

Northeastern Illinois University
CS303A, Object-Oriented Programming and Data Structures
Homework: Exception Handling

Problem 1:

Complete the **sumOfDiagonals** method in **SumOfDiagonals.java** to do the following:

- The method takes a 2D **String** array **x** as a parameter and returns no value.
- The method should calculate and print the sum of the elements on the major diagonal of the array **x**.
- In order to have a major diagonal, the array passed into the method should be a square (***n-by-n***), if it's not a square your program should handle that situation by throwing an exception. (**Do Not worry about ragged arrays**)
- If the array is a square, but there is a non-integer value on the major diagonal, your program should handle that situation by throwing an exception.
- When handling the exceptions, be as specific as you can be, (**i.e. Do Not just use the Exception class to handle all exceptions in one catch block**).
- Make the proper calls to the **sumOfDiagonals** method from the main method to test your **sumOfDiagonals** method on all the String arrays provided in the main method.
- Sample runs provided on the next page.

Argument Array x	Output
<pre>{{"2", "3", "4"}, {"5", "6", "7"}, {"8", "9", "10"}}</pre>	18
<pre>{{"1", "2", "3"}, {"5", "6", "7"}, {"9", "10", "11"}, {"13", "14", "15"}};</pre>	Array not a square
<pre>{{"1", "2", "3", "4"}, {"5", "6", "7", "8"}, {"9", "10", "11", "12"}, {"13", "14", "15", "pp"}};</pre>	Not a valid integer
<pre>{{"1", "2", "3", "4"}, {"5", "6", "7", "8"}, {"9", "10", "11", "12"}, {"13", "14", "15", "16"}};</pre>	34

Instructions:

- Download the needed files and look for SumOfDiagonals.java.
- Complete SumOfDiagonals.java.
- Place SumOfDiagonals.java in a folder named YourName_HW9

Problem 2: What is the exact output of the following program, you should trace this program by hand, and save output as .pdf then add it to YourName_HW9 folder.

```
public class Tracing
{
    public static void main(String[] args)
    {
        String str = "HELLO-WORLD!!";
        String[] a = {"3", "4", "35", "0", "27", "0", "0", "90", "40", "55", "68"};
        String[] b = {"12", "", "0", "23", "w", "0", "f", "0"};

        for(int i = 0; i < a.length + 1; i++)
        {
            try
            {
                if(i < a.length - 2)
                {
                    int x = Integer.parseInt(a[i + 1]);
                    int y = Integer.parseInt(b[i]);
                    System.out.println(x / y);
                    System.out.println(str.charAt(i * 10));
                    System.out.println("No exception this time.");
                }
                else if(i >= b.length)
                {
                    int x = Integer.parseInt(b[i - 2]);
                    int y = Integer.parseInt(a[i]);
                    System.out.println(x / y);
                }
            }
            catch(ArithmeticException e)
            {
                System.out.println("Invalid Arithmetic Operation");
            }
            catch(ArrayIndexOutOfBoundsException e)
            {
                System.out.println("Array Out Of Bounds");
            }
            catch(StringIndexOutOfBoundsException e)
            {
                System.out.println("String Out Of Bounds");
            }
            catch(NumberFormatException e)
            {
                System.out.println("hmmm, can't convert.");
            }
        }
    }
}
```

Problem 3:

Download the needed files and look for the data.txt file. This file should have multiple lines, each line has first name, last name followed by multiple integers.

- Create a class named `MaximumOfEachLine` with a main method.
- Read the data from the "data.txt" file.
- Find the maximum integer in each line.
- Display the first name and the last name followed by the maximum integer found in that line.
- Make sure to have the proper import statements.
- Your code must handle `FileNotFoundException`.
- If the file is missing, your program should display "File Not Found".
- If you did all the above correctly, this is the output you should get:

Samantha Johns	Max: 99
Michael Smith	Max: 100
John Michael	Max: 92
Mary Hernandez	Max: 92
George Johnson	Max: 93
Sara Anderson	Max: 83
Susan John	Max: 84
Mark Smith	Max: 69

General Instructions:

- No hard copies will be collected.
- Do not send your files through the email!
- You should submit your work by the due date, **No** extensions will be given. (See syllabus for late homework policy).
- **DO NOT** turn in multiple files, only one .zip file.

What to turn in:

There should be two **.java** file(SumOfDiagonals.java & MaximumOfEachLine.java) and one **.pdf** file. put the file into a zip file and name it <YourFirstName_YourLastName>.zip, submit the zip file into the Dropbox on D2L.

How to zip multiple files?

On Windows: Select all the files > right click > Send to > Comprised File

On Mac: Select all the files > Click/Tap with two fingers > Compress Items