

Compsci 101, Lists, Loops, Problems

Owen Astrachan
Kristin Stephens-Martinez
February 24, 2018

2/24/2018

Compsci 101, Spring 2018, Lists, Loops,
Problems

1

J is for ...

- **JSON**
 - Format for data transmitted across the web
- **JPEG**
 - Image format based on lossy compression
- **Jacquard Loom**
 - 1804 "automated" loom



2/24/2018

Compsci 101, Spring 2018, Lists, Loops,
Problems

2

PF Thursday

- Analyzing and comparing programming techniques
 - How does indexing work?
- Solving an APT using techniques you know
 - Parallel lists, e.g., as in FileFrequency.py
 - Counting # words, analyzing frequencies

2/24/2018

Compsci 101, Spring 2018, Lists, Loops,
Problems

3

How do we program this?

<http://bit.ly/101-ngram>

Word Tally

the	
been	
message	
persevere	
nation	

2/24/2018

Compsci 101, Spring 2018, Lists, Loops,
Problems

4

Definite Indefinite Loops

```
["a", "b", "c", "a"].index("b") == 1
["a", "b", "c", "a"].index("B") TROUBLE!
["a", "b", "c", "a"].index("B") ??? -1
```

- How do we write a **better index function**?
 - Is this a definite, e.g., a for loop?
 - Is this an indefinite, e.g., a while loop?
 - Best case # iterations, worst case # iterations?

2/20/2018

Compsci 101, Spring 2018, while, api

5

Fast and Built-in

- What does `.index` return when element not in list?
 - Throws exception aka generates error and dies
 - Standard Python idiom shown below, WHAT?

```
9 def index1(source, key):
10     if key in source:
11         return source.index(key)
12     return -1
```

2/20/2018

Compsci 101, Spring 2018, while, api

6

Slower, but fast and illustrative

- What does `break` do in a loop?
 - Why is `found` initialized to `-1` before loop?

```
14 def index2(source, key):
15     found = -1
16     for dex in range(len(source)):
17         if source[dex] == key:
18             found = dex
19             break
20     return found
```

2/20/2018

Compsci 101, Spring 2018, while, api

7

Best and Worst of all

- Reason about loop test/guard when finished
 - Why is `if` needed after loop?

```
22 def index3(source, key):
23     dex = 0
24     while dex < len(source) and source[dex] != key:
25         dex += 1
26
27     if dex == len(source):
28         return -1
29     return dex
```

2/20/2018

Compsci 101, Spring 2018, while, api

8

Indexing.py

- Running to test three solutions. Look up every possible key and keys that aren't in list
 - Timing is shown below
- index1 for 5000 = 1.316
- index2 for 5000 = 4.474
- index3 for 5000 = 13.776

2/24/2018

Compsci 101, Spring 2018, Lists, Loops,
Problems

9

Short Circuit Evaluation

- Short circuit evaluation, these are not the same!

```
while dex < len(source) and source[dex] != key:  
while source[dex] != key and dex < len(source):
```

- As soon as truthiness of expression known ...
 - Stop evaluating
 - In A and B, if A is false, do not evaluate B

2/20/2018

Compsci 101, Spring 2018, while, api

10

Reasoning About Loops

- DeMorgan's Laws

not(A and B) equivalent to **not(A) or not(B)**

not(A or B) equivalent to **not(A) and not(B)**

```
while dex < len(source) and source[dex] != key:
```

- When loop is finished we know
 - Either dex is too large OR we found key at dex
 - Must test, but which test?

2/20/2018

Compsci 101, Spring 2018, while, api

11

Python Logic Summarized

- A and B is True only when A is True and B is True
- A or B is False only when A is False and B is False
- Short-circuit evaluation of A or B ?
 - If A is true, do not evaluate B

```
if x == 3 or functionCall() == 5:
```

 - If x == 5 is function called? If x == 3?
- Reasonable languages have short-circuit boolean expressions

2/20/2018

Compsci 101, Spring 2018, while, api

12

Lessons from Indexing Code?

- The built-in Python functions are optimized
 - Execute using efficient low-level implementation
 - We use C-Python, there are other interpreters
- Behind the scenes in Python
 - You do not need to know this, but ...
 - Don't look behind the curtain!
 - <https://www.youtube.com/watch?v=ublpoPjBUds>

2/20/2018

Compsci 101, Spring 2018, while, api

13

Case Study: FileFrequency.py

- We'd like to analyze word occurrences
 - Google N-Gram, it's easy to do, but ...
 - What about occurrences of "cgat" in genome?
 - What about Rotten Tomatoes?
- We'll build towards more efficient structures
- We'll use an API for opening files

2/20/2018

Compsci 101, Spring 2018, while, api

14

High Level View

- We will use parallel lists to track data
 - Each words stored in a list named **words**
 - Word counts stored in a list named **counts**
 - # occurrences of **words[k]** in **counts[k]**

```
["apple", "fox", "vacuum", "lime"]
[5, 2, 25, 15]
```

- What happens when we read a word?

2/20/2018

Compsci 101, Spring 2018, while, api

15

Pseudo-code for getFileData

- Let user choose a file to open
 - Deal with errors
- Read each line of the file
 - Process each word on the line
 - If word never seen before? Add to words and counts
 - Update # occurrences using .index and location
- Create list of ["apple", 5] pairs of each word
 - Return this list

2/20/2018

Compsci 101, Spring 2018, while, api

16

From Pseudo to Code

```

27 for line in f:
28     data = line.strip().split()
29
30     for word in data:
31         word = word.lower()
32         if word not in words:
33             words.append(word)
34             counts.append(0)
35         location = words.index(word)
36         counts[location] += 1

```

2/24/2018

Compsci 101, Spring 2018, Lists, Loops,
Problems

17

Comparing Two Approaches

- Why do we have a loop in a loop?
 - Code mirrors structure:
 - file has lines, lines have words
- Notice:

```

for line in f:
    data = line.strip().split()

    for word in data:
        word = word.lower()
        if word not in words:
            words.append(word)
            counts.append(0)
        location = words.index(word)
        counts[location] += 1

```

- **.lower**
- **not in**
- **+=**
- **.append**
- **.index**

2/20/2018

Compsci 101, Spring 2018, while, api

18

Comparing Two Approaches

- Why do we have only one loop?
 - Code mirrors structure, which is better?
 - File is a sequence of characters!!

```

for word in f.read().lower().split():
    if word not in words:
        words.append(word)
        counts.append(0)
    location = words.index(word)
    counts[location] += 1

```

2/20/2018

Compsci 101, Spring 2018, while, api

19

WOTO

<http://bit.ly/101spring18-feb20-2>



2/20/2018

Compsci 101, Spring 2018, while, api

20

BagFitter

- <https://www2.cs.duke.edu/csed/pythonapt/bagfitter.html>
- How do you solve this problem by hand?
 - Strength = 2? Strength = 3?
- If we have # occurrences...
 - Similar to cement trucks?



2/24/2018

Compsci 101, Spring 2018, Lists, Loops,
Problems

21

Pseudo code solution

- Determine # occurrences of each unique item
 - Suppose these are in a list of int values
- How many bags needed? Is it $n/\text{strength}$?
 - Use numbers: n is 8, strength is 3 or 4 or 5
- Which should we tackle first? Need to debug
 - What problems have we seen before?

2/24/2018

Compsci 101, Spring 2018, Lists, Loops,
Problems

22

Testing in Eclipse

- Using ideas from FileFrequency.py ...
 - Create parallel lists: strings and ints
 - Strings in `bags []` and # in `counts []`
- Maintain invariant that # occurs of string in `bags [k]` is stored in `counts [k]`
 - When we see string seen before?
 - When we see new string?

2/24/2018

Compsci 101, Spring 2018, Lists, Loops,
Problems

23

From Pseudo to Code

- Same idea we saw in FileFrequency.py
 - We can also change code to use in bags rather than not in bags
 - Call `.append(1)` instead of `.append(0)`

```
for item in food:
    if item not in bags:
        bags.append(item)
        # do something
    location = bags.index(item)
    counts[location] += 1
```

2/24/2018

Compsci 101, Spring 2018, Lists, Loops,
Problems

24

Testing the Module

- Call function with example from description
 - Print different bags and their counts
 - Verify by hand/eye that this works
- Debug and fix as needed, still not ready to test in online APT tester
 - More than one part? Test each separately.
 - After parts work? Integrate and test online

2/24/2018

Compsci 101, Spring 2018, Lists, Loops,
Problems

25

Helper Function

- Given one number and strength, return # bags
 - Eerily similar to midterm **trucksNeeded**
 - More than one way to do this, ideas?
 - Suppose strength is 9, and we have 8 items?
What about 9 or 10 items?
 - Can we use % and //
 - Can we modify value and just use // + 1?

2/24/2018

Compsci 101, Spring 2018, Lists, Loops,
Problems

26

Time to Do Another?

- Eating Good vs. Eating Well?
 - <https://www2.cs.duke.edu/csed/pythonapt/eatinggood.html>
 - How do we guard against "double adding"
 - What else do we need to do to solve this?

```
if name not in names:
    names.append(name)
```

2/24/2018

Compsci 101, Spring 2018, Lists, Loops,
Problems

27