成绩	

重庆邮电大学

实验报告

2020-2021 学年第 2 学期

计算机科学导论

（第 8 次试验）

班级：34082003

姓名：黄凯升

学号：2020215138

指导老师：许汀汀

课程名称：计算机科学导论

实验时间：2021 年 5 月 27 日

实验地点：综合实验大楼 A511/A512

1 实验名称

Exceptions and I/O Streams

2 实验目的

- Be able to write code that handles an exception
- Be able to write code that throws an exception
- Be able to write a custom exception class

3 实验内容

Task#1 Writing a Custom Exception Class

Create an exception class called SocSecException according to UML, whose constructor will call the superclass constructor. It will set the message associated with the exception to “Invalid social security number” concatenated with the error string. And Create a driver program called SocSecProcessor.java. This program will have a main method and a static method called `isValid` that will check if the social security number is valid.

Task #2 Writing Code to Handle an Exception

The task has two parts.

The first part is in the main method. It should read a name and a read a name and social security number from the user as Strings. It should contain a try-catch statement, which tries to check if the social security number is valid by using the method `isValid`. If valid, it prints the name and social security number. If a SocSecException is thrown, it should catch it and print out the name, social security number entered, and an associated error message indicating why the social security number is invalid. And a loop to allow the user to continue until the user indicates that they do not want to continue.

The second part is in the static `isValid` method. It can throw a SocSecException. It returns true if the SSN is valid. It checks it and throws a SocSecException with appropriate message. By checking the length of string, the position of dashes and the non-digit characters in the SSN.

4 实验方法（原理、流程图）

The development environment is:

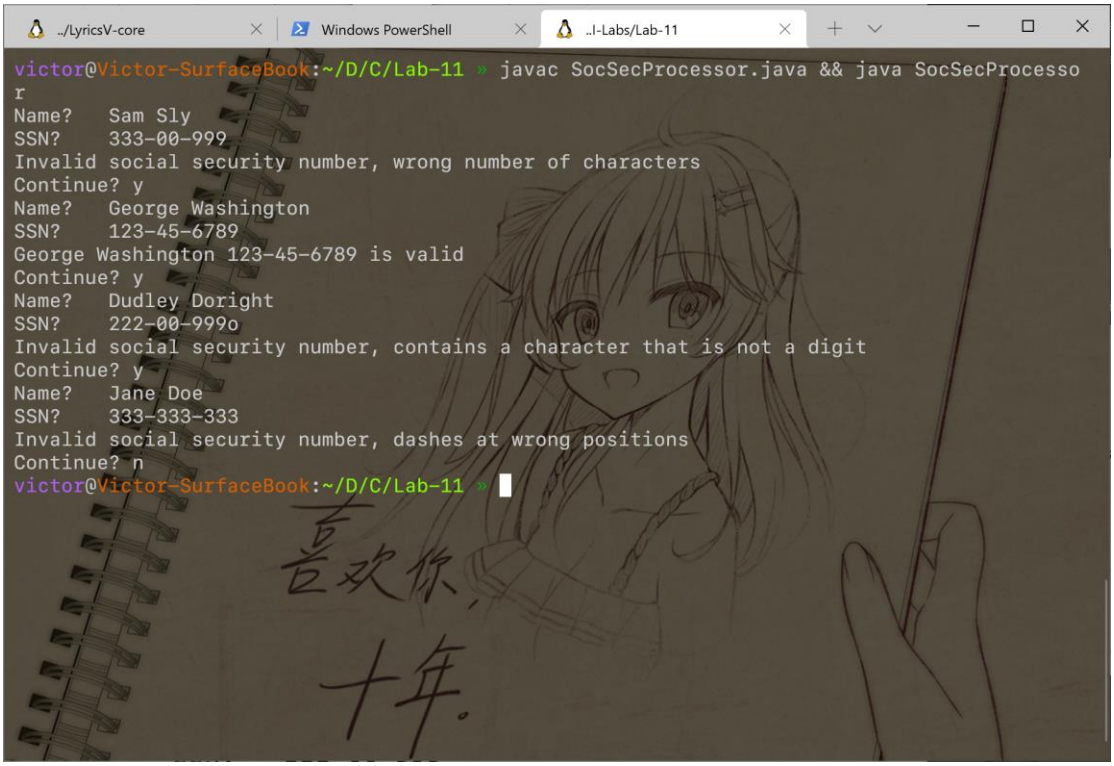
- OS: Ubuntu 20.04.2 LTS on Windows 10 (WSL1, Kernel build 19041)
- IDE/Editor: Visual Studio Code
- Java Runtime: OpenJDK 14.0.2 (build 14.0.2+12-Ubuntu-120.04)

For Task #1, we just create the class SocSecException which extends Exception, whose constructor calls super constructor and concentrate the string, and create the driver class according to UML.

For Task #2, in method `isValid`, we check the length of SSN first. And use a loop and method `String.charAt` to get the character of a specified position in string, to detect an inappropriate dash. Finally, we use `Character.isDigit` to detect any non-digit or dash characters. If detected invalid case, we throw a new SocSecException. In the main method, once we use the method `isValid`, we should wrap it in a try statement, which has a following catch statement to catch the SocSecException thrown.

5 实验结论

The lab has finished successfully. The programs can completely achieve all goals. Here is the screenshot.



6 实验体会和收获

In this lab we practiced exceptions, which is very important in modern programming language. Without exception, we have to write a bunch of if statements to deal with problems in our program, just like C and Golang.

7 程序代码

```
import java.util.Scanner;

class SocSecException extends Exception {
    public SocSecException(String error) {
        super("Invalid social security number, " + error);
    }
}

public class SocSecProcessor {
    public static void main(String args[]) {
        Scanner in = new Scanner(System.in);
        boolean continueRun = true;
        while (continueRun) {
            System.out.print("Name?   ");
            String name = in.nextLine();
            System.out.print("SSN?    ");
            String ssn = in.nextLine();
            try {
                if (SocSecProcessor.isValid(ssn))
                    System.out.printf("%s %s is valid\n", name, ssn);
            } catch(SocSecException e) {
                System.err.println(e.getMessage());
            }
            System.out.print("Continue? ");
            String continueString = in.nextLine();
            if (continueString.length() > 0 && continueString.toLowerCase().charAt(0) == 'y')
                continueRun = true;
            else
                continueRun = false;
        }
    }

    public static boolean isValid(String ssn) throws SocSecException {
        if (ssn.length() != 11) {
            throw new SocSecException("wrong number of characters");
        }
        for (int i = 0; i < ssn.length(); i++) {
            char ch = ssn.charAt(i);
            if (i == 3 || i == 6) {
                if (ch != '-') {
                    throw new SocSecException("dashes at wrong positions");
                }
            }
            if (!Character.isDigit(ch) && ch != '-') {
                throw new SocSecException("contains a character that is not a digit");
            }
        }
        return true;
    }
}
```