Topic 26
Two Dimensional Arrays

"Computer Science is a science of abstraction - creating the right model for a problem and devising the appropriate mechanizable techniques to solve it."

-Alfred Aho and Jeffery Ullman

Based on slides for Building Java Programs by Reges/Stepp, found at http://faculty.washington.edu/stepp/book/
2D Arrays in Java

- Arrays with multiple dimensions may be declared and used
  ```java
  int[][][] mat = new int[3][4];
  ```
- the number of pairs of square brackets indicates the dimension of the array.
- by convention, in a 2D array the first number indicates the row and the second the column
Two Dimensional Arrays

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

This is our abstract picture of the 2D array and treating it this way is acceptable.

(actual implementation is different)

```plaintext
mat[2][1] = 12;
```
int[][] mat = new int[10][12];

// mat is a reference to the whole 2d array
// mat[0] or mat[r] are references to a single row
// mat[0][1] or mat[r][c] are references to single elements
// no way to refer to a single column
// mat.length is the number of rows
// mat[0].length is the number of columns
// in row 0.
2D Array Problems

- Write a method to find the max value in a 2d array of ints
- Write a method that finds the sum of values in each column of a 2d array of doubles
- Write a method to print out the elements of a 2d array of ints in row order.
  - row 0, then row 1, then row 2 ...
- Write a method to print out the elements of a 2d array of ints in column order
  - column 0, then column 1, then column 2 ...
What is output by the following code?

```java
String[][] strTable = new String[5][8];
System.out.print(strTable.length + " ");
System.out.print(strTable[0].length + " ");
System.out.print(strTable[2][3].length());
```

A. 40 0 0  
B. 8 5 0  
C. 5 8 0  
D. 5 8 then a runtime error occurs  
E. No output due to a syntax error.
Use of Two Dimensional Arrays

- 2D arrays are often used when I need a table of data or want to represent things that have 2 dimensions.

- For instance an area of a simulation
Example of using a 2D array

- Conway's Game of Life
  - a cellular automaton designed by John Conway, a mathematician
  - not really a game
  - a simulation
  - takes place on a 2d grid
  - each element of the grid is occupied or empty
Simulation

- Select pattern from menu
- Select region in large area with mouse by pressing the control key and left click at the same time
- Select the paste button
## Generation 0

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>.</td>
<td>*</td>
<td>.</td>
<td>*</td>
<td>.</td>
<td>*</td>
</tr>
<tr>
<td>1</td>
<td>*</td>
<td>.</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td>2</td>
<td>.</td>
<td>.</td>
<td>*</td>
<td>*</td>
<td>.</td>
<td>*</td>
</tr>
<tr>
<td>3</td>
<td>.</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>.</td>
<td>*</td>
</tr>
</tbody>
</table>

* indicates occupied, . indicates empty
**Generation 1**

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>.</td>
<td>*</td>
<td>.</td>
<td>*</td>
<td>.</td>
<td>*</td>
</tr>
<tr>
<td>1</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>*</td>
</tr>
<tr>
<td>2</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>*</td>
</tr>
</tbody>
</table>

* indicates occupied, . indicates empty
Rules of the Game

- If a cell is occupied in this generation.
  - it survives if it has 2 or 3 neighbors in this generation
  - it dies if it has 0 or 1 neighbors in this generation
  - it dies if it has 4 or more neighbors in this generation

- If a cell is unoccupied in this generation.
  - there is a birth if it has exactly 3 neighboring cells that are occupied in this generation

- Neighboring cells are up, down, left, right, and diagonal. In general a cell has 8 neighboring cells
Implement a program to run the simulation

What data type do you want to use for the elements of the 2d array?
A. String
B. char
C. int
D. boolean
E. double
Do you want to use a buffer zone on the edges?

A. No
B. Yes