# CS131 Compilers: Writing Assignment 1
# Due 11:59pm April 2, 2023

## Name - ID

I worked with Name1 Name2 ...
Complteted on March 18, 2023

### Code of Conduct

**This writing assignment should be your own individual work. Discussion on concept, methodology, and class materials are welcomed, but you should list all the people you have discussed with. Copying is strictly prohibited. Plagiarism, once confirmed, may result in assignment grades reduced to zero for all involved people. And this event will be reported. Also you should use LaTeX to produce your response based on this template. Submission in other forms won't be graded.**

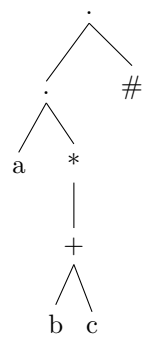



Figure 1: Simple Tree drawn by Tikz-qtree package

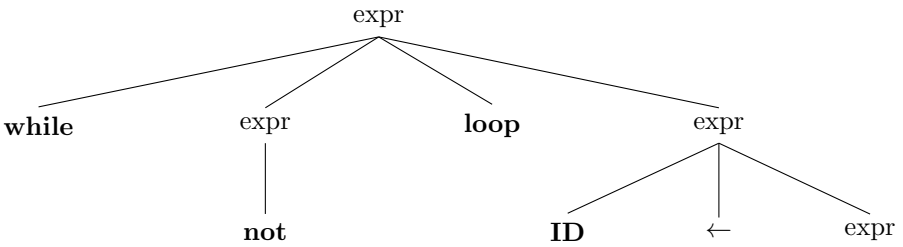



Figure 2: Complex tree Drawn by Tikz package

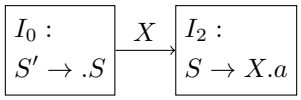



Figure 3: Item automata drawn by Tikz package

| STATE | ACTION | | | | | GOTO | | | |
|---|---|---|---|---|---|---|---|---|---|
| | ( | ) | q | a | $ | E | T | Q | A |
| 0 | s2 | | | | | 1 | | | |
| 1 | | | | | acc | | | | |
| 2 | | | s6 | | | | 3 | 4 | |
| 3 | | | | | r1 | | | | |
| 4 | | s5 | | | | | | | |
| 5 | | | | | r2 | | | | |
| 6 | | | r5 | s9 | | | | | 7 |
| 7 | | | s8 | | | | | | |
| 8 | | r3 | | | | | | | |
| 9 | | | r5 | s9 | | | | | 10 |
| 10 | | | r4 | | | | | | |

Figure 4: A parser table (generated by LatexTable Generator)

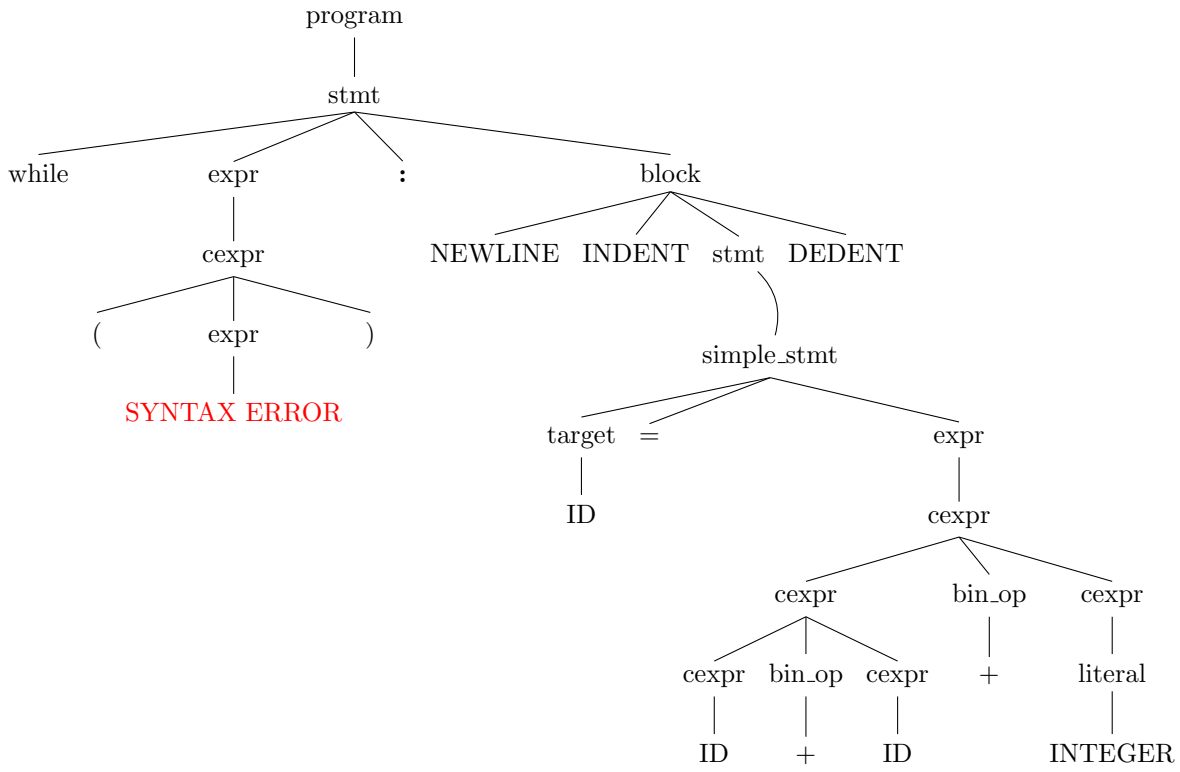



Figure 5: A very large tree drawn by Tikz package

$$
\begin{aligned}
S_{\epsilon} &\rightarrow \epsilon \\
S_{c \in \Sigma} &\rightarrow c_1 | c_2 | \ldots | c_{|\Sigma|} \\
S_{AB} &\rightarrow S_A S_B \\
S_{A+B} &\rightarrow S_A | S_B \\
S_{A^*} &\rightarrow S_A S_{A^*} | \epsilon
\end{aligned}
$$

Figure 6: a context Free grammar that can represent a regular expression

1. ($2+2+2+2+2+4+4=18$ **pts**) For each of the following question, write a **short and clear** answer. (Hint: All answers can be found according to slides and textbook.)

   (a) Did you start your PAs and find one teammate? Who is he/she/they?
   **RESPONSE:**

   (b) Context Free Grammar(CFG) can be described by $(\mathcal{T}, \mathcal{NT}, \mathcal{S}, \mathcal{P})$. What are they?
   **RESPONSE:**

   (c) What do "**Leftmost**" and "**Rightmost**" mean in **derivation**?
   **RESPONSE:**

   (d) What is the **Ambiguity** of a grammar? Then name two methodology to solve **Ambiguity**.
   **RESPONSE:**

   (e) What is **Predictive Parser** and **Recursive Descent Parser**?
   **RESPONSE:**

   (f) $\forall x \in \{L, R, S, A, k\}$, what does $x$ mean in the following expressions?

      i. LL(k)
      **RESPONSE:**

      ii. LR(k)
      **RESPONSE:**

      iii. SLR
      **RESPONSE:**

      iv. LALR
      **RESPONSE:**

   (g) If you are given a **conflict-free LR(1)** automaton, how can you get a **LALR(1)** parse table from it? (Give a short and clear answer)
   **RESPONSE:**

2. ($2+2+2+4=10$ **pts**) Write a relevant CFG or draw a parser(ie. pushdown automata, but not recommended) that correctly represents the context free language described in each question. Your response should be sound and complete.

   (a) $L_1 = \{(^k)^k | k = 0, 1, 2, 3, \dots\}$
   **RESPONSE:**

   (b) $L_2 = \{\text{All non-empty Palindrome strings over } \Sigma = \{0, 1\}\}$
   **RESPONSE:**

   (c) $L_3 = L_1 + L_2 = L_1 \cup L_2$
   **RESPONSE:**

   (d) $L = \{\mathbf{a}^n\mathbf{b}^n\mathbf{c}^m\mathbf{d} | m, n \geq 0\} \cup \{\mathbf{a}^n\mathbf{b}^m\mathbf{c}^m\mathbf{e} | m, n \geq 0\}$
   (Note: this language is unambiguous but has not a unambiguous grammar.)
   **RESPONSE:**

3. $(3+3+3+6+6+9=30$ **pts**) Consider this grammar $G_1$ (simplified from ChocoPy):

$$\begin{aligned} expr &\rightarrow binary|\textbf{id}|expr\ \textbf{if}\ expr\ \textbf{else}\ expr \\ binary &\rightarrow expr\ op\ expr \\ op &\rightarrow \textbf{and}|\textbf{or} \end{aligned}$$

Where *expr* is the start symbol.

(a) How many different **leftmost derivation** parsing trees are there of

**id and id or id if id else id**

**RESPONSE:**

(b) Do Left-recursion elimination on the grammar $G_1$, show me the converted grammar $G_2$.
**RESPONSE:**

(c) Do Left-factoring on the grammar $G_2$, show me the converted grammar $G_3$.
**RESPONSE:**

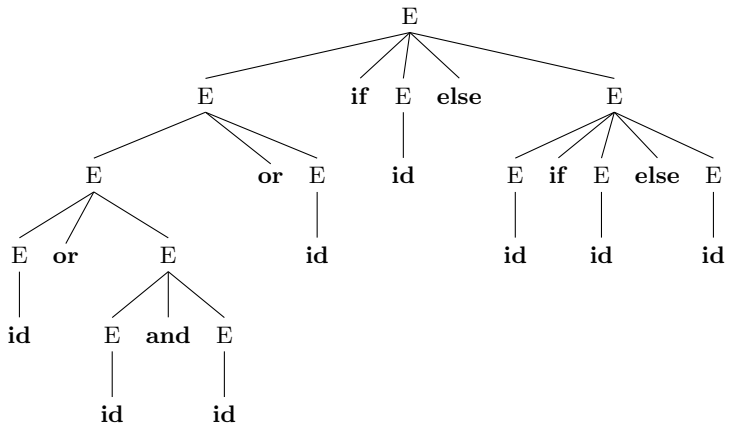
(d) Consider **LL(1)**. Compute the **FIRST** and the **FOLLOW** function of the grammar $G_3$.
**RESPONSE:**

(e) Construct the **LL(1)** parse table of $G_3$. Then tell whether $G_3$ is **LL(1)** grammar.
**RESPONSE:**

(f) Suppose the precedence is **and** > **or** > **if-else**. Also, suppose **and**, **or** are left-associative, and **if-else** is right-associative. Design a new context free grammar $G_1'$ that remove all ambiguity and $L(G_1) = L(G_1')$.
Left-associative means
($L(G)$ means the set of all strings that the grammar $G$ can represent. )
The correct grammar would parse the following input:

**id or id and id or id if id else id and id if id else id**

to some derivation trees like the below. **This tree is not the final parse tree.** It only illustrates the tree structure of operators by the structures of terminals(ie. leaves), which indicates the order of calculating(father computed after decedents, and this is how operator precedence is solved in real world).




**RESPONSE:**

4. $(3+4+4+6+4=21$ **pts**) Consider the following grammar $G_4$, simplified from the ChocoPy.

$$
\begin{aligned}
\textit{stmt} &\rightarrow \textbf{if } \textit{expr} \textbf{ : } \textit{block} \textbf{ else : } \textit{block} \\
\textit{stmt} &\rightarrow \textbf{if } \textit{expr} \textbf{ : } \textit{block} \\
\textit{block} &\rightarrow \textbf{NEWLINE INDENT ID DEDENT} \\
\textit{expr} &\rightarrow \textbf{ID}
\end{aligned}
$$

where *stmt* is the start symbol. You can use abbreviation **NL**, **IND**, **DED** for long terminals.

(a) Suppose we use LR(0) technique, identify **one shift-reduce conflicts** by constructing the items automata. You don't need to draw the whole automata. **Instead, draw the sets of items(ie. the states in the automaton) where the conflicts occur.**
**RESPONSE:**

(b) To construct at SLR(1) parser, Compute FIRST function and FOLLOW function of each Non-terminals.
**RESPONSE:**

(c) Construct the SLR items automaton. The process is not required.
**RESPONSE:**

(d) Based on the SLR items automaton, construct the parsing table. The process is not required. Please refer to the format in the lectures or in the preface 4.
**RESPONSE:**

(e) Based on your parse table, parse this input sequence to a parse tree.

**if ID : NEWLINE INDENT ID DEDENT**

**RESPONSE:**

5. $(15 + 4 + 4 = 23$ **pts**)Consider the CFG below

$$S \to X\ X$$
$$X \to \mathbf{a}\ X$$
$$X \to \mathbf{b}$$

(a) Construct the LALR(1) parse table. **The process is not required. But showing your process may save your score if your answer is accidentally wrong.**

i. Build LR(1) items automata (optional)

ii. Build the LALR(1) parse table. (required)

**RESPONSE:**

(b) Use your parse table, parse the following input. Your response should include the all configurations from the initial state to the ending state.

**a a a b b**

**RESPONSE:**

(c) Build the parse tree of the input above.
**RESPONSE:**

---

## End of This Assignment