
TDDD55, Exercises Part 2, Example Solutions

1.
a.
xz=2323, zyx=323132, z^2=323323, x^7 = 2222222

b.
A^1={1,2,3}, A^2={11,12,13,21,22,23,31,32,33}, A^0={epsilon}

c.
A^* = A^0 U A^1 U A^2 U ...
A^+ = A^1 U A^2 U A^3 U ...

2.

1. <exp> ::= <term>
2. | <exp> + <term>
3. | <exp> - <term>
4. <term> ::= <factor>
5. | <term> * <factor>
6. | <term> / <factor>
7. <factor> ::= (<exp>)
8 | <ident>
9. <ident> ::= A | B | C ... | Z

a.
Example derivations:
A*B-C
Starting with <exp>
<exp> **-> 3 ->** <exp> - <term>
<exp> - <term> **-> 4 ->** <exp> - <factor>
<exp> - <factor> **-> 8 ->** <exp> - <ident>
<exp> - <ident> **-> 9 ->** <exp> - C
<exp> - C **-> 1 ->** <term> - C
<term> - C **-> 5 ->** <term> * <factor> - C
<term> * <factor> - C **-> 8 ->** <term> * <ident> - C
<term> * <ident> - C **-> 9 ->** <term> * B - C
<term> * B - C **-> 4 ->** <factor> * B - C
<factor> * B - C **-> 8 ->** <ident> * B - C
<ident> * B - C **-> 9 ->** A * B - C

A*(B-C)
Starting with <exp>
<exp> **-> 1 ->** <term>
<term> **-> 5 ->** <term> * <factor>
<term> * <factor> **-> 7 ->** <term> * (<exp>)
<term> * (<exp>) **-> 3 ->** <term> * (<exp> - <term>)
<term> * (<exp> - <term>) **-> 4 ->** <term> * (<exp> - <factor>)
<term> * (<exp> - <factor>) **-> 8 ->** <term> * (<exp> - <ident>)
<term> * (<exp> - <factor>) **-> 9 ->** <term> * (<exp> - C)
<term> * (<exp> - C) **-> 1 ->** <term> * (<term> - C)
<term> * (<term> - C) **-> 4 ->** <term> * (<factor> - C)
<term> * (<factor> - C) **-> 8 ->** <term> * (<ident> - C)
<term> * (<factor> - C) **-> 9 ->** <term> * (B - C)

<term> * (B - C) **-> 4 ->** <factor> * (B - C)
<factor> * (B - C) **-> 8 ->** <ident> * (B - C)
<ident> * (B - C) **-> 9 ->** A * (B - C)

A/B/C
Starting with <exp>
<exp> **-> 1 ->** <term>
<term> **-> 6 ->** <term> / <factor>
<term> / <factor> **-> 8 ->** <term> / <ident>
<term> / <ident>**-> 9 ->** <term> / C
<term> / C **-> 6 ->** <term> / <factor> / C
<term> / <factor> / C **-> 8 ->** <term> / <ident> / C
<term> / <ident> / C **-> 9 ->** <term> / B / C
<term> / B / C **-> 4 ->** <factor> / B / C
<factor> / B / C **-> 8 ->** <ident> / B / C
<ident> / B / C **-> 9 ->** A / B / C

-A*B can not be derived since we do not have unary minus in the grammar

Parse trees: (These were drawn during the lesson)

b. <term> ::= <term>*<factor>, Position 0

c.
If every derivation step is rightmost, then this is a canonical derivation.

d. It cannot be derived

e.
$V = N \cup \sum$
$\sum = \{A, B, \ldots, Z, (,), +, -, /, *\}$
N={<exp>,<term>,<factor>,<ident>}

f.

$\sum^{+}$

Strings of terminal symbols only, containing at least one terminal symbol. (is in this set but not in the language L(G).

L(G)
Language generated by grammar G

4.

<even> ::= <start> <middle> <end> | <evennr>
<oddnr> ::= 1 | 3 | 5 | 7
<evennr> ::= 2 | 4 | 6 | 8
<start> ::= <oddnr> | <evennr>
<middle> ::= <oddnr><middle>
| <evennr><middle>
| 0 <middle>
| epsilon
<end> ::= <evennr> | 0

5.

(0 + 1)^+ ((- **+** +) (0 + 1)^+)*

6.

a.

Exactly one vowel
<start> ::= <A> <vowel> <A>
<A> ::= <not_vowel> <A> | epsilon
<not_vowel> ::= b | c | d | f | g | h
<vowel> ::= a | e | i

b.

At least one vowel
<start> ::= <A> <vowel> <A>
<A> ::= <not_vowel> <A> | <vowel> <A> | epsilon
<not_vowel> ::= b | c | d | f | g | h
<vowel> ::= a | e | i

7.

a. a* (b+c)* a*

b. a^n b^n c^n , n=>0
No, regular expressions "can't count".
See for instance *Automata and Computability, Dexter C. Kozen, Springer Verlag.*

8.

a.

1^n 0^n 1^m 0^m | n>0, m>0

b.

1^n 0^m 1^m 0^n | n=>0, m=>0

c.

1^n 1^m 0^m | n>0, m=>0

OR

1^m 0^m 0^n | n>0, m=>0