

15-112 Fundamentals of Programming

Lecture 3 – Language basics

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Announcements

- First assignment has been posted. Due date is January 21 at 10:00pm
- Please get on Piazza and start contributing

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Recap of last class

- ❑ What do we know so far?
 - Draw shapes using Turtle library
 - Repeat a set of statements
 - Define functions that perform specific tasks

What do computers do?

- ❑ Get input
- ❑ Process the input
- ❑ Generate output

What do we need to learn?

- ❑ We will learn instructions in Python for:
 - Asking the user for input
 - Reading input
 - What can be sources of this input?
 - Processing input – What kind of operations we can do on this input and how to do it
 - Printing output
 - What could be possible destinations for this output

Printing in Python

- ❑ Print a word or sentence
 - `print ("what ever you want to print")`
 - `print ('I want to print this statement')`
 - `print ('I want to print "quotes" here')`
- ❑ Print multiple sentences on same line
 - `print ("This is a line" , " on same line")`

Comments

- ❑ Any text in python with a # symbol is ignored until the end of that line

```
# I will print my name
```

```
print ("Saquib Razak") # this line prints my name
```

- ❑ Comments are used to document your code
- ❑ Puts lots of comments that explain what you are doing

Getting input from keyboard

- ❑ Getting input


```
a = input()
```
- ❑ Using input with a message


```
a = input("Please Enter your name")
```
- ❑ Reading Integers


```
a = int(input("Enter your age"))
```

Variables

- ❑ We saw the following code:

```
a = input()
```

- ❑ In this example what is -- a?

- a is a variable
- You store a value in a variable
- You can read the value stored in a variable

```
a = 5
```

```
print (a)
```

```
print (a + 6)
```

```
b = a - 3
```

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

So what exactly are variables?

- ❑ All information in computers is stored in memory.

- What is memory?

- ❑ Variables are ways of accessing memory

جامعة كارنيجي ميلون في قطر
Carnegie Mellon Qatar

Variables

□ Rules for variable names

- Must begin with a letter or underscore
- May include letters, digits, and underscores
- Sin(x) is not a valid variable name

Types

□ Values in Python can be of different types

- int
- float
- string
- boolean

Operations

□ Following are some of the operations

- +, -, *, /, //, %
- **
- =
- <, >, <=, >=, ==, !=
- and, or, not
- <<, >>

Examples

- `print (3 * 2)`
- `print (3 + 2)`
- `print ("abc" + "def «)`
- `print (3 + "def «)`
- `print 2+3*4`
- `print 9**1/2`
- `print 9**1//2`
- `print ("20/3 =", (20//3))`
- `print (" 6/3 =", (6/3))`

More Examples

- `a = 5`
 `print (a)`
- `print (5 < 8)`
- `print (8 < 5)`
- `print (8 == 8)`

More Examples

- `print (8 != 8)`
- `a = 5`
 `b = 6`
 `print (a < b)`
- `print (5 / 0)`
- `print (0 / 5)`

Variables in Expressions

❑ Assign value to a variable

- `age = 21`

❑ Change a variables value

```
age = 21
```

```
print ("You are " , age * 12 , " months old")
```

```
age = age + 1
```

```
print ("You will be " , age * 12 , "months after 1  
year")
```

Variables in Expressions

```
radius = 3.1
```

```
pi = 22/7
```

```
area = pi * radius**2
```

```
print (area)
```

Bitwise Operators

- ❑ Bits and Bytes – What are these?
 - Read handout given at the end of class
- ❑ Binary Numbers
- ❑ Bitwise Operators
 - & (Bitwise AND)
 - | (Bitwise OR)
 - ^ (Bitwise XOR)
 - <<
 - >>

Bitwise Operators: Examples

- ❑ $6 \& 5$
- ❑ $6 | 5$
- ❑ $6 \wedge 5$
- ❑ $6 \ll 1$
- ❑ $6 \ll 2$
- ❑ $6 \gg 1$

Operator Precedence

❑ Operator precedence (highest to lowest):

- **
- Positive, negative, NOT (+x, -x, ~x)
- *, /, %, //
- +, -
- >>, <<
- & (Bitwise AND)
- ^ (Bitwise XOR)
- | (Bitwise OR)

❑ Operators with same precedence are processed left to right

Operator Precedence Examples

❑ print (3 + 4 * 2 + 5)

❑ print (3 * 2 + 2 / 5)

❑ print (-2 ** 4 + 8 >> 2)

Let's work out a problem

- ❑ Write a program that reads current temperature from the user in Fahrenheit and prints the equivalent Celsius value.

Another Example

- ❑ Write a program that reads an integer from the user and prints the sum of its digits.

Approximating Floats

What is the output of the following code?

```
d1 = 0.1 + 0.1 + 0.1
d2 = 0.3
print (d1 == d2)
```

Short Circuit Evaluation

❑ Let's try the following code:

```
x = 0
y = 0
print ((y == 0) or ((x/y) == 0))
print (((x/y) == 0) or (y == 0))
```

Short Circuit Evaluation

□ How about:

```
x = 0
```

```
y = 0
```

```
print ((y > 0) and ((x/y) == 0))
```

```
print ((y == 0) and ((x/y) == 0))
```

Strings

□ Any sequence of characters enclosed within “ ” or ‘ ’ is a string

- “This is a string”
- ‘this is also a string’
- “this is not a string – can you guess why?”
- ‘7his 1s a \$tring’
- “%^%\$#@!*(*&^& - what did you say?”

Indexing and Slicing

- ❑ Used to manipulate information in a string

```
name = "Chris Myers"
```

0	1	2	3	4	5	6	7	8	9	10
C	h	r	i	s		M	y	e	r	s

```
print (name[2:4])
```

```
print (name[:4])
```

```
print (name [3:])
```

```
print (name[:])
```

Math functions

- ❑ `print (math.sqrt(5))` **Does not work**

- ❑ `import math`

```
print (math.sqrt(5))
```

- ❑ `math.log(x[, base])`

- ❑ `math.cos(x)`

- ❑ `math.sin(x)`

- ❑ `math.tan(x)`

- ❑ `math.pi`

- ❑ `math.e`