

# CSI 32 LECTURE NOTES (Ojakian)

## Topic 5: Functions in Python and C++

---

### OUTLINE

PRIMER: 7.1, 7.2.1 (skip: pointers, references)

TRANSITION GUIDE: 5.5

#### 1. Functions

---

##### 1. Functions

- (a) C++: specify return type. Python: not.
- (b) C++: specify input types. Python: not.
- (c) Do some very simple functions for both.
- (d) Note - void return

##### 2. C++ Function Declaration versus Definition

- (a) C++: Can declare before defined.

**PROBLEM 1.** *In both Python and C++ write a main and a function call, putting the definition before and after the main to see what happens.*

##### 3. Pass-by-Value versus Pass-by-Reference

Two general ways an input is given to a function

- (a) Pass-by-Value: The input is copied into the function, so changes made by the function do NOT affect outside variable
- (b) Pass-by-Reference: Just a “reference” or “name” to the input is passed to the function, so internal changes made by the function DO affect outside variable.
- (c) NOTE: In any case you may always RETURN a new value of your choice!
- (d) C++ (**in standard usage with no modifiers!**):
  - i. No changes in function persist outside the function
  - ii. The variables in the function have no connection to the ones outside in terms of modifying
- (e) Python (there are no modifiers!):
  - i. Changes to int, str, bool, float in the function have NO affect on the outside variables
  - ii. More generally: If any input variable is simply re-assigned, this has NO affect on the corresponding outside variable
  - iii. BUT: Any and every change to the internal data of any object (except int, str, bool, float), DOES change the internal of the corresponding outside variable.
- (f) NOTE: In any case you may always RETURN a new value of your choice!