List (and strings) review

• How are lists represented in memory?
  – primitive types are stored directly
  – complex types, like lists, are stored indirectly

• What happens when we assign a variable to a list? (in memory)
  – only the memory address is assigned; the list is not copied

• How is a string similar to a list? Different?
  – both can be indexed (because they are ordered)
  – you can take a sub-list of sub-string
  – lists can be changed, strings cannot be updated after assignment.
Lists in python

def function():
    #YOUR CODE GOES BELOW HERE, MAKE SURE YOU INDENT!

    hybrids = ["puggle", "liger", "mule"]
    hybrids.append("beefalo")
    hybrids[1] = "wolf dog"

    #DO NOT WRITE CODE BELOW THIS LINE

print function()

• `.append()` is a way to add a single element to the end of a list
length in python

```python
def function():
    #YOUR CODE GOES BELOW HERE, MAKE SURE YOU INDENT!
    hybrids = ["puggle", "liger", "mule"]
    counter = 0
    while counter < len(hybrids):
        print hybrids[counter]
        counter = counter + 1

    #DO NOT WRITE CODE BELOW THIS LINE

print function()
```

- **len( )** is a function to return the length of a list or string
python memory visualizer

• let’s use the python memory visualizer to trace through an example:

http://pythontutor.com/visualize.html#code=x+%3D+1%0Asmall+%3D+%5Bx,+2,+3%5D%0Amedium+%3D+%5B43,+76,+180%5D%0Alarge+%3D+%5B234%5D%0AbigList+%3D+%5Bmedium,+small,+large%5D%0Aprint+small%0Aprint+bigList%0Ax+%3D+7%0Aprint+small&mode=display&cumulative=false&heapPrimitives=false&drawParentPointers=false&textReferences=false&showOnlyOutputs=false&py=2&rawInputLstJSON=%5B%5D&curInstr=0

Note the difference between the storage of x, a primitive, and the lists.
Note you’ll need to check if the element is in the list or string before trying to get its index.
in

```python
def function():
    string = "CS112 Intro to programming"
    print ("I" in string)
    print ("Intro" in string)
    print ("cat" in string)

    list = [5, 7, 9, 11, 15]
    print (7 in list)
    print (3 in list)

    print function()
```

- `in` is a keyword and can be used to check if the element is in the list or string before trying to get its index
substrings and sub-lists with [: :]
newlines and tabs

def function():
    #YOUR CODE GOES BELOW HERE, MAKE SURE YOU INDENT!
    string = "CS112:\n\tIntro to programming"
    print string

print function()
Other useful functions

- \texttt{min}(list)
- \texttt{max}(list)
- \texttt{list.remove}(1.5)
- \texttt{sortedList} = \texttt{sorted}(list)
Strings in Python

• ‘cat’ and “cat” are the same
  – can use single or double quote
  – default is single quote
  – useful for “’” versus ‘”’