Python and Computer Memory

Computer Memory

For the purpose of this course, you may think of *computer memory* as a long list of storage locations where each location is identified with a unique number and each location houses a value. This unique number is called a *memory address*. Typically, we will write memory addresses as a number with an "id" as a prefix to distinguish them from other numbers (for example, `id201` is memory address 201).

Variables are a way to keep track of values stored in computer memory. A *variable* is a named location in computer memory. Python keeps variables in a separate list from values. A variable will contain a memory address, and that memory address contains the value. The variable then refers to the value. Python will pick the memory addresses for you.

Terminology

A value has a memory address.
A variable contains a memory address.
A variable refers to a value.
A variable points to a value.

Example: Value `8.5` has memory address `id34`.
Variable `shoe_size` contains memory address `id34`.
The value of `shoe_size` is `8.5`.
`shoe_size` refers to value `8.5`.
`shoe_size` points to value `8.5`.

Extra readings:

- [Chaper 2.4. Variables and Computer Memory: Remembering Values]
- [*Optional reading*]

Jennifer Campbell • Paul Gries
University of Toronto