Extended Introduction to Computer Science
CS1001.py

Module A:
Grammars and Python's Syntax

Instructors: Amir Rubinstein, Michal Kleinbort
Teaching Assistants: Noam Parzanchevsky,
Hussen Abu Hamad, Asaf Cassel, Shaked Dovrat

School of Computer Science
Tel-Aviv University
Fall Semester 20-21
http://tau-cs1001-py.wikidot.com

Syntax vs. Semantics

- Syntax: the form of a valid program
  - Every language has its syntax
  - Example: print("abc") is a valid form

- Semantics: the meaning of the program and the expected results of executing it
  - Example: print("abc") will print the string inside the brackets to the screen

Specifying a Syntax

- The syntax of a programming language is formally defined using a Grammar
  - Similar to the case of Natural Language, yet with much less irregularities

- Reading Grammars takes some getting-used-to, but is not hard

- Before evaluating your program, the interpreter (e.g. IDLE) verifies that it conforms to the Grammar

Specifying Semantics?

- Much more cumbersome to do formally, and we will not cover this in the Intro course

- You will see a bit of that in the 3rd year Compilation course, and more if you study electives related to Software Verification

Grammar

- Grammar is defined by the following:
  - The alphabet of the language
  - A set of variables representing types of phrases or clauses in the sentence
  - The set of rules of the grammar

Example #1

<SENTENCE> → <NP> <VERB>
<NP> → <ARTICLE> <NOUN>
<NOUN> → boy | girl | cat
<ARTICLE> → a | the
<VERB> → touches | likes | sees

Which strings can be formed using this grammar?

<SENTENCE> → <NP> <VERB>
→ <ARTICLE> <NOUN> <VERB>
→ a <NOUN> <VERB>
→ a girl <VERB>
→ a girl likes

Example #1

<SENTENCE> → <NP> <VERB>

<NP> → <ARTICLE> <NOUN>

<NOUN> → boy | girl | cat

<ARTICLE> → a | the

<VERB> → touches | likes | sees

Which of the following strings can be formed using this grammar?

“a girl likes” ✓

“the boy sees” ✓

“the girl likes the cat” ✗

Example #2

<S> → a<S>a | b<S>b | c

Which strings can be formed using this grammar?

<S> → a<S>a

→ aa<S>aa

→ aab<S>baa

→ aabcbaa

Example #2

<S> → a<S>a | b<S>b | c

Which strings can be formed using this grammar?

<S> → c

Example #2

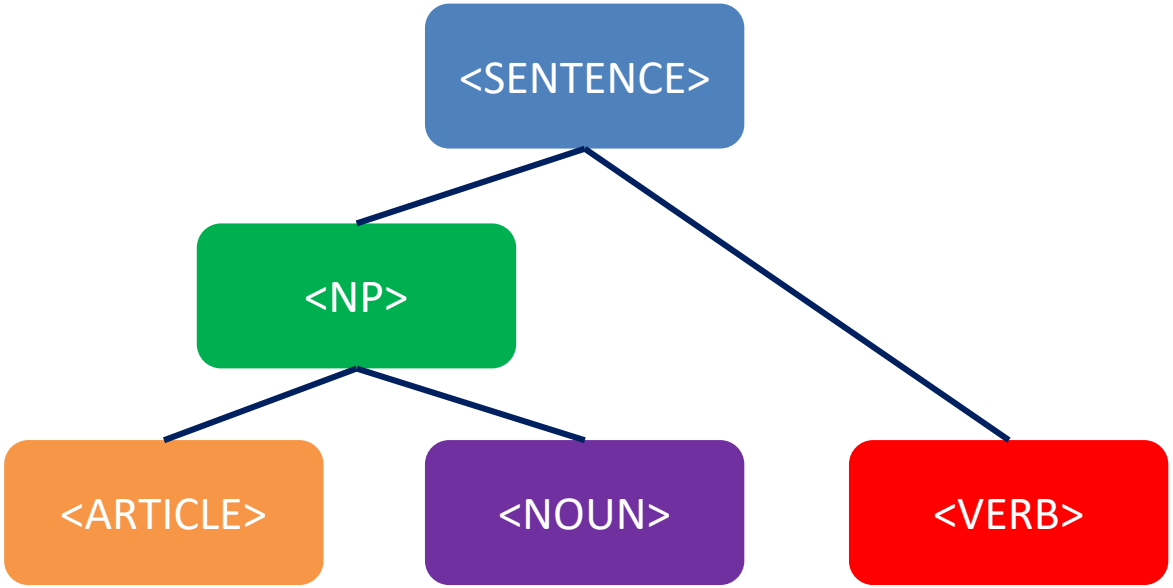
<S> → a<S>a | b<S>b | c

Can you generalize:

which strings can be formed using this grammar?

Parsing a string

- Parsing is the process of analyzing a string conforming to the rules of a grammar



Python's Grammar

- Python's full grammar is defined here
  - You are not required to understand what's written there, though

- Python code is parsed according to this grammar

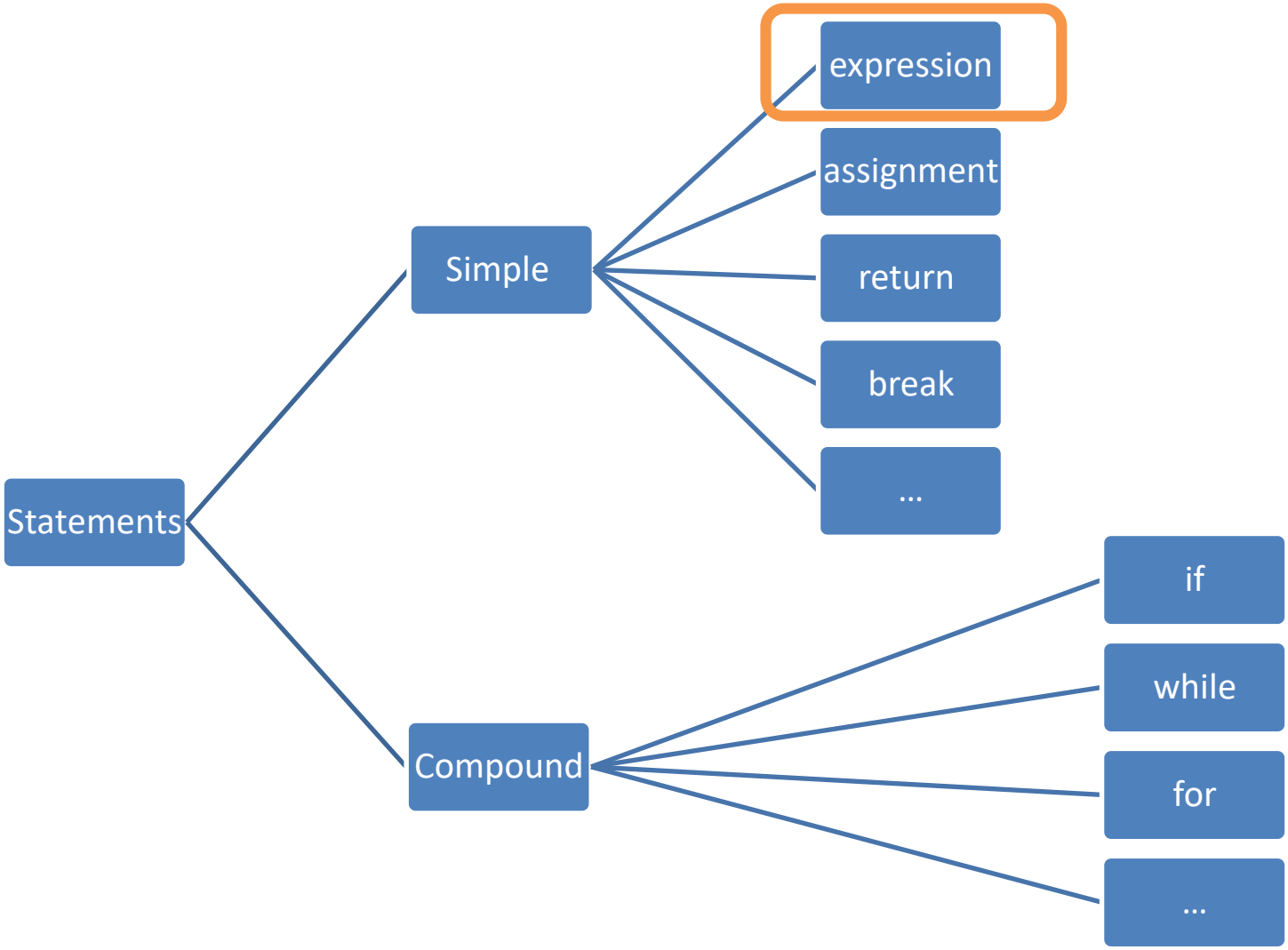
Statements in Python

- Can be either simple or compound

Simple statements	Compound statements

• expression statements, e.g.,
`3+4**2`
• assignment statements, e.g.,
`res = 400`
• return statement
`return res`
• break statement
`break`
 	
and many more examples
	• if statement
	`if a > b:`
	`. . .`
	• while statement
	`while a > b:`
	`. . .`
	• for statement
	`for a in lst:`
	`. . .`
	
	and many more examples

Statements in Python: Sketch



Expressions

- An expression statement is a statement which "has a value". That is, anything that can be the right hand side of an assignment (e.g., `res=…`). Examples:

```
>>> 3+2
5
>>> x
10 #Suppose that x=10
>>> x>y
False
>>> x>y and "A" in "Amir"
False
>>> min([1,2,3,4,5])
1
>>> [x**2 for x in [1,2,3] if x-1 != 0] #list comprehension
[4,9]
>>> "equal" if x==4 else "not equal"    #conditional expression
'equal'
>>> print
<built-in function print>
>>> None
>>> lambda x,y: x+y
<function <lambda> at 0x00000201B973C268>
```

More about lambda expressions soon