Collections, Part Two
Announcements

- Sections announced over email and room locations are now posted.
- Mac Users: We now have two versions of the starter files:
  - One version is purely for 10.7
  - The other version is for 10.6 and 10.8.
- When submitting, make sure to submit your .cpp source files, not the Xcode or Visual Studio project files!
Announcements

• Casual dinner for women studying CS **tonight** at 5:00PM at the Gates Patio.

• Everyone is welcome!

• RSVP through link sent out last Friday, or by visiting

she++
Vector
Vector

• The Vector is a collection class representing a list of things.
  • Similar to Java's ArrayList type.
• Probably the single most commonly used collection type in all programming.
Example: Cell Tower Purchasing
Buying Cell Towers

137  42  95  272  52
Buying Cell Towers

137 42 95 272 52
Buying Cell Towers

14  22  13  25  30  11  9
Buying Cell Towers
Buying Cell Towers

99  100  99
Buying Cell Towers
Given the populations of each city, what is the largest number of people you can provide service to?
Pass-by-Reference and Objects

- Recall: In C++, *all* parameters are passed by value unless specified otherwise.

- When using container types (*Stack*, *Vector*, etc.) it is often useful to use pass-by-reference for efficiency reasons.
  - Takes a *long* time to make a copy of a large collection!
Maximize what's left in here.
Maximize what's left in here.
Maximize what’s left in here.
Maximize what's left in here.
How the Recursion Works
How the Recursion Works
How the Recursion Works
How the Recursion Works

Best is 13
How the Recursion Works

Best is 13

14 22 13

22 13

13

13 13
How the Recursion Works

Best is 13

14  22  13

22  13

+22

Best is 13

Best is 0
How the Recursion Works

Best is 13

Best is 22

Best is 0
How the Recursion Works

14 - 22 - 13

Best is 13

22 - 13

Best is 22

+22

13

Best is 0

+14

14 - 22 - 13
How the Recursion Works

Best is 13
How the Recursion Works

Best is 13

Best is 27

Best is 22

Best is 13

Best is 0

Best is 13

Best is 22

Best is 14
How the Recursion Works
Grid
Two-Dimensional Data

• The **Grid** type can be used to store two-dimensional data.
  • e.g. matrices, scrabble boards, etc.
• Can construct a grid of a certain size by writing
  \[
  \text{Grid}<\text{Type}> \; g(\text{numRows}, \text{numCols})
  \]
• Can access individual elements by writing
  \[
  g[\text{rows}][\text{cols}]
  \]
Next Time

- **Map**
  - A collection for storing associations between elements.

- **Set**
  - A collection for storing an unordered group of elements.

- **Lexicon**
  - A special kind of *Set*. 