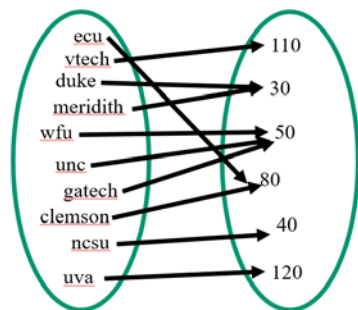


# CompSci 101

## Introduction to Computer Science



November 9, 2017

Prof. Rodger

cps101 fall 2017

1

## Announcements

- Assign 7 due Monday
- APT 7 due Tuesday
- Exam 2 Thursday, November 16
  - See practice exams from Fall 16 and Spring 17
- Today:
  - More problem solving with dictionaries
  - Finish problem from last time

cps101 fall 2017

2

## Be in the know....

### ACM, compsci mailing lists

- Association of Computing Machinery (ACM)
  - Professional organization for computer science
  - Duke Student ACM Chapter – join for free
- Join duke email lists to find out info on **jobs**, **events** for compsci students
  - lists.duke.edu – join lists:
    - compsci – info from compsci dept
    - dukeacm – info from student chapter



cps101 fall 2017

3

## Review Dictionaries

- Map keys to values
  - Counting: count how many times a key appears
    - Key to number
  - Store associated values
    - Key to list or set
- Get all
  - Keys, values or (key,value) pairs
- What question do you want to answer?
  - How to organize data to answer the question

cps101 fall 2017

4

## Dictionary problems

### Number of students in Photo clubs

[bit.ly/101f17-1109-1](http://bit.ly/101f17-1109-1)

```
d = {'duke':30, 'unc':50, 'ncsu':40}
```

```
d['duke'] = 80
```

```
d.update({'ecu':40, 'uncc':70})
```

```
print d.values()
```

cps101 fall 2017

5

## Dictionary problems – part 2

[bit.ly/101f17-1109-2](http://bit.ly/101f17-1109-2)

- Consider the Python dictionary below maps schools to number of students in the Photo Club at their school

```
d = {'duke':30, 'unc':50, 'ncsu':40, 'wfu':50, 'ecu': 80, 'meridith':30, 'clermson':80, 'gatech':50, 'uva':120, 'vtech':110}
```

Dictionary to answer which schools have X students? ... which schools have groups of students 1-49, 50-99, etc?

6

## Inverted Dictionary

[bit.ly/101f17-1109-3](http://bit.ly/101f17-1109-3)

- Start with dictionary of keys to values
  - Schools to number of students*
- Use it to build an inverted dictionary of values to keys (actually list of keys)
  - Number of students to list of schools*
- Lets look at the code

cps101 fall 2017

13

## Dictionary Song problem

[bit.ly/101f17-1109-4](http://bit.ly/101f17-1109-4)

```
songs = ["Hey Jude:Let it be:Day Tripper",  
"Let it be:Drive my car:Hey Jude",  
"I want to hold your hand:Help!:Day Tripper",  
"Born to run:Thunder road:She's the one",  
"Hungry heart:The river:Born to run",  
"The river:Thunder road:Drive my car",  
"Angie:Start me up:Ruby Tuesday",  
"Born to run:Angie:Drive my car"]
```

cps101 fall 2017

14

## Building the dictionary d

*"Hey Jude:Let it be:Day Tripper"*

## APT EmailsCourse bit.ly/101f17-1109-5

You are given a list of strings of course information, where each string is in the format "coursename:person:email". Your task is to determine the course with the most people and to return the emails of those people in the largest course. The emails should be returned as a string with the emails in alphabetical order. If there is more than one largest course, return the emails of such course that comes first in alphabetical order.

```
["CompSci 100:Fred Jack Smith:fjs@duke.edu",  
 "History 117:Fred Jack Smith:fjs@duke.edu",  
 "CompSci 102:Arielle Marie Johnson:amj@duke.edu",  
 "CompSci 100:Arielle Marie Johnson:amj@duke.edu",  
 "CompSci 006:Bertha White:bw@duke.edu",  
 "Econ 051:Bertha White:bw@duke.edu",  
 "English 112:Harry Potter:hp@duke.edu",  
 "CompSci 100:Harry Potter:hp@duke.edu"]
```

Returns "amj@duke.edu fjs@duke.edu hp@duke.edu"<sup>26</sup>

## Step 1 – Work small example by hand

```
["CompSci 100:Fred Jack Smith:fjs@duke.edu",  
 "History 117:Fred Jack Smith:fjs@duke.edu",  
 "English 112:Harry Potter:hp@duke.edu",  
 "CompSci 100:Harry Potter:hp@duke.edu"]
```