Dictionaries/Maps

- Dictionaries/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 Dictionary/Map

Keys map to values

• Create Empty dictionary
  `somemap = {}`

• Put in a key and its value
  `somemap[“Forbes”] = “Strawberry”`

• Get a value for a dictionary
  `value = somemap[“Forbes”]`
  OR `value = somemap.get(“Forbes”, “default”)`

• Change a value for a dictionary
  `somemap[“Forbes’] = “Chocolate”`

More on using a Dictionary/Map

• Get all the keys
  `– listKeys = somemap.keys()`

• Get all the values
  `– listValues = somemap.values()`

• Other methods
  `– clear – empty dictionary`
  `– items – return (key,value) pairs`
  `– iteritems – return (key,value) pairs more efficiently`
  `– update – update with another dictionary`

Change Astrachan’s value
`somemap[“Astrachan”] = Coffee Mocha`

Value could be a set or list

Students | Ice Cream Flavors
---|---
Astrachan | Coffee Mocha
Sun | Chocolate Chip
Rodger | Strawberry
Forbes |

Students | Ice Cream Flavors
---|---
Astrachan | Coffee Mocha
Sun | Chocolate Chip
Rodger | Chocolate Chip
Forbes | Blueberry
Forbes | Strawberry
Forbes | Coffee Mocha
Back to Popular Name Problem:

- Given a list of names, determine the most popular first name and print that name with all of its last names.
- Input: Names are always two words, names are in a file. If multiple names are on the same line they are separated by a “:”
- Output: Most popular first name, followed by a “:”, followed by corresponding last names separated by a blank

Now use a dictionary/map

- We will write three maps for practice
  - First name to count of corresponding last names
  - First name to list of corresponding last names
  - First name to set of corresponding last names
- Which map is most useful to solve this problem?

Compare

- Using two parallel lists?
- Using one dictionary/map