CPS 109 - Lab 5

Agenda

Let’s talk about file I/O (file in and out)and tuples.

File I/O

What is file I/O? Well, your programs don’t exist in a vacuum. You need to be able to communicate with lots of other files on your machine!

File I/O

How do you deal with files? First, you have to open it!

```
our_file = open("a_file.txt", "w")
```

Your first argument is the file you want to open, your second is your option for dealing with the file.

File I/O

You can do this in several ways:

```
our_file = open("a_file.txt", "w") # w means you "write to it"

our_file = open("a_file.txt", "r") # r means you "read from it"

our_file = open("a_file.txt", "a") # a means you "append to the list"
```

File I/O

Now what do we do with our file? We can do lots of things! We can write to it, we can read from it, etc.

File I/O

This is a simple example of opening a file to read, then reading from it.

```
our_file = open("a_file.txt", "r")

st = our_file.read(250)
print("The file reads: ", st)

our_file.close()
```

File I/O

This is a simple example of opening a file to read, then reading from it.

```
our_file = open("a_file.txt", "r")

st = our_file.read(250)
print("The file reads: ", st)

our_file.close()
```

Tuples

A tuple is a data structure that allows you to return or look at associated data. Careful! Tuples are immutable (for whatever reason).

Tuples

What does a tuple look like? Think of a list, but instead of [] around it, you have ().

```
return (num_lines, file_contents)
```

Tuples

Note: you don’t necessarily need only two elements. Just remember, use a tuple instead of a list if you don’t want the data associated inside the tuple to change.