

# Example Solutions for Programming Languages G22.2110 Summer 2007 hw01

Assigned Th 5/17/2007, due We 5/23/2007 at 1pm.

These are example solutions. Please keep in mind that often, there is not just one correct solution to a question. If you come up with different answers to the homework, then it may be that both your answers and these answers here are correct. Of course, these answers here may also contain mistakes; if you spot some, please let us know so we can correct them.

---

## Reading Assignments

- For lecture on 5/17/2007: Scott 1.2, 1.3, 2.1
  - For lecture on 5/24/2007: Scott 6.0, 6.1.0-6.1.1;  
van Rossum 1-5 (<http://docs.python.org/tut/tut.html>)
- 

## Homework Assignments

1. Syntax (18 = 4 + 4 + 5 + 5 points)  
Consider the following regular expression:  
 $(XY)^*XY^*$

- 1a. (4 points) Give two examples of words matched by the regular expressions.

<p><i>Example solutions</i> Positive example 1: XYXYXY Positive example 2: XYYYY</p>
--

- 1b. (4 points) Give two examples of words for which the regular expression does not match.

<p><i>Example solutions</i> Negative example 1: Y Negative example 2: XYXX</p>
--

- 1c. (5 points) Give a context-free grammar in BNF that matches the same language as the regular expression. Do not use Kleene star or meta-level parentheses in your grammar.

Example solutions

start symbol: <S>

<S> -> <XYSTAR> X <YSTAR>

<XYSTAR> ->  $\epsilon$  | XY <XYSTAR>

<YSTAR> ->  $\epsilon$  | Y <YSTAR>

- 1d. (5 points) Give the rightmost derivation for the word XYXY using your grammar from Question 1c.

Example solutions

<S> => <XYSTAR> X <YSTAR> // rule <S> -> <XYSTAR> X <YSTAR>

=> <XYSTAR> XY <YSTAR> // rule <YSTAR> -> Y <YSTAR>

=> <XYSTAR> XY // rule <YSTAR> ->  $\epsilon$

=> XY <XYSTAR> XY // rule <XYSTAR> -> XY <XYSTAR>

=> XYXY // rule <XYSTAR> ->  $\epsilon$

2. More syntax (20 = 5 + 5 + 5 + 5 points)

Consider the following context-free grammar:

start symbol <s>

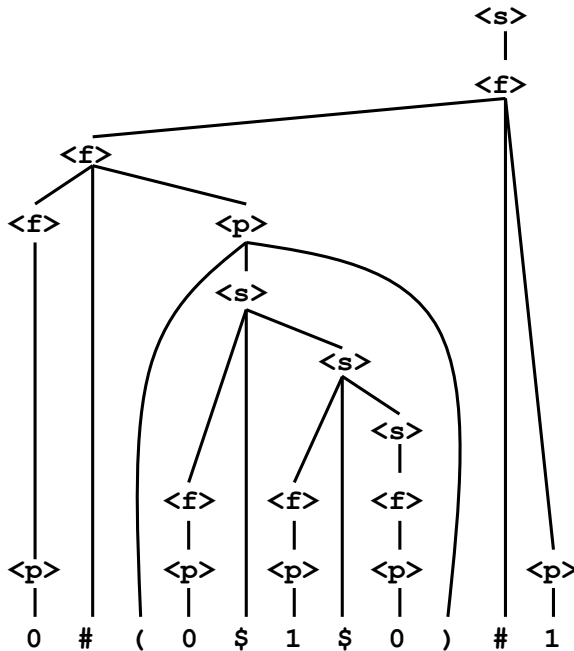
<s> ::= <f> | <f> '\$' <s>

<f> ::= <p> | <f> '#' <p>

<p> ::= '(' <s> ')' | '0' | '1'

- 2a. (5 points) Give the parse tree for the word 0#(0\$1\$0)#1.

Example solutions



- 2b. (5 points) Assume the symbols '\$' and '#' are operators of a programming language. What is their precedence, as given by the grammar?

Example solutions

The precedence of # is higher than the precedence of \$. In other words, # binds stronger than \$. In the absence of parentheses, operands of # are grouped together by nonterminal <f> deeper in the tree, whereas operands of \$ are grouped together by nonterminal <s> higher in the tree.

- 2c. (5 points) What is the associativity of operators \$ and # in the grammar?

Example solutions

- Operator \$ is right-associative. As can be seen by the subtree for '0\$1\$0' in the parse tree from 2a, the operands of \$ group to the right.
- Operator # is left-associative. As can be seen by the top portion of the parse tree from 2b, the operands of # group to the left.

- 2d. (5 points) Modify the grammar by adding a unary prefix operator @ that has a higher precedence than both \$ and #.

Example solutions

The modification adds one more alternative to the nonterminal <p>:  
<p> ::= '(' <s> ')' | '0' | '1' | '@' <p>

The rest of the grammar remains unchanged. The operator @ is unary, because it has only one operand <p>. It is a prefix operator, because it comes before its operand. It has higher precedence than both \$ and #, because it is an alternative of nonterminal <p>, which is treated as atomic by the nonterminals for \$ and #.

3. Python (0 points)

Start teaching yourself Python by doing the following:

- 3a. If possible, find peers (other students who want to learn Python together with you) and gurus (people who already know Python, whom you can ask questions when you get stuck).
- 3b. Make sure you have access to a Python interpreter in your preferred working environment. If you are lucky, it may already be installed. Otherwise, you can get it from the Python webpage, or through one of the common GNU package managers, such as Cygwin for Windows or Fink for Mac OS.
- 3c. Read the first 5 sections of the Python tutorial (see reading assignment above). Along the way, try things out with the Python interpreter that you installed in Step b.
- 3d. Familiarize yourself with the structure of the online Python documentation, so you can find information quickly when you need it. In particular, find the URLs for the Python library reference and the Python language reference.

4. More Python (12 = 3 + 3 + 3 + 3 points)

Continue teaching yourself Python by reading the example code from

<http://www.cs.nyu.edu/courses/summer07/G22.2110-001/hw01-py-example.txt>

4a. (3 points) What does the script print for the example data at its top?

Example solutions

operator	precedence	arity	associativity	description
or	2	binary	left	Boolean OR
and	3	binary	left	Boolean AND
not x	4	unary	n/a	Boolean NOT
in, not in	5	binary	left	Membership tests
is, is not	6	binary	left	Identity tests
<, <=, >, >=, <>, !=, ==	7	binary	chained	Comparisons

4b. (3 points) Add the line `print str(table)` to the bottom of the script. Based on the output, describe briefly what function `read_table` does.

Example solutions

Function `readtable` reads the file in comma-separated-values format, and turns it into a list of lists, where each inner list corresponds to one line. Here is a part of the output:

```
[['operator', 'precedence', 'arity', 'associativity', 'description'],  
 ['or', '2', 'binary', 'left', 'Boolean OR'],  
 ['and', '3', 'binary', 'left', 'Boolean AND'],  
 ... ]
```

4c. (3 points) Try the regular expression stand-alone in the interactive Python interpreter:

```
>>> import re  
>>> re.findall(r'^[,]*|"[^"]+"', 'a,"b,c",d')
```

Based on the output, describe briefly what the regular expression does.

Example solutions

Match either substrings that involve no commas and double-quotes, or substrings that are enclosed by double-quotes and may contain commas.

4d. (3 points) Try the string interpolation stand-alone in the Python interpreter:

```
>>> '%-*s' % (3, 'a')  
>>> '%-*s' % (4, 'aa')
```

Based on the output, describe briefly what the format string does.

Example solutions

Treat the second element of the tuple as a string, pad it on the right to the width specified by the first element of the tuple.

---

<http://www.cs.nyu.edu/courses/summer07/G22.2110-001/hw01-example-solutions.pdf>

Total points: 50.