ITI 1120 Intro to Computing

Prof. Waessim EL

Winter 2021

Lecture n° 2.

ITI Intro to Computing

Why is programing fun?

- It's a creative process
- A single person can build software that is as complicated as the space shuttle system.
- there is a large open source system (people who create software in their free time, and distribute it for free

formal and natural languages

- Natural languages: how we naturally speak, not designed by people, but rather evolved naturally (ambiguous, we use context clues + other info to deal with this full of metaphors + idioms)
- Formal languages: designed by people for specific applications. Have strict syntax rules (tokens & structure)

  ex: 3+3 = 6 vs. 3$ = (not ambiguous: each $ has one meaning).
  H_2O vs. $_2H^o$
- Programing languages are formal languages.

What is a programing language?

(Java, python, C, C+, pearl, (High level languages))

- advantages: easier (less time, shorter easier to read) more likely to be correct, portable.
- disadvantanges: have to be translated before running (compiling or interpreting)
- almost all programs are written in them.

Low level languages: machine assembly

- disadvantages: only runs on one kind of computer. Have to be rewritten on another computer
- Only used for a few special applications (applications where a life is at risk, ie. elevator)

Interpreting / compiling

- There are 2 ways to translate a program written in a high level language. (source code)
- Interpreting: a program that reads a high-level and what it says.
- Compiling: a program that reads a high-level program and translates it all at once before running it

  Compiles a program first and then runs the compiled code later. The compiled code is called executable.

Algebric Expressions

- The python interactive shell can be used to evaluate algebric expressions
- 14//3 is the quotient when 14 is divided by 3 and 14%3 is the remainder
- 2**3 = 2^3
- abs(), min(), and max() are functions
- abs() takes a number as input and returns its absolute value.
- min(), max() takes arbitrary number of inputs and return the "smallest" or "largest" among them.
- method = function. absolute apps function takes an integer and makes it an absolute value.

Lecture nº2

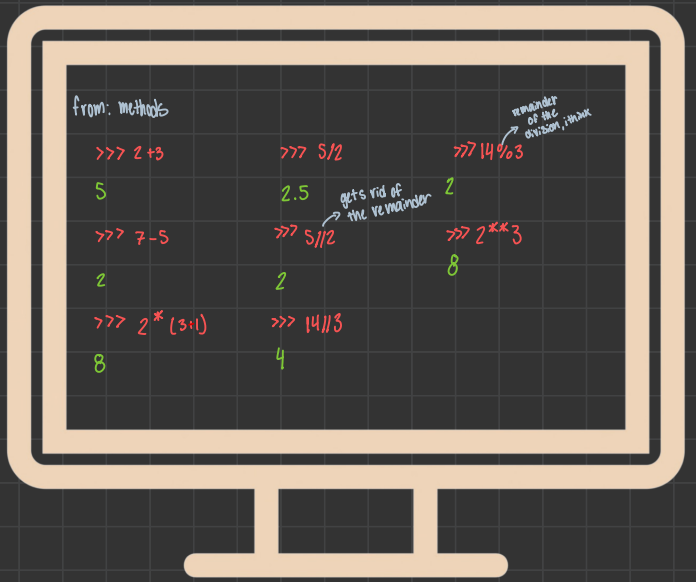
Boolean Expressions

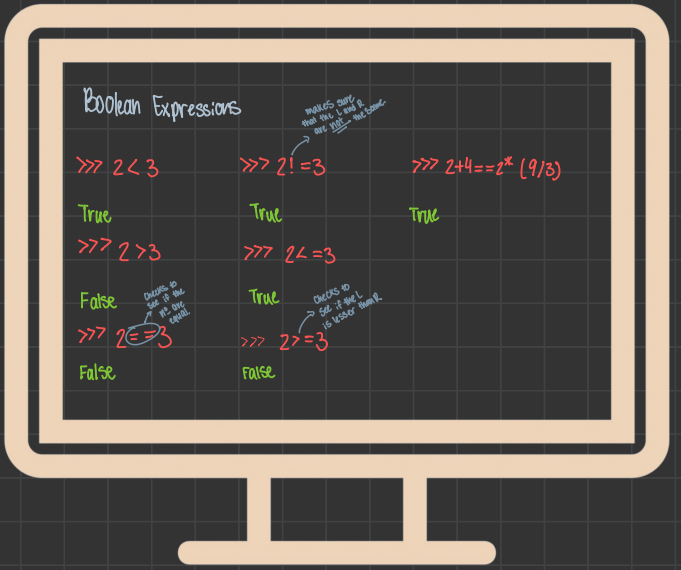
- Evaluates True or False
- Involves comparason operators

  <, >, ==, !=, <= and >=

In an expression containing algebraic + comparason operators:

- Algebraic operators are evaluated first
- Comparason operators are evaluated next.





Lecture nº2

Boolean Operators

- In addition to algebric expressions, python can evaluate Boolean expressions
  - Boolean expressions look at True or False
  - Boolean expressions may include Boolean operators and, or, and not
- In an expression containing algebric, comparison, and Boolean operators:
  - Algebric expressions are evaluated first
  - Comparason operators are evaluated next
  - Boolean operators are evaluated last

```
>>> 2>3 and 3<4
True
>>> 4==5 and 3<4
False
>>> False and True
False
>>>True and True
True
>>> 4==5 or 3>4
True
>>> False or True
True
>>> False or False
False
>>> not (3>4)
False
>>> not (True)
False
>>> not (False)
True
>>> 4+1==5 or 4-1<4
True
```

Representation of numbers in Python

- Integers can be represented exactly. There is no limit to their size (other than the size of your memory)
- Floating points are approximation of real numbers. Python uses a double-precision standard format (IEEE 754) That can represent real numbers in a range of 10^{-308}.

Operations: -, **, *, /, %, //, +, <, <=, ==, !=, >, >=

* == Should be avoided in floats

Lecture nº 2

Number type operators

Operation	Description	Type (if x and y are integers)
$x + y$	Sum	Integer
$x - y$	Difference	Integer
$x * y$	Product	Integer
x / y	Division	float
$x // y$	Integer division	Integer
$x \% y$	Remainder of x/y	Integer
$-x$	Negative x	Integer
abs(x)	Absolute value of x	Integer
$x ** y$	x to the power y	Integer

- Number type operators. Listed are the operators that can be used on number objects (e.g. boolean, int, float)
- If one of the operands is a float the result will always be a float
- if one of the operands is not a float, the result will be an int. Except for the division (/) operator, which always gives a float value.

Operator Precedence

Operator	Description
[expressions...]	List definition
x[], x [index:index]	Indexing operator
**	Exponentiation
+x, -x	Positive, negative signs
*, /, //, %	Product, division, integer division, remainder
+, -	Addition, subtraction
in, not in, <, <=, >, >= <>, !=, ==	Comparisons, including membership and identity tests
not x	Boolean NOT
and	Boolean AND
or	Boolean OR

- The operators are listed in order of precedence from highest on top to lowest at the bottom
- The ones on the same row have the same importance
- Higher-Precedence operations perform first
- the ones on the same level perform from left to right

Lecture 2

Variables and Assignments.

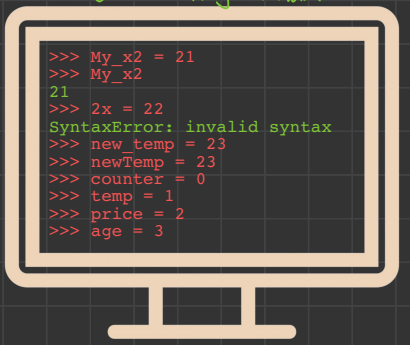
- Just like in algebra, a value can be assigned to a variable, such as x
- When x appears inside an expression, it evaluates it as its assigned value.
- A variable (name) does not exist until it is assigned
- The assignment statement has the format

<variable> = <expression>

(resulting value is assigned to) ← (evaluated first)

Naming Rules

- Variable names allowed:
  - A-Z
  - a-z
  - '_'
  - nº 0-9
- names cannot start with a digit (won't print
- multiple names:
  - either use the '_' as the delimiter
  - Or camel case capitalization
- * Shorter and meaningful names are best! *



The End!