

# CS536 Homework 4

Due at 11 PM on Feb 25 2020

## Questions

Homework assignments must be done individually. Collaboration on homework assignments is *not* allowed.

For this homework you will define a syntax-directed translation for the CFG given below, which defines a very simple programming language.

```

program → MAIN LPAREN RPAREN LCURLY list RCURLY
list → list oneItem
      | epsilon

oneItem → decl
        | stmt

decl → BOOL ID SEMICOLON
     | INT ID SEMICOLON

stmt → ID ASSIGN exp SEMICOLON
     | IF LPAREN exp RPAREN stmt
     | LCURLY list RCURLY

exp → exp PLUS exp
    | exp LESS exp
    | exp EQUALS exp
    | ID
    | BOOLLITERAL
    | INTLITERAL

```

### Question 1:

Write a syntax-directed translation for the CFG given above to extract all the **boolean literals and integer literals**.

Your translation rules should use the following notation:

- $\{ \}$  is an empty set
- $\{ \text{BOOLLITERAL.value} \}$  is a set containing the value of the `BOOLLITERAL` token
- $\{ \text{ID.value} \}$  is a set containing the value of the `ID` token
- $s_1 \cap s_2$  is the intersection of sets  $s_1$  and  $s_2$
- $s_1 \cup s_2$  is the union of sets  $s_1$  and  $s_2$
- $s_1 - s_2$  is the set of all items that are in  $s_1$  but not in  $s_2$

Note that you should not try to use something like " $\{ a, b \}$ " to mean a set with two elements; instead, use set union to combine two sets that each contain one element.

Use the notation that was used in class and in the on-line readings; i.e., use `nonterminal.trans` to mean the translation of a nonterminal, and `terminal.value` to mean the value of a terminal. Assume that `ID.value` is a `String` (the name of the identifier). Use subscripts for translation rules that include the same nonterminal or the same terminal more than once.

### Question 2:

Draw a parse tree for the program given below and annotate each nonterminal in the tree with its translation.

```

main ( ) {
    int x;
    bool y;
    if (y == true) {
        y = false;
        x = x + 2;
    }
}

```