**CS210: IMPORTANT CONCEPTS**

**Sep7**

print(a,p) prints 2 variables a and p with a space in between. Similarly, you can string together multiple variables, constants, and literals. Literals can be enclosed in single or double quotes.

integer result, fractional part truncated: 3 // 2 = 1

Modulus (%) can be used between real numbers as well giving real results.

Arithmetic precedence rules follow as normal.

a\*\*b means a^b. Here a and b can be +ve, -ve, integers, or real.

Converting real number to integer: int(real #)

Converting a string representing int/float to int/float: int(string) or float(string).

Math.pow(a,b) = a^b. Raises an error when ‘a’ is negative and ‘b’ is not an integer.

# multi line string with three double quotes

multi\_line = """line 1,

line2, and

line 3"""

print(multi\_line)

line 1,

line2, and

line 3

Else you can just normally use the new line character (\n).

Tab character: \t

+ concatenates strings. But you can't directly concatenate non-string parts - you need to use str function to convert to string first. When you use a number as first operand for +, it thinks you are adding (not concatenating to a string).

Formatting the output:

print("{0}/{1} = {2}".format(m,n,m/n))

print(f'{m}/{n} = {m/n}')

Rounding variable cval to 2 decimal places can be done in 2 ways:

-> print('{:.2f}'.format(cval))

-> print(f'{cval:.2f}')

Semicolon starts the formatting.

There is no double type in Python, but float is equivalent to double.

Boolean type: bool

Boolean Operators highest to lowest precedence: not, and, or

**Sep9**

**Functions:**

If there are any default arguments (i.e. parameters with default values), they must be at the end.

If multiple return values from a function, then they are returned as a tuple.

Say, a function returns 3 values. This is the way to catch them:

a, b, c = func()

Help->python reference. Here you can search for information on any python topic.

**String Methods:** instance methods of a string object.

endswith

startswith

find

isalpha

isdigit

replace(thing to replace, what to replace it with)

->if needed an extra argument specifying the number occurrences that you only want to replace from the left to the right.

strip

->as an argument you can also specify the exact leading and trailing characters you want to strip out. Default: whitespace characters. Characters are removed from the leading end until reaching a string character that is not contained in the set of characters in *chars*. A similar action takes place on the trailing end.

Replace and strip on a string return a new string without affecting the original one.

**Decisions**

if, elif, else

**Steps to writing python programs:**

**->**launch atext file

->rename as convert.py

->copy needed stuff from notebook

->ctrl S to save

->launch a terminal

->pyhton3 convert.py

**Catching errors**

try:

except:

to catch a particular kind of error ->

try:

except …..:

except takes the same scope as try. Things defined in try are still accessible in except.

Common type of error: Value Error

**Taking input from users**

Import sys

Sys.argv[0] is the name of the program

Sys.argv is a string array with all the parameters.

Input() is a blocking function. Waits until user presses enter.

While loops

**Sep 16th**

For loops

range

string multiplier

Iterating over string characters

List functions

Len(name of list)

List(string) -> makes a list of all the characters in the string

Sum(list) -> returns sum of the values in the string

Max(list)

Indexing, slicing,striding

Index -1: last element

Index -2: 2nd to last

When you slice a list, you get another list

Indexing into list within a list

Membership for element in list or character in string: in

Extend

(list object).extend(list)

->adds elements in list at the end of the list object

->the original list object changes

Append

(list object).append(element)

Adds element at the end of the list

->the original list object changes

Unpacking

A,b = [4,6]

\_,b = [4,6]

\*

Other list methods: index, insert, remove, reverse, count, sort

If you don’t want to change original list for sorting then use sorted(list)

Usually: list.method(params)

List.remove(x) throws an error if x is not in the list.

**Sep 21st**

Tuples are immutable. You cannot change the content of a tuple.

Tuple(string) -> creates a tuple with the characters in the string

Similarly, tuples can be formed from lists.

An element of a tuple could be a list.

String can be considered as a list of characters.

You can do the following with strings:

Slicing, len, find (similar to index for lists) (returns -1 if the element is not present), striding

Str.lower() does not change str -> it returns a new string

Lst = txt.split()

List Comprehension

List of list using list comprehension

Enumeration

Sorting

Lambdas

**Sep 23rd**

Dictionaries

You can consider it like a list but you index into it using the key.

Ternary Operator

Some dictionary methods

.Get -> you can specify default value

In as usual

.keys, .values, .items

Each element in dict.items( ) is a tuple.

Values can be heterogenous.

Default Dictionary

Ordered Dictionary -> behaves like a stack

Counter -> Methods: .update(list of keys whose values you want to add by 1), .most\_common

Counter can be used for strings as well.

String is an iterable like lists.

Reading and Writing Files

Sorted Function can be used for sorting dictionaries

Set is a collection with unique elements. Can be formed from a list. Results in a list with only unique elements of the input list. Set(list)

Converting a list of characters into a string:

s = ''.join(first\_letter)

you can compare dictionaries directly.

sorted\_d = sorted(d.items(), key=lambda item: (-item[1], item[0]))