**COMP 132 - Homework # 6**

**Array Collections**

Note: As usual, object diagrams can be hand-drawn and scanned or photographed, then pasted into your solution.

1. Draw an object diagram of the array that would be created by each of the following array declarations. Be sure to include the array variable and draw all references as arrows.

 a. int[][] arrayOne = new int[2][3];

 b. int[][] arrayTwo = new int[3][];

 arrayTwo[0] = new int[4];

 arrayTwo[1] = new int[2];

 arrayTwo[2] = new int[3];

 c. int[][] arrayThree = new int[3][3];

 arrayThree[2] = new int[4];

2. Consider the following declaration of a ragged array:

 int[][] myArray = new int[5][];

 myArray[0] = new int[5];

// omitted code that creates rows 1…3

 myArray[4] = new int[3];

Given the above declaration, write Java statements that will perform each of the following tasks. You should assume that rows 1…3 are sufficiently long such that all of the array indices needed are valid.

a. Print the length of the 2nd row of myArray. Recall that array indices begin with 0, so the 2nd row of the array is at index 1.

 b. Set the element in the 3rd row, 4th column of myArray to be 7

c. Set the element in the last column of the last row of myArray to be twice the element in the first column of the first row.

d. Set every element of myArray to be twice the sum of its row and column indices.

e. Swap the 2nd and 3rd rows of myArray. Solve this problem using the fact that a 2d array is really a 1d array of reference to the 1d arrays for the rows. Do not create any new arrays or copy the integers contained in any of the arrays. You may however find it useful to create temporary references to existing arrays.

3. Give Java statements that will create a triangular ragged array with 100 rows. Note: In a triangular array the first row has 1 column, the 2nd row has 2 columns and so on. Hint: Use a for loop!

4. For each of the following snippets of code, draw an object diagram that shows the shape and contents of the array arr after the code is executed.

 a. **int**[] arr = **new** **int**[5];

arr[1] = 3;

arr[arr[1]] = 5;

arr[0] = arr[1+2];

arr[2] = arr[4];

arr[4] = 7;

b. **int**[][] arr = **new** **int**[2][2];

arr[0][0] = 5;

arr[1][0] = 7;

**int**[] arr2 = **new** **int**[3];

arr2[0] = arr[0][0];

arr[1] = arr2;

arr2[2] = 9;

arr[1][1] = 4;