

Computer Science 134C

Introduction to Computer Science, in Python

Lecture #4 (Booleans and Conditionals)

September 14, 2018

Keywords

bool, if, interface, lists, None, pass,
refactoring, while,

We think carefully about how decisions are made in programs.

1. Questions?
2. Review where we are.
3. Functions.
 - (a) Catching up with end of Lecture 3.
 - (b) Gathering results into lists:
 - i. Lists are ordered collections between brackets ([3, 5, 4]).
 - ii. Elements of lists can be referenced by index, starting at 0 (1[3]).
 - iii. Lists can be concatenated through addition (11 + 12).
 - iv. The length of a list is determined by len(l).
 - v. Many other operations (peek ahead to Chapter 10!).
 - (c) Computing the orbit of the syr function: return as a list.
 - (d) Generalizing orbit.
 - (e) Notions of *interface*, *refactoring*, *reuse*, and *interface*.
4. Careful thought about conditions.
 - (a) Basic operations: generating boolean values (e.g. word in wordList).
 - (b) Combining boolean values: and, or, and not.
 - (c) The boolean nature of non-boolean values.
 - (d) The *sameness* of objects, the meaning of is, and the solitary None.
 - (e) Source of midterm questions: simplifying boolean expressions.
5. Full details on if.
 - (a) The if-then-else version.
 - (b) The if-then-elif-elif-...-else version.
 - (c) Nesting.
 - (d) Suites and the pass statement.
6. The while loop, break, and continue.