```
int getRandomNumber()
{
    return 4;  // chosen by fair dice roll.
               // guaranteed to be random.
}
```

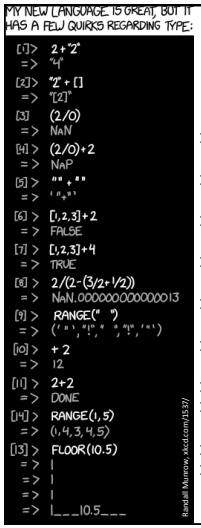
Randall Munroe xkcd.com/221

CSE 111 Bio: Program Design I
Lecture 6: Functions

Tanya Berger-Wolf (CS) & Boris Igić (Bio)
University of Illinois, Chicago
September 13, 2016

Announcements

- Prof. Berger-Wolf office hours changed to Tuesdays 3:30-4:30 (1136 SEO)
- **Engineering resume Expo**
  Friday, Sept 16, 9am-12pm
  Student Center East Tower Rm 302
  ecc.uic.edu



Types of data in Python

```
>>> goodNum = 42
```
← Integer

```
>>> pi = 3.1415926
```
← "Floating point" number

```
>>> special = [2.718, 3.141, 42]
```
← List

```
>>> okFood = "spam"
```
← String (" or 'work)

```
>>> greatFood = 'chocolate'
```

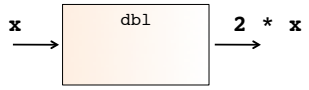
```
>>> file = open("HBB.txt", "r")
```
← Text file

```
>>> 5 > 6
>>> False
```
← Boolean

```
>>> 'spam' != 'chocolate'
>>> True
```

Defining your own functions!

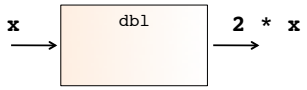
```
def dbl(x):
  return 2 * x
```



Notice the indentation. This is done using "tab" and it's absolutely necessary!

Functions can have more than one line

```
def dbl(x):
  return 2 * x
```




```
def dbl(myInput):
  myOutput = 2 * myInput
  return myOutput
```

= is **not** "symmetric"!

```
2 * myInput = myOutput
```

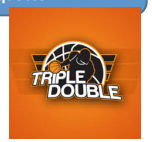
Docstrings

```
def dbl(x):
  """This function takes a number x as input
  and returns 2 * x """

  return 2 * x
```

Single quotes and double quotes are both fine! **But convention is triple double quotes**

This is sort of like teaching your program to talk to you!



Python Demo

Comments

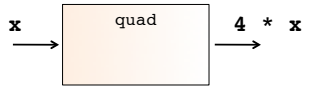
```
# Doubling program
# Authors:  Boris and Tanya
# Date:  September 13, 1960

def dbl(x):
  """ This function takes a number x as input
  and returns 2 * x """
  # Comments begin with a hash mark...
  return 2 * x
```

Natasha

Composition of functions

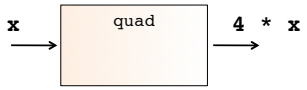
```
def quad(x):
  return 4 * x
```



How would you reuse dbl(x)?

Composition of functions

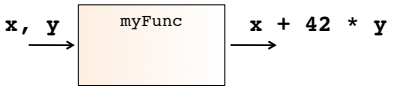
```
def quad(x):
  return 4 * x
```




```
def quad(x):
  return dbl(dbl(x))
```

Doubly cool!

Multiple inputs...




```
# myFunc
# Authors:  Boris and Tanya
# Date:  September 13, 1960

def myFunc(x, y):
  """returns x + 42 * y"""
  return x + 42 * y
```

myFunc(3, 2)

A. 3, 2
B. 3, 84
C. 90
D. 87
E. No clue

Mapping with Python...

```
def dbl(x):
  """ returns 2 * x """
  return 2 * x

>>> map(dbl, [0, 1, 2, 3, 4, 5])
[0, 2, 4, 6, 8, 10]

def evens(n):
  myList = range(n)
  doubled = map(dbl, myList)
  return doubled
```

Docstrings would help

…or alternatively

```
def evens(n):
  return map(dbl, range(n))
```

`Mapping` with Python...

```
def plusone(x):
  """ returns x + 1 """
  return x + 1

>>> map(plusone, [1, 2, 3, 4])
```

A. [1, 2, 3, 4, 5]
B. 11
C. [1, 2, 3, 4, 1]
D. [2, 3, 4, 5]
E. None of the above
F. No clue

`Mapping` with Python…

```
def add(x, y):
  """ returns x + y """
  return x + y

>>> map(add, [1, 2, 3, 4])
```

A. [3, 5, 7]
B. [3, 7]
C. 10
D. None of the above
E. No clue