**CSc 110 Lab 8**

**Objectives:**

* Review of paper pseudo/coding exercises.
* Pass by reference review
* 2D Arrays
* Help with assignment.

**Review**

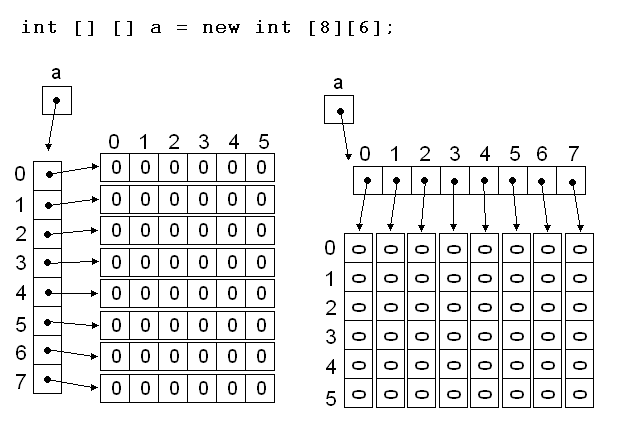
On-paper conceptual problem solving, using arrays: [Possible Solutions](http://webhome.csc.uvic.ca/~csc110l/2012_5/Lab8/arraySolns.html).

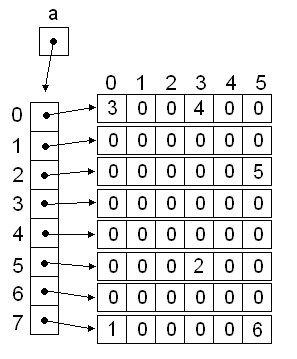
**EXERCISES**

**Example - Pass-by-Value vs Pass-by-Reference**

* Save [this program](http://webhome.csc.uvic.ca/~csc110l/2012_5/Lab8/code/Sample.java).
* Compile and run it.
* Look at the code and see what values are being printed.
* Answer these questions:
  1. What is the initial value of myArray[0], myInt and myArray[1]?
  2. What is the final value (at the end of the program) of myArray[0], myInt and myArray[1]?
  3. What is the scope of 'a'?
  4. What is the type of the variable: myArray?
  5. What is the type of: myArray[0]?
  6. What is the type of: myInt?
  7. Would this compile: passByValue(myArray) Why or why not?

***2D Arrays Quick Introduction***

Here is a two-dimensional array of primitives in memory.  
  
You can think about the first dimension as either the rows or columns of a matrix. As long as you are consistent in your ordering. That is, don't define an array like 'a' (in the figure) and then try to access an out-of-bounds index: a[3][7]. This will crash your program because 'a' is 8 by 6, in that order.

**How would you access the following ints: 1, 2, 3, 4, 5 and 6 in the array below?**  


***Exercise 1*** Demo program (define, initialize, search)

Download the program [TwoD.java](http://webhome.csc.uvic.ca/~csc110l/2012_5/Lab8/code/TwoD.java) and the data file [grades.txt](http://webhome.csc.uvic.ca/~csc110l/2012_5/Lab8/code/grades.txt) into the same folder.  Examine.

In this exercise you should first examine the program downloaded.  Note there are 2 two dimensional arrays defined. One is hard coded and the other will be read from a file.

In java a two dimensional array is in fact an array of arrays, a fact that you can ignore, for the most part, when the row length is the same for every row.  You can ignore that fact all together, when the array is square i.e. 3X3 or 4X4 etc.  Otherwise during coding you must be aware of the array of array nature of java 2D arrays in order to determine length of a given array as well as the number of them.

For example:

The numbers in the grades 2D array represent sets of 10 scores for 20 students.  All 200 scores taken together are stored in a 2D array.  Any one set of grades is a single dimension array with 10 scores.  And any one value is an integer.  These 3 things can be referenced syntactically in the following way:

grades02 //a 2D array of scores

grades02[2] //a 1D array representing the 10 scores of the 3rd student

grades02[2][2] //The 3rd score of the 3rd student

**additional syntactical references:**

grades02.length //represents the number of students

grades02[n].length //represents the number of scores for the nth student

**therefore to traverse a 2D array:**

for (int r = 0; r < a.length; r++)

{

for (int c = 0; c < a[r].length; c++)

{

<visit the current element> // comment 1

}

}

/\* comment 1

'visit' can mean print, find, check to see if the item is lesser than previous or greater,

sum, ... or any operation you may wish to perform on the 2D array a, or using the

2D array a.

\*/

Now given the code fill in the methods one at a time and test.

***Exercise 2***(still in TwoD.java)

Add a method printHard that given a 2D array prints out the values in the other order: through the different students instead of across the grades for one student at time.  Think about this.  You want a nested for loop that visits a[0][0], a[1][0], a[2][0] ... a[0][1], a[1][1], a[2][1] ... ... a[17][9], a[18][9], a[19][9].  It often helps to have paper drawings of smaller arrays say 3X3 or 5X5, when working out your logic.

**If you are using the 2d-array read in from grades.txt, the output should look like this**:  
7 7 8 6 10 10 8 10 5 10 5 8 5 9 5 9 5 9 10 10  
6 9 7 7 6 9 6 9 6 8 6 5 5 6 7 9 5 5 7 9  
10 6 9 10 7 6 9 6 9 7 7 10 5 5 8 10 10 10 7 8  
7 6 5 6 5 6 7 9 9 5 8 5 10 6 8 5 8 9 5 10  
10 7 8 8 6 7 6 5 8 6 9 5 7 5 5 8 5 8 8 7  
9 9 8 6 6 9 8 5 5 10 5 5 7 8 10 8 9 6 8 9  
7 6 7 10 8 6 7 5 10 5 10 9 9 6 7 5 5 10 9 7  
5 10 10 5 5 7 5 9 9 9 5 6 6 5 6 5 6 5 6 5  
6 10 10 6 7 7 7 6 6 8 9 8 10 7 5 7 7 10 10 10  
10 8 5 7 6 5 10 6 6 10 5 8 6 6 6 6 9 8 10 6

***Exercise 3***(still in TwoD.java)

a) Design a method that will calculate the average of each row and **return a 1d array of doubles** that represent those averages.

The output for the 2d-array from grades.txt should match this, *when printed*:  
7.7 7.8 7.7 7.1 6.6 7.2 7.3 7.0 7.3 7.8 6.9 6.9 7.0 6.3 6.7 7.2 6.9 8.0 8.0 8.1

The first number is the average for the first row, the second for the second row, etc.

b) Create a printArray that given an array of **doubles** prints it.

You may print in one row (separated by spaces, commas, tabs...) or in a column (separated by newlines). **The output example from part a, above, is in one row with the numbers separated by spaces.**

***Exercise 4***Battleship (May be done in pairs)

1) Create a new java file called Battleship.java. You will need to use Scanner (java.util.Scanner), and two dimensional arrays (no imports required for 2D arrays).

2) In your main method print out a welcome message and create a 2D array of Strings or chars. This will be your game board or 'ocean'.

3) A boat should be placed on the board. You can 'hard code' the position of your boat for now. An example of the board and the boat is below:  
~ ~ ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
~ b ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
This would be for a boat of size 1 and an ocean which was 7x7. The boat here could be at (1,3) or (3,1), depending on how you are printing the 2D array.

4) The board should be printed **without showing the boat's location**! Like so:  
~ ~ ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
And you should, in a loop, prompt the user to guess co-ordinates, for example:  
   Please enter a row:  
   Please enter a column:

5) Each incorrect guess should be marked on the board. For example:  
~ ~ ~ x ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
~ x ~ ~ x ~ ~  
~ ~ ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
~ ~ ~ ~ ~ ~ ~  
If the player had made three incorrect guesses for the above co-ordinates.

6) When the player guesses correctly, print a message and end the program:  
   YOU WIN!!

Feel free to add improvements as long as your program does at least as much as explained above. If your boat is larger than one space on the grid, only exit the game once all of the boat has been exposed/destroyed.

**Looking back...**

In this lab you'll have practiced and/or seen examples of:

* 2D Arrays: declaring, defining, initializing, assigning, indexing and referencing.
* Writing methods which print and edit 2d-arrays (often using for-loops)

The End