CompSci 6
Programming Design and Analysis

```
offering        [213]
old     [25, 329]
on      [33, 58, 86]
on,     [215]
on,''   [108]
on.     [178]
once    [127]
one     [30, 58, 93, 96, 224, 230, 245]
one.    [186]
only    [9, 325]
opportunity,    [23]
or      [226, 227]
orbs    [111]
orders  [91, 92]
ornamented      [222]
other   [19, 244, 245]
our     [214, 216, 301]
```

April 20, 2010

Prof. Rodger

Announcements

- Next time: More maps
- Assignment 9 out – Due Thursday

Last Time – Recursive Art

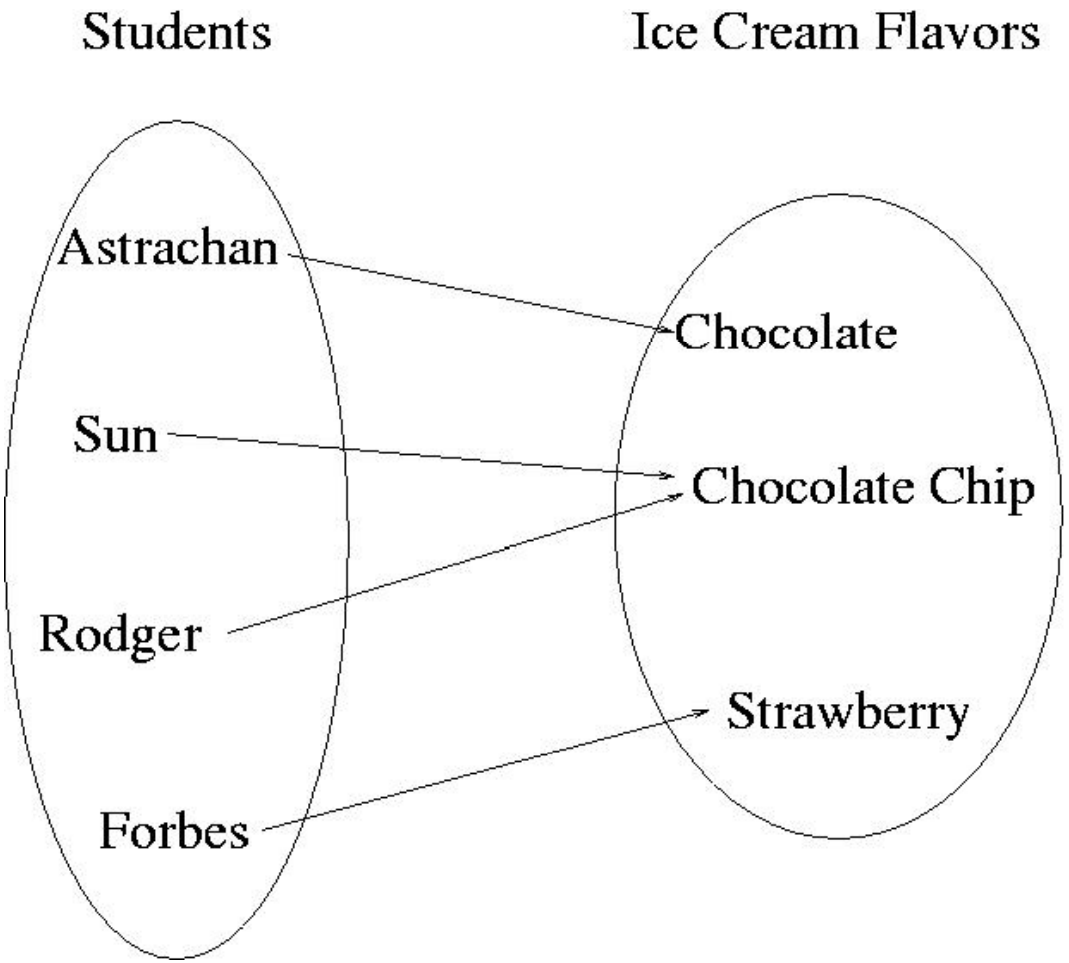
- Two ways to draw art recursively
  - One object
    - Repeatedly draw the same object smaller
  - Multiple objects
    - Each object is “linked” to a smaller object
    - Each object draws itself
    - See the myNext variable

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



Implementing a Map

- We will use TreeMap in Java
- Example:

```
 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”);
```

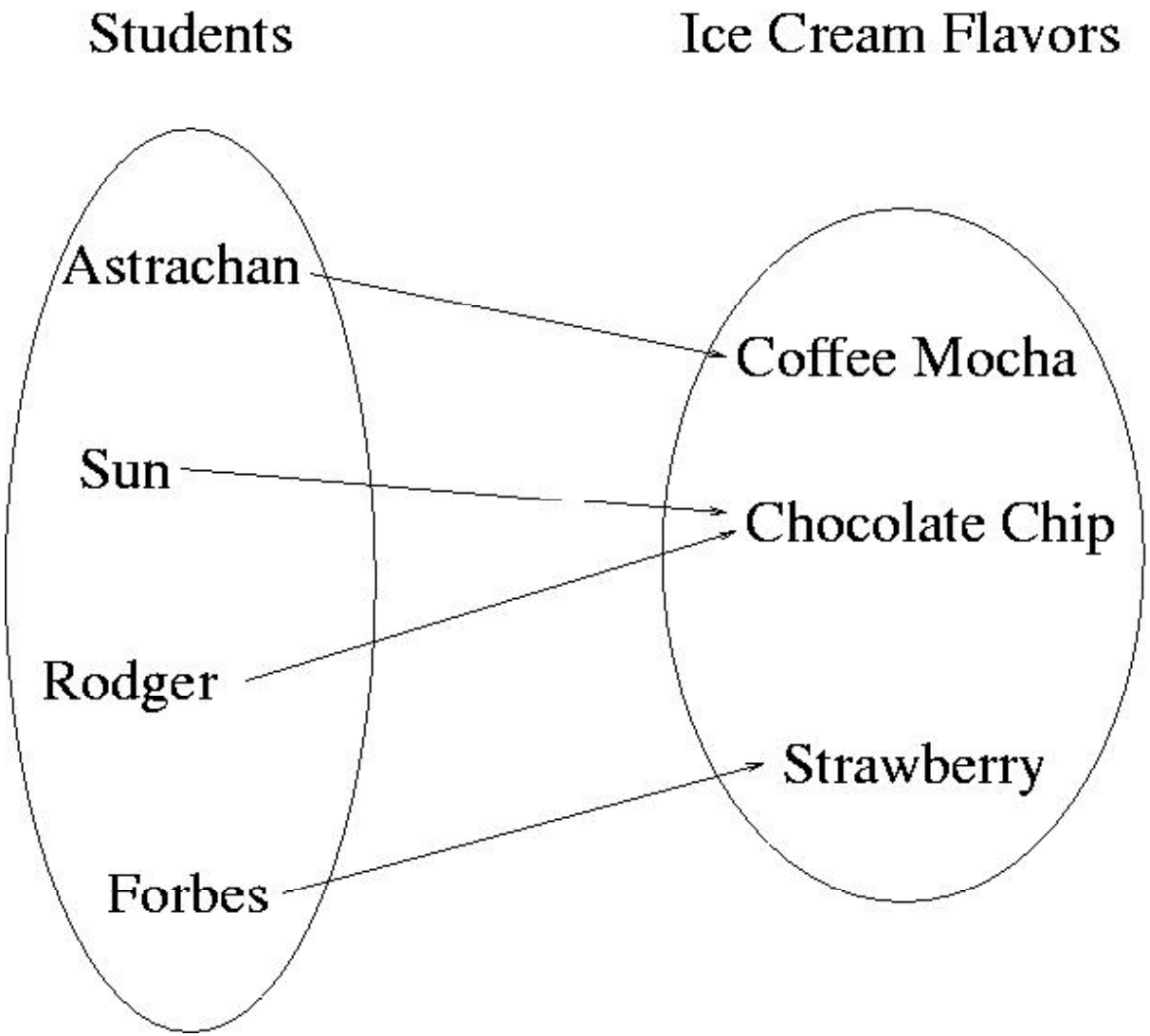
- Get a value for a key

```
val = fav.get(“Forbes”);
```

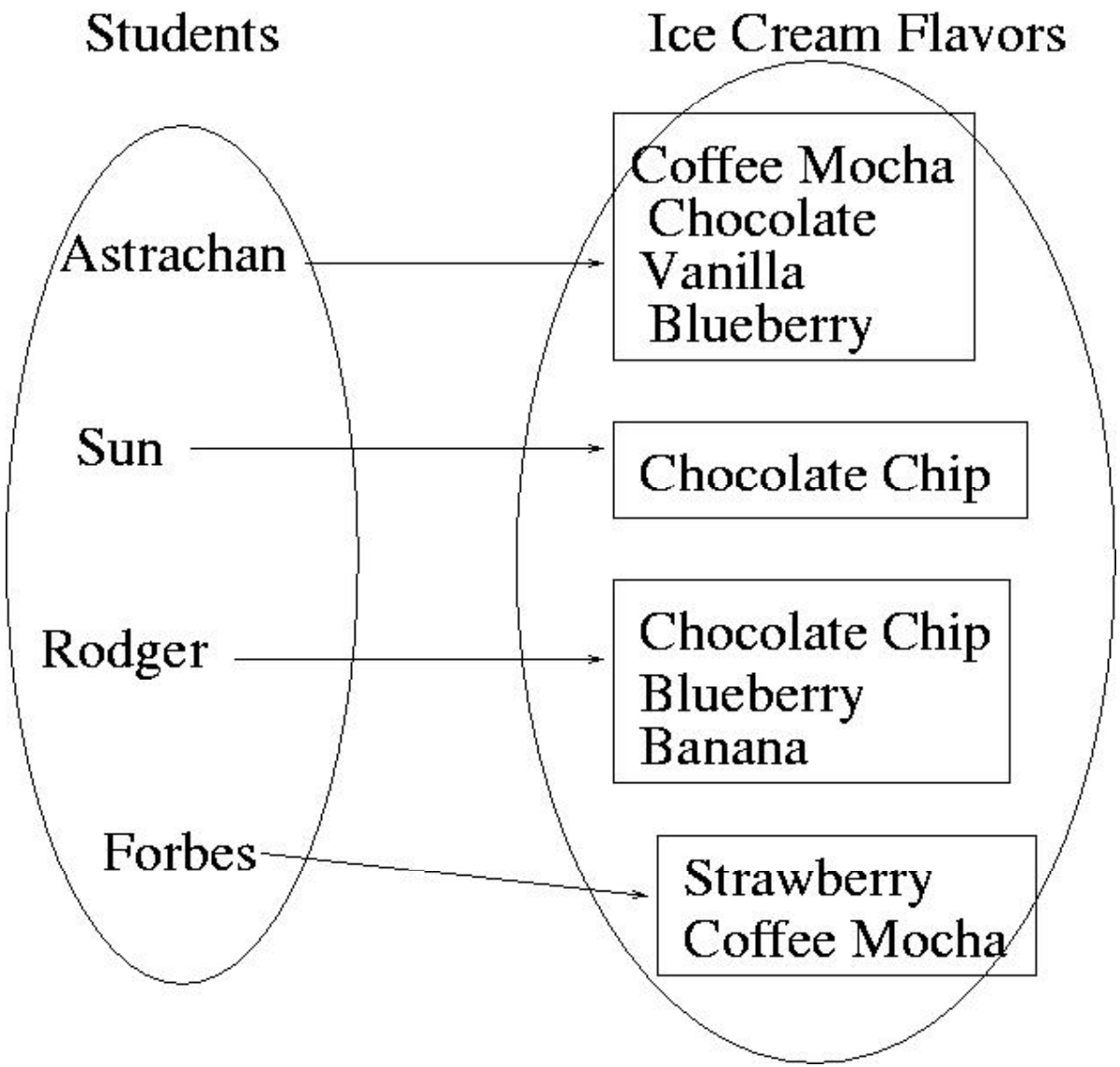
- Change value for key

```
fav.put(“Astrachan”, “Coffee
  Mocha”);
```

Change Astrachan's value



Value could be a set



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

```java
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;
}
```

printResults

```
public void printResults (Map<String, ?> results)
{
    for (String key : results.keySet())
    {
        System.out.println(key + "\t" +
                results.get(key).toString());
    }
}
```

Output

```
aid       1
aided,    1
air       1
alarming          1
all       2
all,      1
aloud-    1
am        1
among     2
an        6
and       54
another   2
answer    1
answer.   1
answered          1
any       1
aperture          1
```

Todo: getLineNumbers

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

```
offering        [213]
old     [25, 329]
on      [33, 58, 86]
on,     [215]
on,''   [108]
on.     [178]
once    [127]
one     [30, 58, 93, 96, 224, 230, 245]
one.    [186]
only    [9, 325]
opportunity,    [23]
or      [226, 227]
orbs    [111]
orders  [91, 92]
ornamented      [222]
other   [19, 244, 245]
our     [214, 216, 301]
```

Todo: getFrequencies

- Map each letter of alphabet to words

```
T       [The, Then, There, These, They, Three, Throwing,
U       [Unsheathing]
W       [We, When, Will, With, Withdrawing, Within]
Y       [You, You,, Your]
`       [``A, ``Amontillado!'', ``Amontillado?, ``And,
a       [a, about, above, absconded, accosted, admired,,
b       [back, back., back;, backed, bargain.'', be, be
c       [called, came, cannot, cap, carnival, carnival!
d       [d'or,, damp, damp., dampness, damps.'', dear, (
e       [each, earliest, earth,, eighth,, ejaculated, e.
f       [fabric, face,, fails, fall, familiarly,, famil
g       [gait, gave, gemmary,, gesticulation, getting, (
h       [ha!, ha!-he!, had, hairs, half, hand, hand,'',
i       [idea, ignoramus,'', ill,, imbedded, immediate,
j       [jest,'', jest., jingled, jingled., jingling, j
k       [key, kill, knew,, knocked, know]
l       [labour., labours, lacessit.'', laid, largely, .
m       [made, madness, make, man, man., manner., many,
```