Announcements

- P6 due today at 6pm
- Final exam: 5/13, noon-2:30pm
- Email Kelly Patwell with your exam info if you have an exam conflict
- Please fill out course evaluation on-line
- Pick up papers from consultants at Carpenter
- Revised office/consulting hours next week
- Read announcements on course website!

Previous Lecture:
- Two-dimensional array of numbers
- Manipulating 2-d arrays

Today's Lecture:
- 2-d array of objects
- Review arrays of objects (1- and 2-d)

Reading:
- Sec 7.6

Instantiating 2-d arrays

- A 2-d array is a 1-d array of 1-d arrays
- You can create one dimension at a time:
  1. Declare a reference variable for the 2-d array
  2. Set 1st dimension (# rows)—create a 1-d array to hold the row references
  3. Set 2nd dimension (# columns) one row at a time—create the individual arrays that store the values (or object references) of interest
  4. Now can assign values (or references) into the cells of the array

Example: cubicle world

Implement a class `CubicleWorld` that has a 2-d array of `Cubicles`, i.e., a floor plan.
The array has dimensions just big enough to store the entire floor plan including internal spaces.

A `Cubicle` object has fields `name`, `row`, `column`.

<table>
<thead>
<tr>
<th>Row</th>
<th>Alice</th>
<th>Dilbert</th>
<th>Dogbert</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Ratbert</td>
<td>Wally</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Asok</td>
<td>Carol</td>
<td>Calbert</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>column</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 2 3 4</td>
</tr>
</tbody>
</table>

When might I use public fields?

- Client needs easy access to fields
- The only "service" that the class provides is to collect related data under one name
- One should still consider using private fields though!

What we learned...

- Develop/implement algorithms for problems
- Develop programming skills
  - Design, implement, document, test and debug
- Apply programming languages
  - Control structures
  - Function/methods for reducing redundancy
  - Data structure
  - Fundamentals of object oriented programming, including inheritance
What we learned... (cont'd)

- Specific tasks
  - Simulating systems
  - Sorting
  - Searching
  - Plotting numeric data

Final Exam

- Fri, May 13, noon-2:30pm, Barton East
- 2/3 Java, 1/3 MATLAB
- Closed-book exam, no calculators
- Bring student ID card

- Check for announcements on webpage:
  - Study break office/consulting hours
  - Review session time and location
  - Review questions
  - List of potentially useful functions/methods