Linnéuniversitetet
Kalmar Växjö

Linnaeus University

1DV532 – Starting out with Java
Assignment 3 Report

Student: Evan Huynh
Student ID: eh223im@student.lnu.se

Table of Contents

Report

Exercise 1: Dice

In this exercise, the main idea is to create two variables for two dice then another variable to sum them up. After each turn, we add 1 into each value of the array recorded result and export it in the appropriate form.

Exercise 2: Person

This exercise requires a lot of classes and inheritances. For each class we make two constructors, a no-argument and a full-argument constructor. For each of the parent, there will be a special method to export to all parent's field to ArrayList (namely, *callMeWhenYouNeedSomeHelp* and *callMeWhenYouNeedSomeMoreHelp*). All *toString* methods have similar algorithm: call the parents' arraylist, add their own fields and export the arraylist to String using its' own *toString* method. This way we can keep exporting all the fields simple and avoid rewriting multiple lines of code.

For the PersonMain class, it is self-explanatory, an implementation of the Person class.

Exercise 3: Message

The idea is similar to the exercise 2 with some extensions. At the main Message class, it contains only *text* field and some related methods. The *encode* method, however, is stored Message since this method affects mostly the sending message, not any other field. Of course, we can overload this method with a string as parameter to encode other fields in the child classes such as SMS and Email, but it does not really matter.

Since this one is simpler than exercise 2, *callMeWhenYouNeedSomeHelp* only returns the text field (only in parents' class), all the *toString* methods will add subsequence fields to the ArrayList and export it using its own *toString* method for the best-looking output.

Of course, MessageMain is a test to see that Message works properly.

Exercise 4: Number

This exercise requires file reading and writing plus data processing in arrays, therefore we will use the blind array method:[1] Create an empty array (length 0), if the file still have next, extend the array one more slot, copy all data from the previous array to the new array, assign the new array to the previous one and add the data to it, repeat until the file is empty. By this algorithm, we do not have to worry that there is an empty 0 in the array (0 in array but not in the file), therefore not changing the result of the average and standard deviation calculation.

To calculate average and standard deviation, it simply follows the algorithm in the exercise, given details by the instruction document.

Exporting the results is simply write everything to a string, println to console using System.out

[1] There should be a scientific name for this algorithm

and to file using PrintWriter.

Exercise 5: Name

Since this exercise contains two files, one for girls and one for boys, there will be two different Scanner and FileInputStream classes to read them. Each file will be stored in a 2-D object array, the array extension is similar to exercise 4 (empty array first, extend it if it has next, copy everything from the previous array then add newest value to the end). After that each element will be copied to their corresponding arrays (a0 for male-ranking and a1 for number of boy names, c0 and c1 for female-ranking and number of female ranking). After that it is just some output decorations in the console.

Note

Since the file is stored in the same folder as the Java file, in order to read the file, we need to change the file location in the String "dir". In this case, entering a full directory path to the folder containing those files are enough since the program will read (and overwrite) those files.

It is obvious that you should not simply throw Exception in the *main* method since you will not be able to catch it and it should be considered a bad practice.

Source code

Here is my source codes for all the exercise.

1

Dice.java

```
package eh223im_assign3;

import java.util.Random;

public class Dices {
    public static void main(String[] args) {
        int a1, a2, a3;
        Random r = new Random();
        int[][] b = new int[11][2];
        for (int i = 0; i<b.length;i++) {
            b[i][0] = i+2;
        }

        for (int i = 0; i<10000; i++) {
            a1 = r.nextInt(6) + 1;
            a2 = r.nextInt(6) + 1;
            a3 = a1 + a2;
            b[a3 - 2][1] += 1;
        }

        for (int i = 0; i<b.length;i++) {
            System.out.println(b[i][0]+"\t"+b[i][1]);
        }
    }
}
```

2

Person.java

```
package eh223im_assign3;

import java.util.ArrayList;

public class Person {
    private String name;
    private String address;
    private String phonenumber;
    private String emailaddress;

    public Person(String name, String address, String phonenumber, String
emailaddress) {
        this.name = name;
        this.address = address;
        this.phonenumber = phonenumber;
        this.emailaddress = emailaddress;
    }

    public Person() {
```

```
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }

    public String getPhonenumber() {
        return phonenumber;
    }

    public void setPhonenumber(String phonenumber) {
        this.phonenumber = phonenumber;
    }

    public String getEmailaddress() {
        return emailaddress;
    }

    public void setEmailaddress(String emailaddress) {
        this.emailaddress = emailaddress;
    }

    ArrayList callMeWhenYouNeedSomeHelp() {
        ArrayList al = new ArrayList();
        al.add(getName());
        al.add(getAddress());
        al.add(getPhonenumber());
        al.add(getEmailaddress());
        return al;
    }

    public String toString() {
        return callMeWhenYouNeedSomeHelp().toString();
    }
}

class Student extends Person {
    private String classStatus;

    public Student (String name, String address, String phonenumber,
```

```
String emailaddress, String classStatus) {
        super(name, address, phonenumber, emailaddress);
        this.classStatus = classStatus;
    }

    public Student() {

    }

    public String getClassStatus() {
        return classStatus;
    }

    public void setClassStatus(String classStatus) {
        this.classStatus = classStatus;
    }

    public String toString() {
        ArrayList<String> al = callMeWhenYouNeedSomeHelp();
        al.add(classStatus);
        return al.toString();
    }
}

class Employee extends Person {
    private String dateHired;
    private int salary;

    public Employee (String name, String address, String phonenumber,
String emailaddress, String dateHired, int salary) {
        super (name, address, phonenumber, emailaddress);
        this.dateHired = dateHired;
        this.salary = salary;
    }

    public Employee() {

    }

    public String getDateHired() {
        return dateHired;
    }

    public void setDateHired(String dateHired) {
        this.dateHired = dateHired;
    }

    public int getSalary() {
        return salary;
    }

    public void setSalary(int salary) {
        this.salary = salary;
```

```
    }

    ArrayList callMeWhenYouNeedSomeMoreHelp() {
        ArrayList al = callMeWhenYouNeedSomeHelp();
        al.add(getDateHired());
        al.add(getSalary());
        return al;
    }

    public String toString() {
        ArrayList al = callMeWhenYouNeedSomeMoreHelp();
        return al.toString();
    }
}

class Faculty extends Employee {
    private String rank;
    private String officeHour;

    public Faculty(String name, String address, String phonenumber,
String emailaddress, String dateHired, int salary, String rank, String
officeHour) {
        super(name, address, phonenumber, emailaddress, dateHired,
salary);
        this.rank = rank;
        this.officeHour = officeHour;
    }

    public Faculty() {

    }

    public String getRank() {
        return rank;
    }

    public void setRank(String rank) {
        this.rank = rank;
    }

    public String getOfficeHour() {
        return officeHour;
    }

    public void setOfficeHour(String officeHour) {
        this.officeHour = officeHour;
    }

    public String toString() {
        ArrayList al = callMeWhenYouNeedSomeMoreHelp();
        al.add(rank);
        al.add(officeHour);
        return al.toString();
```

```
    }
}

class Staff extends Employee {
    private String title;

    public Staff(String name, String address, String phonenumber, String
emailaddress, String dateHired, int salary, String title) {
        super(name, address, phonenumber, emailaddress, dateHired,
salary);
        this.title = title;
    }

    public Staff() {

    }

    public String getTitle() {
        return title;
    }

    public void setTitle(String title) {
        this.title = title;
    }

    public String toString() {
        ArrayList al = callMeWhenYouNeedSomeMoreHelp();
        al.add(title);
        return al.toString();
    }
}
```

PersonMain.java

```
package eh223im_assign3;

public class PersonMain {
    public static void main(String[] args) {
        Person person = new Person("John","Los
Angeles","0123456789","john@wwe.com");
        Student student = new Student("Cena","New
York","0518123456","cena@newyork.edu","freshmen");
        Employee employee = new Employee("Simeon","Los
Angeles","0123456789","simeon@pms.com","20130917",150000000);
        Staff staff = new Staff("Franklin","Los
Angeles","0123456789","franklin@pms.com","20130917",10000000,"CEO");
        Faculty faculty = new Faculty("Lamar","Los
Angeles","0123456789","lamar@pms.com","20130917",150000,"Vice
President","0800-1700");
        System.out.println(person.toString());
        System.out.println(student.toString());
        System.out.println(employee.toString());
        System.out.println(staff.toString());
        System.out.println(faculty.toString());
    }
}
```

3

Message.java

```java
package eh223im_assign3;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collections;

public class Message {
    private String text;

    public Message(String text) {
        this.text = text;
    }

    public String callMeWhenYouNeedSomeHelp() {
        return text;
    }

    public String encode() {
        StringBuilder sb = new StringBuilder();
        char[] temp1 = "abcdefghijklmnopqrstuvwxyz".toCharArray();
        char[] temp2 = "ABCDEFGHIJKLMNOPQRSTUVWXYZ".toCharArray();
        char[] temp3 = text.toCharArray();

        ArrayList<Character> al1 = new ArrayList<>();
        for (char i:temp1) {
            al1.add(i);
        }

        ArrayList<Character> al2 = new ArrayList<>();
        for (char i:temp2) {
            al2.add(i);
        }

        ArrayList<Character> al3 = new ArrayList<>();
        for (char i:temp3) {
            al3.add(i);
        }

        for (int i = 0; i< al3.size(); i++) {
            char j = al3.get(i);
            if (al1.contains(j)) {
                j = al1.get( (al1.indexOf(j) + 1) % 26);
            } else if (al2.contains(j)) {
                j = al2.get( (al2.indexOf(j) + 1) % 26);
            }
            al3.set(i,j);
        }

        for (Character character : al3) {
            sb.append(character);
        }
```

```
        return sb.toString();
    }

    public void setText(String text) {
        this.text = text;
    }

    public String getText() {
        return text;
    }
}

class SMS extends Message {
    private String recipientContactNo;

    public SMS(String recipientContactNo,String text) {
        super(text);
        this.recipientContactNo = recipientContactNo;
    }

    public String getRecipientContactNo() {
        return recipientContactNo;
    }

    public void setRecipientContactNo(String recipientContactNo) {
        this.recipientContactNo = recipientContactNo;
    }

    public String toString() {
        ArrayList<String> al = new ArrayList<String>();
        al.add(recipientContactNo);
        al.add(callMeWhenYouNeedSomeHelp());
        return al.toString();
    }
}

class Email extends Message {
    private String sender;
    private String receiver;
    private String subject;

    public Email(String sender, String receiver, String subject, String
text) {
        super(text);
        this.sender=sender;
        this.receiver=receiver;
        this.subject=subject;
    }

    public String toString() {
        ArrayList<String> al = new ArrayList<String>();
        al.add(sender);
        al.add(receiver);
```

```
        al.add(subject);
        al.add(callMeWhenYouNeedSomeHelp());
        return al.toString();
    }

    public String getSender() {
        return sender;
    }

    public void setSender(String sender) {
        this.sender = sender;
    }

    public String getReceiver() {
        return receiver;
    }

    public void setReceiver(String receiver) {
        this.receiver = receiver;
    }

    public String getSubject() {
        return subject;
    }

    public void setSubject(String subject) {
        this.subject = subject;
    }
}
```

MessageMain.java

```
package eh223im_assign3;

public class MessageMain {
    public static void main(String[] args) {
        Message m = new Message("Hello World");
        SMS sms = new SMS("0123456789",m.getText()+" from SMS.");
        System.out.println(sms.toString());
        System.out.println(sms.encode());
        Email e = new Email("alice@gmail.com","bob@gmail.com","Hello
Bob",m.getText()+" from email.");
        System.out.println(e.toString());
        System.out.println(e.encode());
    }
}
```

4

Numbers.java

```
package eh223im_assign3;

import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.PrintWriter;
import java.util.Scanner;
```

```
public class Numbers {
    public static void main(String[] args) throws Exception {
        System.out.print("Enter directory path in which number.txt is
located and analysis.txt is going to be written: ");
        String dir = new Scanner(System.in).next();
        FileInputStream fis = new FileInputStream(dir+"/numbers.txt");
        Scanner s = new Scanner(fis);
        int[] a = new int[0];
        while (s.hasNext()) {
            int[] b = new int[a.length+1];
            System.arraycopy(a,0,b,0,a.length);
            a=b;
            a[a.length-1] = s.nextInt();
        }

        int c = 0;
        for (int i = 0; i < a.length; i++) {
            c+=a[i];
        }
        double d = (double) c/a.length;

        FileOutputStream fos = new FileOutputStream(dir+"/analysis.txt");
        PrintWriter pw = new PrintWriter(fos);
        String o1 = "Average: "+d;
        pw.println(o1);
        double e = 0;

        for (int i = 0; i < a.length; i++) {
            e += Math.pow((a[i] - d),2);
        }
        e /= a.length;
        e = Math.sqrt(e);
        String o2 = "Standard deviation: "+e;
        pw.println(o2);

        System.out.println(o1);
        System.out.println(o2);

        s.close();
        fis.close();
        pw.close();
        fos.close();
    }
}
```

5

Names.java

```
package eh223im_assign3;

import java.io.FileInputStream;
import java.util.Objects;
import java.util.Scanner;

public class Names {
    public static void main(String[] args) throws Exception {
        System.out.print("Enter directory path in which boynames.txt and
girlnames.txt are located: ");
        String dir = new Scanner(System.in).next();
        FileInputStream fisB = new FileInputStream(dir+"/boynames.txt");
        Scanner sB = new Scanner(fisB);
        Object[][] a = new Object[0][2];
```

```
        while (sB.hasNext()) {
            Object[][] b = new Object[a.length + 1][2];
            System.arraycopy(a, 0, b, 0, a.length);
            a = b;
            a[a.length-1][0] = sB.next();
            a[a.length-1][1] = sB.nextInt();
        }

        FileInputStream fisG = new FileInputStream(dir+"/girlnames.txt");
        Scanner sG = new Scanner(fisG);
        Object[][] c = new Object[0][2];
        while (sG.hasNext()) {
            Object[][] d = new Object[c.length + 1][2];
            System.arraycopy(c, 0, d, 0, c.length);
            c = d;
            c[c.length-1][0] = sG.next();
            c[c.length-1][1] = sG.nextInt();
        }
        sB.close();
        fisB.close();
        sG.close();
        fisG.close();

        String[] a0 = new String[a.length];
        for (int i = 0; i< a0.length; i++) {
            a0[i] = Objects.toString(a[i][0]).toLowerCase();
        }
        int[] a1 = new int[a.length];
        for (int i = 0; i< a1.length; i++) {
            a1[i] = Integer.parseInt(Objects.toString(a[i][1]));
        }
        String[] c0 = new String[c.length];
        for (int i = 0; i< c0.length; i++) {
            c0[i] = Objects.toString(c[i][0]).toLowerCase();
        }
        int[] c1 = new int[c.length];
        for (int i = 0; i< c1.length; i++) {
            c1[i] = Integer.parseInt(Objects.toString(c[i][1]));
        }

        Scanner s = new Scanner(System.in);
        System.out.print("Enter name: ");
        String ss = s.next();
        ss=ss.toLowerCase();
        int rB = -1;
        int rG = -1;
        for (int i = 0; i<a0.length; i++) {
            if (ss.equals(a0[i])) {
                rB = i+1;
            }
        }
        for (int i = 0; i<c0.length; i++) {
            if (ss.equals(c0[i])) {
                rG = i+1;
            }
        }
        ss=ss.substring(0,1).toUpperCase()+ss.substring(1).toLowerCase();
        if (rB == -1) {
            System.out.println(ss + " is not ranked among the top 1000
boy names.");
        } else {
```

```
            System.out.println(ss + " is ranked "+rB+" in popularity
among boys with "+a1[rB]+" naming.");
        }
        if (rG == -1) {
            System.out.println(ss + " is not ranked among the top 1000
girl names.");
        } else {
            System.out.println(ss + " is ranked "+rG+" in popularity
among girls with "+c1[rG]+" naming.");
        }
    }
}
```