# CmpE 260 - Principles of Programming Languages
# Spring 2019
# Assignment 1

İbrahim Özgürcan ÖZTAŞ — 2016400198

## Solution 1

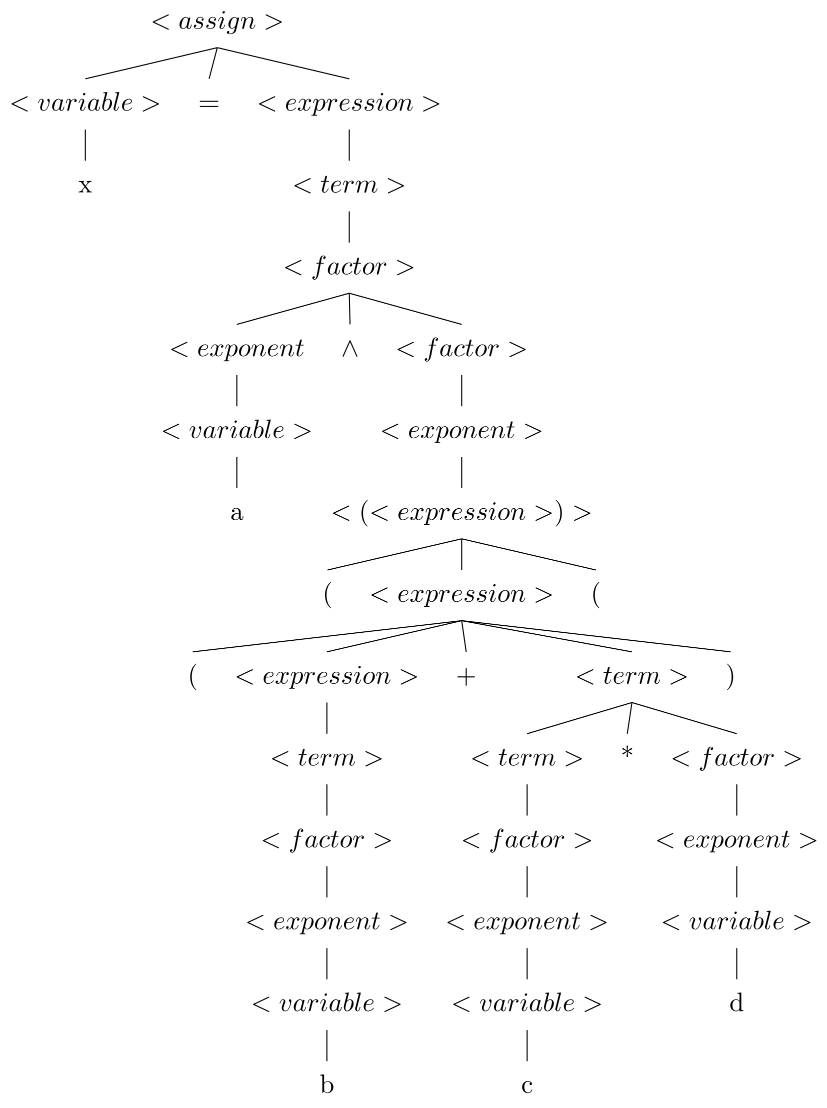
**a)**

$$\begin{aligned}
<assign> &\rightarrow <variable>=<expression> \\
<expression> &\rightarrow <expression> + <term> \mid <expression> - <term> \mid <term> \\
<term> &\rightarrow <term> * <factor> \mid <term> / <factor> \mid <factor> \\
<factor> &\rightarrow <variable> \mid (<expression>) \\
<variable> &\rightarrow a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z
\end{aligned}$$

**b)**

$$\begin{aligned}
<assign> &\rightarrow <variable>=<expression> \\
<expression> &\rightarrow <expression> + <term> \mid <expression> - <term> \mid <term> \\
<term> &\rightarrow <term> * <factor> \mid <term> / <factor> \mid <factor> \\
<factor> &\rightarrow <exponent> \wedge <factor> \mid <exponent> \\
<exponent> &\rightarrow <variable> \mid (<expression>) \\
<variable> &\rightarrow a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z
\end{aligned}$$

**c)**



$< assign >$
$< variable >$
$=$
$< expression >$
x
$< term >$
$< factor >$
$< exponent$
$\wedge$
$< factor >$
$< variable >$
$< exponent >$
a
$< (< expression >) >$
(
$< expression >$
(
(
$< expression >$
$+$
$< term >$
)
$< term >$
$< term >$
$*$
$< factor >$
$< factor >$
$< factor >$
$< exponent >$
$< exponent >$
$< exponent >$
$< variable >$
$< variable >$
$< variable >$
d
b
c

## Solution 2

```
$M_{repeat}$ (repeat<st-list> until <bool> ,s) $\Delta =$
    if $M_{statement\text{-}list}$(<st-list>,s) = error
        then error
    else
        if $M_{boolean}$(<bool>,$M_{statement\text{-}list}$(<st-list>,s)) = error
            then error
        else
            if $M_{boolean}$(<bool>,$M_{statement\text{-}list}$(<st-list>,s)) = true
                then $M_{statement\text{-}list}$(<st-list>,s)
            else $M_{repeat}$ (repeat<st-list> until <bool>, $M_{statement\text{-}list}$(<st-list>,s))
$M_{boolean}$(<var>$_1$==<var>$_2$,s) $\Delta =$
    if VarMap(<var>$_1$,s) =undef
        then error
    else
        if VarMap(<var>$_2$,s) = undef
            then error
        else
            if VarMap(<var>$_1$,s) = VarMap(<var>$_2$,s)
                then true
            else false
$M_{statement\text{-}list}$(<ass-st><st-list>,s) $\Delta =$
    if $M_{assign}$(<ass-st>,s) = error
        then error
    else $M_{statement\text{-}list}$(<st-list>,$M_{assign}$(<ass-st>,s))
$M_{statement\text{-}list}$(<ass-st>,s) $\Delta =$
     $M_{assign}$(<ass-st>,s)
$M_{assign}$(<var>$_1$ = <var>$_2$,s) $\Delta =$
    if VarMap(<var>$_2$,s) = undef
        then error
    else
         <$i_1$,$v_1$>,...,<$i_n$,$v_n$> where
                $v_j$=VarMap($i_j$,s) , if $i_j$!= <var>$_1$
                VarMap(<var>$_2$,s) , if $i_j$= <var>$_1$
```

## Solution 3

$< world > \rightarrow < katara-bender >< toph-bender >< zuko-bender >< aang-bender >$
$< katara-bender > \rightarrow katara < katara-elements >$
$< toph-bender > \rightarrow toph < toph-elements >$
$< zuko-bender > \rightarrow zuko < zuko-elements >$
$< aang-bender > \rightarrow aang < aang-elements >$
$< element > \rightarrow W|E|F|A$
$< katara-elements >_1 \rightarrow < element >< katara-elements >_2$
$< katara-elements >_1$.number $\leftarrow if < element > == W$
then $< katara-elements >_2$.number + 1
else $< katara-elements >_2$.number
$< katara-elements > \rightarrow < element >$
$< katara-elements >$.number $\leftarrow if < element > == W$
then 1
else 0
$< toph-elements >_1 \rightarrow < element >< toph-elements >_2$
$< toph-elements >_1$.number $\leftarrow if < element > == E$
then $< toph-elements >_2$.number + 1
else $< toph-elements >_2$.number
$< toph-elements > \rightarrow < element >$
$< toph-elements >$.number $\leftarrow if < element > == E$
then 1
else 0
$< zuko-elements >_1 \rightarrow < element >< zuko-elements >_2$
$< zuko-elements >_1$.number $\leftarrow if < element > == F$
then $< zuko-elements >_2$.number + 1
else $< zuko-elements >_2$.number
$< zuko-elements > \rightarrow < element >$
$< zuko-elements >$.number $\leftarrow if < element > == F$
then 1
else 0
$< aang-elements >_1 \rightarrow < element >< aang-elements >_2$
$< aang-elements >_1$.number $\leftarrow < aang-elements >_2 .number + 1$
$< aang-elements > \rightarrow < element >$
$< aang-elements >$.number $\leftarrow 1$
predicate : $< katara-elements >$.number == $< toph-elements >$.number &&
$< toph-elements >$.number == $< zuko-elements >$.number &&
$< zuko-elements >$.number == $< aang-elements >$.number