CompSci 101
Sorting, CSV

S is for …

• Software
  • Joy, sorrow, fun, changing the world

• System and sys
  • Connecting to the machine at different levels

• Sorting
  • From hat to timsort to more

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Like a Rolling Stone</td>
<td>Bob Dylan</td>
</tr>
<tr>
<td>2</td>
<td>Satisfaction</td>
<td>The Rolling Stones</td>
</tr>
<tr>
<td>3</td>
<td>Imagine</td>
<td>John Lennon</td>
</tr>
<tr>
<td>4</td>
<td>What's Going On</td>
<td>Marvin Gaye</td>
</tr>
<tr>
<td>5</td>
<td>Respect</td>
<td>Aretha Franklin</td>
</tr>
<tr>
<td>6</td>
<td>Good Vibrations</td>
<td>The Beach Boys</td>
</tr>
<tr>
<td>7</td>
<td>Johnny B. Goode</td>
<td>Chuck Berry</td>
</tr>
<tr>
<td>8</td>
<td>Hey Jude</td>
<td>The Beatles</td>
</tr>
<tr>
<td>9</td>
<td>Smells Like Teen Spirit</td>
<td>Nirvana</td>
</tr>
<tr>
<td>10</td>
<td>What'd I Say</td>
<td>Ray Charles</td>
</tr>
</tbody>
</table>

Barbara Liskov

• Among first women in US to earn Ph.D. in Computer Science: 1968
• Turing Award 2008, Software Engineering and Programming Languages
• Object-Oriented
  • CLU
• Liskov Substitution Principle

“Every time you exchange e-mail with a friend, check your bank statement online, or run a Google search, you are riding the momentum of her research” – MIT President Rafael Reif about Liskov

Announcements

• APT 5 due today!
• Assignment 5 due Thurs, April 6
• No Lab this week
• Reading and Sakai Quizzes due next week
• APT Quiz 2 – today through Monday
APT Quiz 2 March 30-April 3

- Opens March 30, Thursday, 1:15pm
- Closes at 11pm Mon 4/3 – must finish all by this time
- There are two parts based on APTs 1-5
  - Each part has two APT problems
  - Each part is 3 hours – more if you get accommodations
  - Each part starts in Sakai under tests and quizzes
  - Sakai is a starting point with countdown timer that sends you to a new apt page just for each part
  - Could do each part on different day or same days
- Old APT Quiz so you can practice (not for credit) – on APT Page

PFTD

- Sorting
  - Sorting using standard Python APIs
- CSV Library
  - How to read data using standard Python APIs

Why Sort Data?

- Help understand data
  - Great American Eclipse, August 21, 2017
  - Spotify tracked the playing of the song
Why Sort Data?

• Every field needs to visualize and understand data
  • Sorting helps with this from movies to policy to sports to location of infections to


How To Sort: Algorithms

• Does scale matter? It depends!

• You’re playing Spades, Hearts, Bridge, Go-Fish
  • How you sort doesn’t really matter, but whether you sort makes play more efficient? Better?

• Many ways to sort
  • Bubblesort, Insertion sort, Selection sort
  • Quicksort, Mergesort, Timsort, Bogo sort
  • Python uses Timsort

Solve a Larger Problem

• Suppose I were to give you the top 1000 artists
  • Top 1,000 songs, find top 10 artists
  • How many songs per artist?
Scale

• As the size of the problem grows we want ...
  • The algorithm to still work and be fast!
  • What to do?

• Search example
  • Google search results work
  • SoundHound/Shazam results work
  • ContentID on YouTube results work

Python to the Rescue

• Using .sort(...), sorted(...), and lambda
• Using CSV library and its API
  • CSV – Comma Separated Values

• Why use the CSV library?
  • How to handle the song “Hello, I Love You”?
  • Row 166 in spreadsheet

Hits by Artists: SongReader.py

• What is returned by this function?
  • details of csv: next and no split and ...

```python
def countByArtist(name):
    csvf = open(name, 'r', encoding='utf-8')
    freader = csv.reader(csvf)
    header = next(freader)
    print("header row labels", header)
    data = {}
    for row in freader:
        artist = row[2]
        if artist not in data:
            data[artist] = 0
            data[artist] += 1
    csvf.close()
    return data
```

WOTO-2 countByArtist
Two APIs: CSV and Sorting

- **CSV Library to read and process data**
  - Comma-separated, but can separate by ":\.", or any character as we'll see later

- **Similar to reading a file – returned by open**
  - Iterable is returned by `csv.reader`
  - The `next` function advances iterable
  - Don't call `split`, we can access by index
    - Also by header-row label with `csv.dictreader`

CSV API

- `freader = csv.reader(file)` – returns an iterable
  - Every line from the file in a form ready for you
- `line = next(freader)`
  - Gives you next row as list of strings
- `for row in freader`:
  - Gives you the rest of rows as list of strings

What does this do? `freader` an iterable
Where name is a filename

```python
csvf = open(name, 'r', encoding='utf-8')
freader = csv.reader(csvf)
print("freader", freader)
header = next(freader)
print("header", header)
for row in freader:
    print("row", row)
```

What if you call `next` one extra time?
Where name is a filename

```python
csvf = open(name, 'r', encoding='utf-8')
freader = csv.reader(csvf)
print("freader", freader)
header = next(freader)
print("header", header)
nextline = next(freader)
print("next", nextline)
for row in freader:
    print("row", row)
```
Sorting to Print/Visualize

- Dictionary is tuples, (key,value) like ('Beatles', 51)
  - But tuples not in order, so we must put in order...

```python
if __name__ == '__main__':
    counts = countByArtist("data/top1000.csv")
    print('nFirst 5 artists:')
    for artist in sorted(counts.items())[:5]:
        print(artist)
    print('nTop 5 artists:')
    sortbycount = sorted([(a[1], a[0]) for a in counts.items()])
    sortedArtists = [(a[1], a[0]) for a in sortbycount]
    for artist in sortedArtists[-5:]:
        print(artist)
```

Sorting API and Sorting Concepts

- What is `counts.items()` – how is it sorted?
  - A list, you can slice a list, look for clues!
  - What can be sorted? A sequence
  - `sorted(counts.items())`

- What does `sorted` return?
  - A list, you can slice a list, look for clues!

Sorting by Number of Songs

- Sort by first value vs sort by second value
  - Need to put sequence back to original format

```python
print('nFirst 5 artists:')
for artist in sorted(counts.items())[:5]:
    print(artist)
print('nTop 5 artists:')
sortedArtists = sorted([(a[1], a[0]) for a in counts.items()])
sortedArtists = [(a[1], a[0]) for a in sortedArtists]
for artist in sortedArtists[-5:]:
    print(artist)
```