**Outline for October 13, 2020**

**Reading:** §3, 5

**Assignments:** Homework 1, due October 15, 2020

1. Office hours are Wednesday and Friday at 11:00am–11:50am Pacific time; use the class Zoom link.

2. Simultaneous assignment [*swap.py*]
   (a) Simple assignment: `variable = expression`
   (b) Simultaneous assignment: `variableA, variableB = expressionA, expressionB`

3. Decision structures [*if0.py*]
   (a) `If` statement

4. Conditions
   (a) Resolves to boolean value
   (b) Literal booleans: `True, False`
   (c) Relational operators
      i. Use two arithmetic expressions connected with relational operators to create a boolean
      ii. Relational operators: `>`, `>=`, `<`, `<=`, `==`, `!=`
      iii. Precedence: resolved after arithmetic operators
      iv. `6 > 2 + 3; "UCD" == "Sac State"`

5. Two-way decisions [*if1.py*]
   (a) `if ... else` statements
   (b) `else` very powerful when the positive condition is easy to describe but not the negative

6. Multi-way decisions [*if2.py*]
   (a) Can execute code based on several conditions
   (b) `elif` (else if)
   (c) `else` only reached if all previous conditions false
   (d) Compare to nested if statements

7. Iteration
   (a) Definite loops: execute a specific (definite) number of times
   (b) Indefinite loops: execute until a general condition is false

8. For loops
   (a) General form: `for i in iterator`
   (b) `Iterator` is either list or something that generates a list
   (c) Very common form: `for i in range(1, 10)`

9. `range()` in detail [*for.py*]
   (a) `range(10)` gives 0 1 2 3 4 5 6 7 8 9
   (b) `range(3, 10)` gives 3 4 5 6 7 8 9
   (c) `range(2, 10, 3)` gives 2 5 8
   (d) `range(10, 2, -3)` gives 10 7 4

10. While loops [*while.py*]
    (a) Repeats until condition is false