On your way in...

Pick-up
1. HW4, due Monday
2. POGIL Activity 14 – Reading Files
3. POGIL Activity 15 – Writing Files
Welcome to CS 134!
Introduction to Computer Science
Iris Howley
-Files-
Spring 2019
Iris is Out of Town Wednesday-Sunday

• Thursday office hours are being moved to:

**MONDAY, FEBRUARY 4, 6-8PM**
TPL205

• We still have class on Friday!
Midterm Exam is Tuesday, March 5 in Wege Auditorium (this room)

• Choose one of 2 time slots:
  1. 5:45-7:45pm
     • If your accommodations letter provides extended time, attend this session
  2. 8-10pm

• Midterm Review Session:
  • Sunday, 3/3 7-9pm in Wege

• Iris’ Office Hours:
  • Monday, 3/4 6-8pm in TPL 205
Johnson’s MATH150 or Williams’ BIOL102?

• If you’re currently taking:
  • Stewart Johnson’s MATH 150
  • Heather Williams’ BIOL 102

• Please come talk to me after class!
• (alternatively, send me an email)
Midterm Exam is Tuesday, March 5

• Closed book exam

• Covers topics up to and including this week
  • Review your homeworks!

• Next week’s lab will be experiential
NO CLASS WEDNESDAY MARCH 6
A Common Mistake

- `l = []`
- `type(l)`
  - `<class 'list'>`
- `l = l.append(2019)`
- `type(l)`
  - `<class 'NoneType'>` **What happened?**
Sort and Sorted

- \( l = [100, 75, 50, 25, 0] \)
- \( ls = \text{sorted}(l) \)
- \( ls \)
  - \([0, 25, 50, 75, 100]\)
- \( l \)
  - \([100, 75, 50, 25, 0]\)
- \( l.\text{sort}() \)
- \( l \)
  - \([0, 25, 50, 75, 100]\)

What’s the difference between \( l.\text{sort}() \) and \( \text{sorted}(l) \)?
Isograms

• Isogram: a word where each letter appears only once
  • ex: ‘duane’
  • ’iris’ is not an isogram → two ‘i’\’s!
  • ‘duane bailey’ is not an isogram

• How do we find all the isograms in the dictionary?
  • Hint: use set (...)

Take a minute to discuss with a partner
Isograms

• if len(set(s)) == len(s):
  • print("isogram: " + s)
• else:
  • print("not isogram!")

OR

• print("isogram: " + s) if len(set(s)) == len(s) else print("not isogram!")
Sets Revisited

- \( s = \{4, 3, 3, 3, 9, 1, 1\} \)
- \( s \)
  - \( \{9, 3, 4, 1\} \)
- \( s[0] \)
  - TypeError: 'set' object does not support indexing
- \( s.add(2019) \)
- \( s \)
  - \( \{1, 3, 4, 2019, 9\} \)
- \( fs = \text{frozenset}(s) \)
- \( fs \)
  - \( \text{frozenset}\{1, 3, 4, 2019, 9\} \)
- \( fs[0] \)
  - TypeError: 'frozenset' object does not support indexing
- \( fs.add(2019) \)
- \( fs \)
  - AttributeError: 'frozenset' object has no attribute 'add'

'pydoc3 set'
Sets Revisited

- $s = \{1, 2, 3, 4\}$
- $s_2 = \{1, 2\}$
- $s_2.\text{issubset}(s) \rightarrow \text{True}$
- $s.\text{issubset}(s_2) \rightarrow \text{False}$
- $s.\text{issuperset}(s_2) \rightarrow \text{True}$
- $s_2.\text{add}(99)$
- $s_2.\text{issubset}(s) \rightarrow \text{False}$
- $s.\text{union}(s_2)$
  - $\{1, 2, 3, 4, 99\}$
- $s.\text{intersection}(s_2)$
  - $\{1, 2\}$
- $s.\text{difference}(s_2)$
  - $\{3, 4, 5\}$
- $s_2.\text{difference}(s)$
  - $\{99\}$

‘pydoc3 set’
TODAY’S LESSON

Files

(reading and writing)
Files

• A persistent store between python sessions

• Existing files must be opened, read, and closed:
  - `f = open(‘tomsawyer.txt’, ‘r’)
  - `for line in f:
    - …
  - `f.close()"

  **The mode**
  - ‘r’ for reading the file (text, by default)
  - ‘w’ lets you write to the file
    (creates or deletes it first)
  - lots more options, see ‘pydoc3 open’

It’s hard to remember to close the file!
Reading Files

• `f = open('tomsawyer.txt', 'r')`
• `for line in f:`
  • `...`
• `f.close()`

• `with open('huckfinn.txt','r') as f:`
  • `for line in f:`
    • `...`
  • `# file is implicitly closed`

Fewer resource leaks!
POGIL Activity 15- Writing to Files

• Let’s step through a simplified version of Question 1
POGIL Activity 15

• Question 1K.

• How might we ask for three students’ information and write that to a file?
Writing Files

You can also specify a file to print to using the `print( .. )` function:

```python
with open('table.dat', 'w') as out:
    for a in range(1, 10+1):
        for b in range(1, 10+1):
            print('{:4}'.format(a*b), end='', file=out)
    print(file = out)
# out.close() happens here, implicitly
```

See printwrite.py and filewrite.py in examples/02.27
More About the Built-in Format Function

```python
print('{:4}'.format(a*b), end=''', file=out)
```

- `print('Num is {}'.format(3))`
  - 'Num is 3'
- `print('Num is {:10}'.format(3))`
  - 'Num is 3' → Gives 10 total characters to the {} (fills in with spaces)
- `print('Num is {:^10}'.format(3))`
  - 'Num is 3' → Center-aligns {} within those 10 spaces
More About the Built-in Print Function

\[
\text{print(’{:4}’}.format(a*b), \text{end=’’, file=out)}
\]

- **end=’’**
  - print() usually ends with a ‘\n’ by default
  - Can change that!
- **file=out**
  - Print() usually prints to std out (the terminal) by default
  - Can change that!
Leftover Slides