CompSci 101
Introduction to Computer Science

Nov 15, 2016

Review
for
exam

Prof. Rodger

Announcements

- Exam 2 Thursday
- Reading and RQ for next week – coming…
- Assignment 7 due Nov 29
- APT 8 due today
  - Doing extra ones – good practice for exam
- No Lab this week!
- No Consulting Hours Thursday night
- Review Session – Wed 7:30pm LSRC B101
- Today:
  - Finish notes from last time – Dictionary timings
  - Reviewing for the exam

Clever Hangman

- Version of Hangman that is hard to win.
- Program keeps changing secret word to make it hard to guess!
- User never knows!
- Once a letter is chosen and shown in a location, program picks from words that only have that letter in that location
- Program smart to pick from largest group of words available

Clever Hangman - Dictionary

- Builds a dictionary of categories
- Start with list of words of correct size
- Repeat
  - User picks a letter
  - Make dictionary of categories based on letter
  - New list of words is largest category
    - Category includes already matched letters
    - List shrinks in size each time

Clever Hangman Example

- Possible scenerio after several rounds

```
 (secret word: calls) # words possible 176
You guessed a letter
You have this many guesses left: 4
Letters not guessed: bcdfghjklmnpqrstvwxyz
guessed so far: _ a _ _ _
guess a letter or enter + to guess a word: d
```

- From list of words with **a** the second letter. From that build a dictionary of list of words with **no d** and with **d** in different places:

_a___	147	Choose “no d”, most words, 147
add	1	
_a_d_	17	Only 17 words of this type
_ad__	3	
dadd_	1	
da_d_	1	Only 1 word of this type
da___	6	

Exam logistics

- Only need a pen or pencil
- No scratch paper
- See the reference sheet of Python information you will get with the test (see resources page)
- Closed book, closed notes, closed neighbor
- Covers lecture, lab and assigned reading
- Have put old quizzes back up as quiz review
  - This is NOT for a grade, for studying only

Understand old and new topics

- Old topics: if, for, while, lists, strings
- list comprehension, enumerate
- Files – write code - Will give you a file already opened and ready for reading
- Sets, Dictionaries – write code – create and use
- Understand items on Python review sheet on resources page
- HAVE NOT COVERED TOPICS – regular expressions or recursion

The best way to study

- Write code on paper!
- Resources page has old tests and solutions
  - Try writing code, then look at solutions
- Rewrite an APT
- Rewrite code we did in lecture
- Rewrite code we did in classwork or lab

Looping by index or by element

- Strings and lists: use either
  - range(len(x)) for index, can get element
  - enumerate(somelist)
- Sets and Dictionaries: element only
  - Loop over d or d.keys() for dictionary
  - The keys are a set, so similar to set loop
- Which is best when choice? It depends!
  - Can you get element from index?
  - Can you get index from element?

Questions
bit.ly/101f16-1115-1

Unpacking a list comprehension

```
[f(x) for x in foo if condition with x]
[w for w in words if w.endswith('e')]
[(w,words.count(w)) for w in set(words)]
```

- Always possible to use a loop

```
build = [ ]
for x in foo:
   if condition with x:
      build.append(f(x))
```

```
build = [ ]
for w in set(words):
   build.append((w,words.count(w)))
```

Set Concepts

- Set union, intersection, difference
  - s.intersection(t) is the same as s&t
  - s.union(t) is the same as s|t
  - s.difference(t) is the same as s-t

- Sets aren't in order during iteration
  - Convert to list, create from list
  - Sets are really, really efficient for add/search

Dictionaries

- Build a dictionary
  - Counting dictionary
    - string to number
  - Grouping dictionary
    - string to list of items related
- Use a dictionary
  - Get values
  - Get keys
  - Get key,value pair

Questions
bit.ly/101f16-1115-2

Now go over Test Practice problems