**Lab 3:** **Multi-Dimensional Arrays – Three Parts**

1. PRE-LAB: The code for the first program (Array2DTest) is provided below; add the code to also calculate column totals; compile and run the program. Turn in source and output.

// Array2DTest.java

// simple use of two-dimensional arrays

public class Array2DTest

{

public static void main (String[] args)

{

int [] [] A = new int [2] [5]; // 2 rows, 5 columns

// fill each cell with the sum of its row and column numbers

for (int i = 0; i < 2; i++)

{

for (int j = 0; j < 5; j++)

{

A[i] [j] = i + j;

}

}

int [] [] B = {{1,2,3,4,5}, {6,7,8,9,10}}; // 2 rows, 5 columns

System.out.println (“A.length = “ + A.length);

System.out.println (“B.length = “ + B.length);

System.out.println (“Number of columns in A is “ + A[0].length);

System.out.println (“Number of cells in A is “ + A.length \* A[0].length);

// compute row sums for B

for (int i = 0; i < B.length; i++)

{

int sum = 0;

for (int j = 0; j < B[i].length; j++)

{

sum = sum + B[i] [j];

}

System.out.println (“The sum of row “ + i + “ is “ + sum);

}

// compute column sums for B here

}

}

1. Answer these three review questions on a separate sheet of paper and attach to lab.
   1. Code a segment of a program to assign the value of 5 to every position in the array A declared below:

int [][] A = new int [20][12];

* 1. Given a 2-D array called B, what expression indicates the number of rows\_\_\_\_\_\_\_\_\_\_\_\_\_? The number of columns\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_?
  2. Draw a diagram that would represent the table with the values given in the declaration below.

int [][] A = {{3,-2,4,0} , {0,8,10,12} , {5,5,1,2}};

1. LAB: Create a program that will read in two values, R and C, and create an array of that size. Clear the array to all zeros using two for loops. Using a while loop that tests for a sentinel, read the rest of the file and place each value in the row and column given. End of file is determined by detecting an entry of -1 -1 -1. (Do a preliminary read of r, c, and value outside of the while loop and again at the bottom of the while loop.) Print the array as shown in handout, again using two for loops. Compile and run this program with the data on handout (type data as shown into a new kate file and save as input.txt) and with one set of data that you create (See step 4 below). Turn in the source and 2 outputs. Also turn in a copy of the data file that you create.
2. Instead of typing in the input for **each** run, read from a text file for each run by doing the following:

Add throws IOException to the main method heading line

Type the data into kate or similar text editor and name it, i.e. input.txt

Use: Scanner scan = new Scanner (new File((“input.txt”));

Also add: import java.io.\*; // necessary for File command

You still need: import java.util.Scanner;

Now all your Scanner methods will read from the file – no prompts needed.

5. Data:

6

4

0 0 45

1 1 9

2 2 569

3 2 17

2 3 -17

5 3 9999

-1 -1 -1

1. Turn in:
   1. Source and output from the first part (row and column sums)
   2. Three questions - answered
   3. Source from second part – just one source even though you ran it twice
   4. Two outputs from second part of lab
   5. Copy of your test data file – the one you made up