Lecture 15 (2-18-26)

Python (Regular Expressions, Processes, Requests)

Containters

Sequence:

- In C : dynamic arrays, linked list
- In Python: list
    - syntax: [7,1,4]

Fixed:

- In C: Array
- In Python: tuple
    - syntax: (7,1,4)

Associative:

- In C: hastables / maps
- In Python: Dictionary
    - syntax: {"duncan" : "south, "siegreid" : "mod"}

Membership:

- In C: set
- In Python: set
    - syntax: {"pho", "banh mi"}

Example: Anagrams

- **Problem:**
    - given two words, determine if they are anagrams

```
def count_letters(string: str) -> dict[str, int]:
```

```
    counts = {}

    for letter in string:
        if letter.isspace():
            continue
        try: #also could do `if letter in counts:`
            counts[letter] += 1
        except KeyError: # `else`
            counts[letter] = 1
        # You could replace the whole block within the for loop with:
        # counts[letter] = counts.get(letter, 0) + 1


    return counts

def main(arguments=sys.argv[1:], stream=sys.stdin) -> None:
    #parse command line arguments
    ignorecase = False

    while arguments:
        argument = arguments.pop(0)
        match argument:
            case '-i': ignorecase = True
            case '-h': usage(0)
            case _: usage(1)

    #process from standard input
    for line in steam:
        if ignore_case:
            line = line.lower()

        word = line.rstrip().split()
        word1 = words[0]
        word2 = words[1]
        #or you could just do `word1, word2 = line.rstrip().split()`

        if is_anagram(word1,word2):
            print('Anagram!')
        else:
            print('Not Anagram!')


def is_anagram(word1: str, word2: str) -> bool:
    counts1 = count_letters(word1)
    counts2 = count_letters(word2)
    return counts1 == counts2
```

Writing python script that shows a cow for `cowsay` at random

```
#!/usr/bin/env python3

import os
import random
import sys

#constants

FORBIDDEN = {'bang', 'sandomized', 'kiss', 'head-in', 'satanic'}
#FORBIDDEN is a set --> this allows for constant time lookup

#main execution

def main():
     for line in enumerate(os.popen('cowsay -l')):
          if not index: # equivalent to `index != 0`
               continue
          for cow in line.split():
               if cow not in FORBIDDEN:
                    cow.append(cow)

     chosen = random.choice(cows)
     message = ' '.join(sys.argv[1:])
     os.system(f'cowsay -f {chosen} {message}')

if __name__ == '__main__':
     main()
```