INTRO TO PROGRAMMING

Basic Data Structures

DATA STRUCTURES

- List - an ordered sequence of dynamic data
- Tuple - an ordered sequence of data that can't be changed (immutable)
- Dictionary - a key-value store of data, basically a collection of variables where each name is a "property" of the dictionary

LISTS

```
schedule = [
    'Algorithms and Data Structures',
    'Computer Operating Systems'
]

# you can overwrite elements similarly
schedule[1] = 'Computational Geometry'

# or delete
del schedule[0]

# and add to the end without knowing
# how many items are in the list
schedule.push('Computer Graphics')
```

LISTS

```
# you can access elements by their index
firstClass = schedule[0]

# you can also get the length of a list
len(schedule)  # results in 3

# iteration, variable class gets set to each value
for class in schedule:
    print(class)

# can pass index of first element (inclusive)
# and index of last element (exclusive)
# to get a 'slice'
technicalClasses = schedule[0:2]
```

LISTS

- Also known as an *array* in other languages
- Each element can be referenced by its position in the list known as its *index*
- Usually stored contiguously in memory

TUPLES: IMMUTABLE LISTS

```
# can't change history, so a tuple makes sense
earlyPresidents = (
    'George Washington',
    'John Adams',
    'Thomas Jefferson',
    'James Madison')

earlyPresidents[0] # results in 'George Washington'
```

TUPLES

- Tuples have all the same methods as lists

DICTIONARY

```
student = {
    'name': 'Jack',
    'age': 20,
    'classes': ['Physics', 'Literature', 'CS']
}

# Can refer to properties by their key
# NOTE: can't use dot operator like other languages
lookupName = student['name']
```

ITERATION

- explicitly specifying the element you want works when working with small bits of data
- for larger datasets or dynamically populated information you need a generic way to process
- python provides tools for iterating over these data types

ITERATION

```
subjects = ['Math', 'Science', 'CS']

for elem in subjects:
    print(elem + " is ")
    if elem == 'CS':
        print("my favorite class")
    else:
        print("pretty cool")

i = 0
while i < len(subjects):
    print("I still have homework in: ", subjects[i])
```

NUMERICAL PROBLEMS

- random walk
  - conditionals to determine direction from random float
  - store each state in the walk
  - add arguments for how many steps, how many runs
  - write function to compute distance