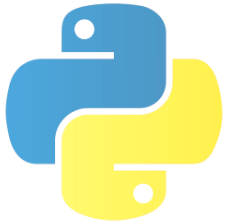


Syntax Revisited

Williams College
CS334 - S09
Prof. McGuire



Python Features

- Eager
- Let

```
(let ([x null])  
  (set! x 3)
```
- Res
- ations!

```
(let ([x null])  
  (set! x 4)  
  x)
```
- N
- x)
- Array/Table-centric library and syntax

Python Syntax

Anonymous closure creation:

lambda [*id* (, *id*)*] : *exp*

Procedure declaration:

def *id* ([*id* (, *id*)*]):
exp⁺

Application:

exp (*exp**)

Iteration:

for *exp* **in** *exp* :
exp⁺

Conditional:

if *exp* :
exp⁺
[else:
exp⁺**]**

Assignment:

id = *exp*

[*exp* .]* *id* = *exp*

Return (*only in procedure*):

return *exp*

Spaces and newlines indicate block and expression extent.

Special syntax for constructing arrays, strings, tuples, and tables.

Case Study: a[i]

Java

$$\frac{\begin{array}{l} \text{exp}_a \Rightarrow \text{Array}\langle v_0, v_1, \dots, v_{n-1} \rangle \\ \text{exp}_i \Rightarrow i \in \{0 \dots n - 1\} \end{array}}{\text{exp}_a[\text{exp}_i] \Rightarrow v_i}$$

C++

$exp_a[exp_i] \Rightarrow exp_a.operator[] (exp_i)$

array[5] = true table["course"] = 334

$$\frac{exp_a \Rightarrow Array\langle v_0, v_1, \dots, v_{n-1} \rangle \quad exp_i \Rightarrow i \in \{0 \dots n - 1\}}{exp_a[exp_i] \Rightarrow v_i}$$
$$\frac{exp_a \Rightarrow Array\langle v_0, v_1, \dots, v_{n-1} \rangle \quad exp_i \Rightarrow i \notin \{0 \dots n - 1\}}{exp_a[exp_i] \Rightarrow \text{unspecified}}$$

(The Array behavior here is for `std::vector`, which is in the standard library but not part of the C++ language specification. Many programmers use their own array classes, like `G3D::Array`, that ascribe semantics closer to Java's)

Python

$$\frac{\neg(\text{exp}_a \Rightarrow \text{Array}\langle v_0, v_1, \dots, v_{n-1} \rangle)}{\text{exp}_a[\text{exp}_i] \Rightarrow \text{exp}_a.\text{get}(\text{exp}_i)}$$

$$\frac{\text{exp}_a \Rightarrow \text{Array}\langle v_0, v_1, \dots, v_{n-1} \rangle \quad \text{exp}_i \Rightarrow i \in \{-n \dots n - 1\}}{\text{exp}_a[\text{exp}_i] \Rightarrow v_i \text{ mod } n}$$

if a[-1]="~": print a + "is a temp file"

$$\frac{\text{exp}_a \Rightarrow \text{Array}\langle v_0, v_1, \dots, v_{n-1} \rangle \quad \text{exp}_L \Rightarrow i \in \{-n \dots n - 1\} \quad \text{exp}_H \Rightarrow i \in \{-n \dots n\}}{\text{exp}_a[\text{exp}_L : \text{exp}_H] \Rightarrow \text{Array}\langle v_L \text{ mod } n \dots v_{(H-1) \text{ mod } n} \rangle}$$

if a[-3:0]=".jpg": print a + "is an image file"

(Array is called List in Python)

C

$$\text{exp}_a \Rightarrow a \in \tau \qquad \text{exp}_i \Rightarrow i$$

$$\text{exp}_a[\text{exp}_i] \Rightarrow \text{Read sizeof}(\tau) \text{ bytes from memory location } a + \text{sizeof}(\tau) * i \text{ and interpret them as a value of type } \tau.$$