Introduction to Computer Programming

Lecture 14
Functions, Modules
Variable Scope

• **Scope**: the part of a program in which a variable may be accessed
• Scoping is necessary for modularization
  – Prevents confusion and errors
  – Controls the boundary between the function and the “outside”
• A variable scope is either global or local to a function
Variables Local to a Function

```python
def show_sum(num1, num2):
    tmp = num1 + num2
    print(tmp)
```

Parameters are local to `show_sum`

Other variables local to `show_sum`
Variables Global to All Functions

• A variable has a **global scope** if it is created outside of any function.

• A global variable can be accessed in any function after it is created.

• A global variable **cannot be changed** in any function (unless it is declared as global).
Global Variable Example

```python
value = 10

def show_value() :
    print(value)

show_value()
```
Global Variable Example

```python
number = 0  # global variable

def show_number() :
    print("The number is:", number)

def main() :
    global number
    number = int(input("Enter a number:"))
    show_number()

main()
```
Warnings about Global Variables

• Global variables make debugging difficult.
  – Have to look at all places where variable is defined
• Functions that use global variables cannot easily be used in other programs.
• Global variables make programs hard to understand.
  – Have to know all places where variables are modified.
Writing Your Own Value-Returning Functions

• To write a value-returning function, you write a function and add one or more return statements
  – Format: return expression

```python
def add_nums(num1, num2) :
    ans = num1 + num2
    return ans
```
Writing Your Own Value-Returning Functions

```python
result = add_nums(5, 7)

def add_nums(num1, num2):
    ans = num1 + num2
    return ans
```
Adventure Game 2
Find the pot of gold

Foyer  Parlor
---    ---
|      |      |
|      |      |

Cellar  Kitchen  Bedroom

Vault
Adventure 2 with Functions

main()

wander(start_room)

foyer()
parlor()
kitchen()
bedroom()

get_direction()
Create your own Module

• Create a file which is a collection of functions
• File should be saved with a mymodule.py extension
• You can import your module:
  ```python
  import mymodule.py
  ```
• To call a function:
  ```python
  area = mymodule.compute_area(4, 3)
  ```
Commenting Functions

• Every function should have a comment describing what the function does

  – The comment should enable a programmer to be able to use the function without reading the function code.

  – The comment should describe every parameter, its expected type and usage