

## Outline for January 23, 2019

### Reading: §4

---

1. Functions [*hello.py*]
  - a. What functions are
  - b. Defining them
  - c. Using them
2. Quick look at using them [*quad.py*]
  - a. Passing values to functions
  - b. Returning values from functions
3. In more detail: passing values to functions [*args.py*]
  - a. Formal parameters in subject definition
  - b. Actual parameters (arguments)
  - c. Matching arguments to formal parameters
  - d. Local variables
4. In more detail: how Python does function calls [*quad.py*]
  - a. Caller suspends execution at point of call, remembers where it left off
  - b. Formal parameters assigned values from actual parameters
  - c. Execute function body
  - d. Return control to where caller left off
5. Refactoring code
  - a. Compute the perimeter of a triangle [*peri0.py*]
  - b. Collapse similar statements: make the distance between 2 points a function [*peri1.py*]
  - c. Collapse similar statements: make the prompts a function [*peri2.py*]
  - d. Refactor for clarity only: make the perimeter computation a function [*peri3.py*]
  - e. Add error checking: “peri0.py” done right [*peri-c.py*]
6. Add error checking: “quad.py” done right [*quad-c.py*]