
15            } else {
16                wordCount.put(word, 1);
17            }
18        }
19        for (String key : wordCount.keySet()) {
20            System.out.println("单词:" + key + ", 出现次数:" + wordCount.get(key));
21        }
22    }
23 }
```



```
运行: Main x
D:\Java\bin\java.exe "-javaagent:D:\JetBrain
请输入一个句子:
abc Abc.cba,, cbA a b c
单词:a,出现次数:1
单词:b,出现次数:1
单词:abc,出现次数:2
单词:cba,出现次数:2
单词:c,出现次数:1
进程已结束,退出代码0
```

题目 3: 修改 wordTypeCount.java, 使其计算每个字母的出现次数而不是每个单词的出现次数。

（一）实验环境（集成开发环境、jdk 版本、字符编码等）

集成开发环境：IntelliJ IDEA

jdk 版本：17.0.5

字符编码：ASCII

（二）实现过程（**本部分为主要评分依据**，请描述解题思路，比如总共设计几个类，各个类的用途、成员、主要方法等及其之间调用关系等）

此处仅说明改动部分

1. 将 myMap 从<String, Integer>型改为<Character, Integer>型，其他有调用的位置随之更改。
2. createMap 方法中 `String[] tokens = input.split(" ");` 改为 `char[] chars = input.toLowerCase().replaceAll("\\s", "").toCharArray();`即将 input 转化为小写后删除所有空格再调用 toCharArray 方法转化为字符数组。
3. 后面将调用到 tokens 的地方全部更改为 chars 即可。

（三）过程截图（**本部分为主要评分依据**，一张全屏截图（必须）、若干运行结果展示图（可选），主要代码（可选））

```

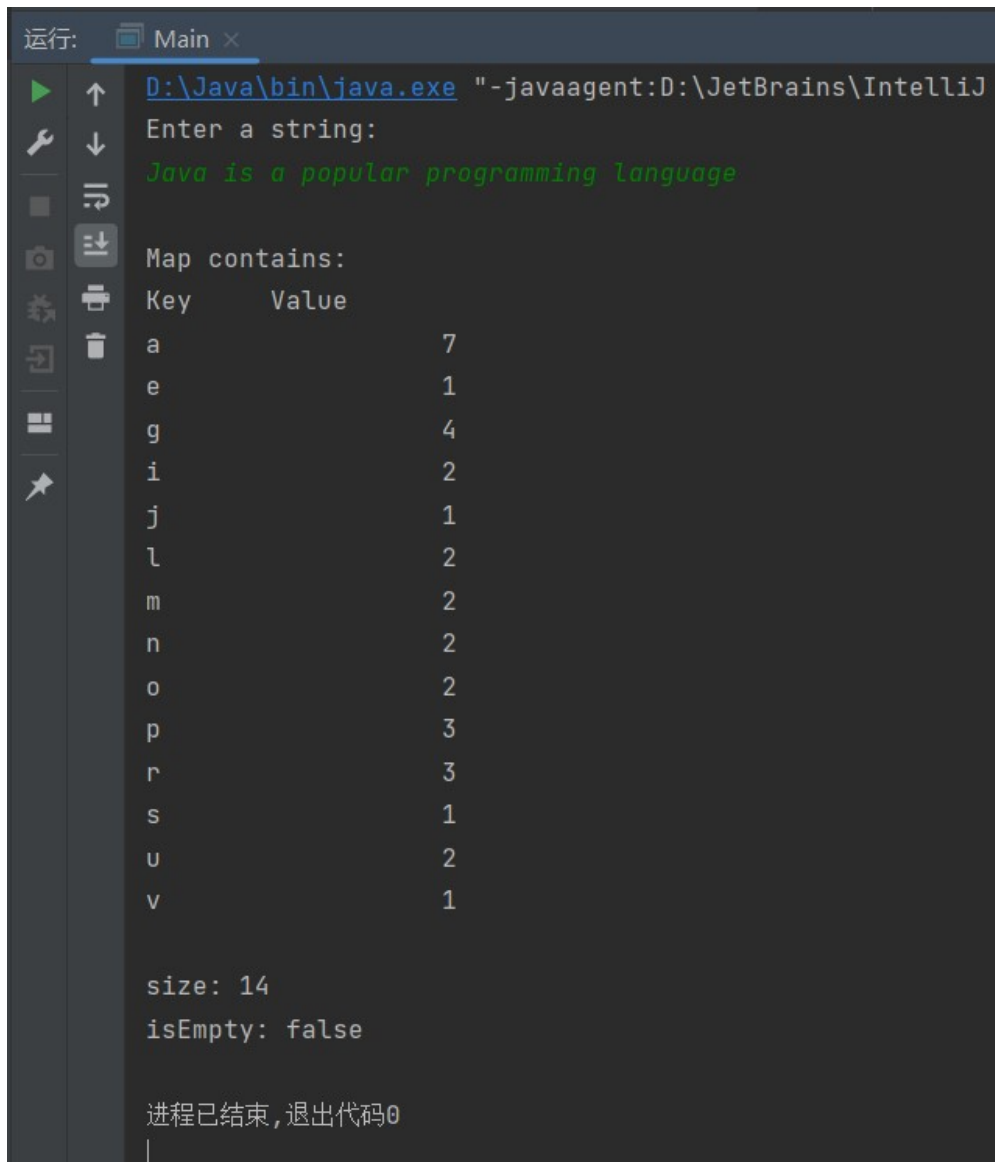
Main.java x
1  import java.util.Map;
2  import java.util.HashMap;
3  import java.util.Set;
4  import java.util.TreeSet;
5  import java.util.Scanner;
6
7  0个用法
8  public class Main {
9      0个用法
10     public static void main(String[] args) {
11         // create HashMap to store String keys and Integer values
12         Map<Character, Integer> myMap = new HashMap<>();
13
14         createMap(myMap); // create map based on user input
15         displayMap(myMap); // display map content
16     }
17
18     // create map from user input
19     1个用法
20     private static void createMap(Map<Character, Integer> map) {
21         Scanner scanner = new Scanner(System.in); // create scanner
22         System.out.println("Enter a string:"); // prompt for user input
23         String input = scanner.nextLine();
24
25         // tokenize the input
26         char[] chars = input.toLowerCase().replaceAll(regex: "\\s", replacement: "").toCharArray();
27
28         // processing input text
29         for (char c : chars) {
30             // String word = token.toLowerCase(); // get lowercase word

```

```

31         // if the map contains the word
32         if (map.containsKey(c)) { // is word in map?
33             int count = map.get(c); // get current count
34             map.put(c, count + 1); // increment count
35         }
36         else {
37             map.put(c, 1); // add new word with a count of 1 to map
38         }
39     }
40
41     // display map content
42     1个用法
43     private static void displayMap(Map<Character, Integer> map) {
44         Set<Character> keys = map.keySet(); // get keys
45
46         // sort keys
47         TreeSet<Character> sortedKeys = new TreeSet<>(keys);
48
49         System.out.printf("%nMap contains:%nKey\t\tValue%n");
50
51         // generate output for each key in map
52         for (Character key : sortedKeys) {
53             System.out.printf("%-10s%-10s%n", key, map.get(key));
54         }
55
56         System.out.printf(
57             "%nsize: %d\nisEmpty: %b%n", map.size(), map.isEmpty());
58     }
59 }

```



```
运行: Main x
D:\Java\bin\java.exe "-javaagent:D:\JetBrains\IntelliJ
Enter a string:
Java is a popular programming language

Map contains:
Key      Value
a         7
e         1
g         4
i         2
j         1
l         2
m         2
n         2
o         2
p         3
r         3
s         1
u         2
v         1

size: 14
isEmpty: false

进程已结束,退出代码0
```

题目 4: 编写一个程序, 它从用户处获得一个整数输入, 然后判断它是否为质数。如果不显示该数的全部质因子。质数的因子是 1 和自身。不为质数的数, 都具有唯一的因子分解形式。例如 54 的因子为 2、3、3、3, 这些值相乘的结果是 54。因此对于 54, 输出因子应为 2 和 3.需在程序中使用 Set。

(一) 实验环境 (集成开发环境、jdk 版本、字符编码等)

集成开发环境: IntelliJ IDEA

jdk 版本: 17.0.5

字符编码：ASCII

（二）实现过程（**本部分为主要评分依据**，请描述解题思路，比如总共设计几个类，各个类的用途、成员、主要方法等及其之间调用关系等）

1. 新建一个 `int` 型 `n` 用于获取用户输入的整数，建立一个 `double` 型的 `sqrt_n` 用于储存 \sqrt{n} 。
2. 新建一个 `Integer` 型的 `set` 用于储存其除了 1 与自身外的质因子。
3. 通过 `for` 循环 `i` 从 2 开始到 \sqrt{n} ，若 `n` 能整除 `i` 则将其除到不能再整除为止，并将 `i` 存入 `set`。
4. 若最后 `set` 的长度为 0 则该数为质数，否则不是，并输出 `set` 的所有内容。

（三）过程截图（**本部分为主要评分依据**，一张全屏截图（必须）、若干运行结果展示图（可选），主要代码（可选））


```
Main.java x
1  import java.util.*;
   0 个用法
2  ▶ public class Main {
   0 个用法
3  ▶  public static void main(String[] args) {
4      Scanner scanner = new Scanner(System.in);
5      System.out.println("请输入一个整数:");
6      int n = scanner.nextInt();
7      double sqrt_n=Math.sqrt(n);
8      Set<Integer> set = new HashSet<>();
9      for(int i=2;i<=sqrt_n;i++){
10         while(n%i==0){
11             set.add(i);
12             n=n/i;
13         }
14     }
15     if(set.size()==0)System.out.println("该数是质数!");
16     else{
17         System.out.println("该数不是质数, 质因子:");
18         for(int i:set)System.out.print(" "+i);
19     }
20 }
21 }
```

运行: Main x

D:\Java\bin\java.exe

请输入一个整数:

54

该数不是质数, 质因子:

2 3

进程已结束, 退出代码0

运行: Main x

D:\Java\bin\java.exe

请输入一个整数:

23

该数是质数!

进程已结束, 退出代码0

三、实验总结与心得记录

本部分根据实验过程的所得所想描述，记录可供以后复习回看 {可以记录调试过程遇到的问题，自己哪些知识点掌握不够，设计是否有缺陷（比如耗时？耗内存？）是否有亮点，是否有精妙的算法，或者设计模式的应用，可吐槽，也可与其他语言作适当对比。}（本部分不作为平时评分依据）

备注：

建议附带代码提交的方式：导出工程压缩包。

平时实验成绩以考查参与度为主，所有实验要求自己完成，一旦发现抄袭或者其他投机取巧，取消所有平时成绩