Overview

This week we will read more than one file as input, and also create an output file. More specifically, we will take in two matrices and multiply them. The result will then be written to a file. Most of the code and logic you need to finish this lab is presented in lecture this week, and also included in this lab assignment. Your job is to combine each portion to do the job correctly.

Before you get started, read chapters 5.7 and 5.9. Answer activities 5.7.3 and 5.9.2 in a Word or text file. Also, answer the Assessment questions as you encounter them in the next section. The prompts for answering assessment questions are placed immediately following the material to which the listed questions relate.
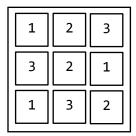
Getting Started

After following the import instructions in the assignment page, you should have a Java project in Eclipse titled Lab 21_13. This PDF document is included in the project in the **doc** directory. The Java files you will use in this lab are in the **src/matrix** directory.

Part 1: Multiplying Matrices

The file format we will use has 2 types of information. The first is the dimensions of the array. The second is the actual values inside the array. In order to encode this information, we create a structured format that is easy to understand.

The first line will contain 2 numbers. The first is the # of rows and the second is the # of columns. Then each row of values will be printed in corresponding columns in a tab-delimited manner.

The 3×3 matrix shown above would be translated to the following file:

```
3    3
1    2    3
3    2    1
1    3    2
```

We will introduce is the new keyword `null` which means an 'empty' object to which a pointer can point. We can also check if a pointer is valid by checking it against the object, `null`.

Task 1: Analyze code
Analyze the following code and answer the assessment questions that follow:

```
1  public static int[][] proc(String filename) {
2
3      int[][] arr = null;
4      try{
```

```
5          Scanner sc = new Scanner(new FileReader(filename));
6          int row = sc.nextInt();
7          int column = sc.nextInt();
8          arr = new int[row][column];
9
10         for (int i = 0; i < row; i++){
11             for (int j = 0; j < column; j++){
12                 arr[i][j] = sc.nextInt();
13             }
14         }
15         sc.close();
15     }catch(NoSuchElementException e){
16         System.out.println(e);
17     }catch(FileNotFoundException e){
18         System.out.println(e);
19     }
20     return arr;
21 }
```

[Answer assessment questions 1a – 1g]

Task 2: Fill-in `readMatrix`
Fill in the code to read in the values from the file whose name is specified by the given string parameter inside the `readMatrix` method of `MatrixMultiply.java` [Hint: see Task 1].

Task 3: Analyze code
Analyze the following code sample and answer the assessment questions that follow:

```
String filename = "Result.txt";
try {
    FileWriter output = new FileWriter(filename);
    String ostr = "";
    for (int i = 0; i < arr2D.length; i++) {
        for (int j = 0; j < arr2D[0].length; j++) {
            System.out.print(ostr = (arr2D[i][j] + "\t"));
            output.write(ostr);
        }
        System.out.println();
        output.write("\r\n"); // Carriage return
    }
    output.close();
}catch(Exception e){
    System.out.println(e);
}
```

[Answer assessment questions 2a – 2c]

Task 4: Fill-in `writeMatrix`
Fill in the code to write the values after multiplying the two matrices [Hint: See Task 3]. You will have 3 input combinations to test for this lab. Each output should be saved to a different file. They should be named `Result1.txt`, `Result2.txt` and `Result3.txt`.

`Result1.txt` is `matrix1.txt` and `matrix2.txt` multiplied.
`Result2.txt` is `matrix2.txt` and `matrix3.txt` multiplied.
`Result3.txt` is `matrix2.txt` and `matrix4.txt` multiplied.

You may do this by simply renaming the output file the program produces. Or you may modify filenames in code for different runs of the program. Other techniques are also possible.

Part 2: Fill in Survey

Please complete the survey in `survey.txt` provided in the project's **doc** directory.

Part 3: (Assessment) Logic Check and Level of Understanding

1) For the `proc` method shown in Task 1, answer the following:
   a. What is the return type of the method, `proc`? What line in the above code can you find this information?
   b. What is line 3 doing? (Answer in terms of variable name, type and initial value)
   c. Why do we need to use `try` in line 4?
   d. What line(s) read in the dimension of the matrix?
   e. What is line 8 doing?
   f. Why is there no while-loop in this method, and why we can use a for-loop?
   g. How many times does line 12 get executed? (Answer in terms of an expression involving # rows and # of columns)
2) Answer the following questions concerning the code given in Task 3:
   a. What does `arr2D.length` mean?
   b. What does `arr2D[0].length` mean?
   c. What does `arr2D[i][j] + "\t"` create?

What to hand in

When you are done with this lab assignment, submit all your work through Catcourses.

Before you submit, make sure you have done the following:
- Included answers to activities 5.7.3 and 5.9.2, and Assessment questions (1 – 2) in a Word or text file named `Part3`
- Attached filled in `MatrixMultiply.java`, and the results files `Result1.txt`, `Result2.txt` and `Result3.txt`, the filled in survey `survey.txt`, and `Part3` files
- Filled in your collaborator's name (if any) in the "Comments…" text-box in the submission page