Lecture 8: October 24, 2019

Reading: §8 **Assignments:** Homework 2, due on October 24 at 11:59pm

1. Lists
   (a) Sequence of values (ints, floats, strings, other lists, etc.)
   (b) Denoted by square brackets `[ ]` with values separated by commas
   (c) Lists are mutable
   (d) How to create a list
2. Lists and strings [*datecvt.py*]
3. Program to print words in a line [*lines.py*]
4. What you can do with lists
   (a) Check membership: `in`, `not in`
   (b) `+`: concatenation
   (c) `*`: repetition
   (d) `list[a:b]`: slice list from a to $b-1$
   (e) `del list[i]`: delete element `list[i]`; i can be a slice
5. Searching a list
   (a) Example use: linear search [*linsearch.py*]
6. Objects, references, aliasing
   (a) For strings, one copy: assume `a = "banana"`
      i. After `b = a` or `b = a[:]`, then `a is b` is `True`
   (b) For lists, multiple copies: assume `A = [ 1, 2, 3 ]`
      i. After `B = A` then `A is B` is `True`
      ii. After `B = A[:]`, then `A is B` is `False`
7. Lists as parameters: can change list elements in function and they are changed in caller [*args2.py*]
   (a) Add elements to, remove elements: `L.append(x)`, `L.extend(ls)`, `L.insert(i, x)`, `L.pop()`, `L.remove(x)`
   (b) Element ordering: `L.reverse()`, `L.sort()`
   (c) Other: `L.count(x)`, `L.index(x)`
8. More on parameters: named arguments and variable number of arguments [*args3.py*]
9. Tuples
   (a) Used to group data
   (b) Like lists, but immutable
10. `isinstance(obj,type)` function
    (a) `type` is `bool`, `float`, `int`, `list`, `str`, `tuple`