April 13, 2006

Prof. Rodger
Announcements

• No Reading for next time, more on Maps
• No Reading Quiz for next time
• Final Exam – Thursday, May 4, 2-5pm
Last Time – Recursive Art

• Two ways to draw art recursively
  – One object
    • Repeatedly draw the same object smaller
  – Multiple objects
    • Each object is “linked” to a smaller object
    • Each object draws itself
    • See the myNext variable
Maps

• Maps are another way of organizing data
• Keys and Values
  – Each key maps to a value
  – Some keys can map to the same value
  – Can change the value a key maps to
Example

- Each student could be mapped to their favorite ice cream flavor

![Diagram showing students and ice cream flavors]

- Students: Astrachan, Sun, Rodger, Forbes
- Ice Cream Flavors: Chocolate, Chocolate Chip, Strawberry
Implementing a Map

• We will use TreeMap in Java
• Example:
  
  ```java
  Map<String, String> fav = new TreeMap<String, String>();
  ```
• Keys map to values
To use a Map

- Put in a key and its value
  ```
  fav.put("Forbes", "Strawberry");
  ```
- Get a value for a key
  ```
  val = fav.get("Forbes");
  ```
- Change value for key
  ```
  fav.put("Astrachan", "Coffee Mocha");
  ```
Change Astrachan’s value

Students
- Astrachan
- Sun
- Rodger
- Forbes

Ice Cream Flavors
- Coffee Mocha
- Chocolate Chip
- Strawberry
Value could be a set

Students

- Astrachan
- Sun
- Rodger
- Forbes

Ice Cream Flavors

- Coffee Mocha
- Chocolate
- Vanilla
- Blueberry
- Chocolate Chip
- Blueberry
- Banana
- Strawberry
- Coffee Mocha
Classwork today

• File of words
  – Determine number times each words appears
  – For each word, determine all line numbers it appears on
  – For each alphabetical letter, determine all the words that start with that letter.
First look at methods given

• main
• getWordcounts
  – Given a Scanner bound to a file
  – Return a Map of words to counts
• printResults
  – Given a map print key followed by value
public Map<String, Integer> getWordCounts (Scanner input) {
    Map<String, Integer> results = new TreeMap<String, Integer>();

    while (input.hasNext()) {
        String word = input.next();

        Integer count = results.get(word);
        if (count == null) {
            results.put(word, 1);
        } else {
            results.put(word, count + 1);
        }
    }

    return results;
}
public void printResults (Map<String, ?> results)
{
    for (String key : results.keySet())
    {
        System.out.println(key + "\t" + results.get(key).toString());
    }
}
<table>
<thead>
<tr>
<th>Word</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>aid</td>
<td>1</td>
</tr>
<tr>
<td>aided</td>
<td>1</td>
</tr>
<tr>
<td>air</td>
<td>1</td>
</tr>
<tr>
<td>alarming</td>
<td>1</td>
</tr>
<tr>
<td>all</td>
<td>2</td>
</tr>
<tr>
<td>all,</td>
<td>1</td>
</tr>
<tr>
<td>aloud-</td>
<td>1</td>
</tr>
<tr>
<td>am</td>
<td>1</td>
</tr>
<tr>
<td>among</td>
<td>2</td>
</tr>
<tr>
<td>an</td>
<td>6</td>
</tr>
<tr>
<td>and</td>
<td>54</td>
</tr>
<tr>
<td>another</td>
<td>2</td>
</tr>
<tr>
<td>answer</td>
<td>1</td>
</tr>
<tr>
<td>answer.</td>
<td>1</td>
</tr>
<tr>
<td>answered</td>
<td>1</td>
</tr>
<tr>
<td>any</td>
<td>1</td>
</tr>
<tr>
<td>aperture</td>
<td>1</td>
</tr>
</tbody>
</table>
Todo: getLineNumbers

- Map each word to a set of line numbers it occurs on

<table>
<thead>
<tr>
<th>word</th>
<th>lines</th>
</tr>
</thead>
<tbody>
<tr>
<td>offering</td>
<td>[213]</td>
</tr>
<tr>
<td>old</td>
<td>[25, 329]</td>
</tr>
<tr>
<td>on</td>
<td>[33, 58, 86]</td>
</tr>
<tr>
<td>on,</td>
<td>[215]</td>
</tr>
<tr>
<td>on,'</td>
<td>[108]</td>
</tr>
<tr>
<td>on.</td>
<td>[178]</td>
</tr>
<tr>
<td>once</td>
<td>[127]</td>
</tr>
<tr>
<td>one</td>
<td>[30, 58, 93, 96, 224, 230, 245]</td>
</tr>
<tr>
<td>one.</td>
<td>[186]</td>
</tr>
<tr>
<td>only</td>
<td>[9, 325]</td>
</tr>
<tr>
<td>opportunity</td>
<td>[23]</td>
</tr>
<tr>
<td>or</td>
<td>[226, 227]</td>
</tr>
<tr>
<td>orbs</td>
<td>[111]</td>
</tr>
<tr>
<td>orders</td>
<td>[91, 92]</td>
</tr>
<tr>
<td>ornamented</td>
<td>[222]</td>
</tr>
<tr>
<td>other</td>
<td>[19, 244, 245]</td>
</tr>
<tr>
<td>our</td>
<td>[214, 216, 301]</td>
</tr>
</tbody>
</table>
Todo: getFrequencies

- Map each letter of alphabet to words

<table>
<thead>
<tr>
<th>T</th>
<th>[The, Then, There, These, They, Three, Throwing, Unsheathing]</th>
</tr>
</thead>
<tbody>
<tr>
<td>U</td>
<td>[We, When, Will, With, Withdrawing, Within]</td>
</tr>
<tr>
<td>W</td>
<td>[You, You, Your]</td>
</tr>
</tbody>
</table>
| Y  | `A, `Amontillado!, `Amontillado?, `And, a, about, above, absconded, accosted, admired, back, back:, back;, backed, bargain.'', be, bed called, came, cannot, cap, carnival, carnival! d'or,, damp, damp., dampness, damps.'', dear, each, earliest, earth,, eighth,, ejaculated, e. fabric, face,, fails, fall, familiarly,, family gait, gave, gemmery,, gesticulation, getting, ha!, ha!-he!, had, hairs, half, hand, hand,'' idea, ignoramus,'', ill,, imbedded, immediate, jest,'' jest., jingled, jingled., jingling, jok key, kill, knew,, knocked, know l [labour., labours, lacessit.'', laid, largely, made, madness, make, man, man., manner., many,