CS323 Assignment 5

1 Requirements

You are expected to complete all required homework exercises. For submission, please put all your answers in a single PDF file and submit it via the assignment channel on SAKAI. The name of the file should follow the format **"studentID_A#"** (e.g., 30003554_A5). **The submission deadline is 11:55 PM, December 20, 2022.** Late submissions are allowed within one week after the deadline (grace period). If you submit your assignment during the grace period, your score will be 80% of the score you could get if the submission was made in time. Assignment submitted after the grace period will not be graded, meaning that you will get a zero for the assignment.

2 Required Exercises (100 points)

Exercise 1: For the SDD in Figure 1, give annotated parse tree for the following expression (the symbol '**n**' is the end marker): $(1 + 2 * (3 + 4) + 5) * 6\mathbf{n}$. [20 points]

	PRODUCTION	SEMANTIC RULES
1)	$L \to E\ \mathbf{n}$	$L.val = E.val$
2)	$E \to E_1\ +\ T$	$E.val = E_1.val + T.val$
3)	$E \to T$	$E.val = T.val$
4)	$T \to T_1\ *\ F$	$T.val = T_1.val \times F.val$
5)	$T \to F$	$T.val = F.val$
6)	$F \to (\ E\ )$	$F.val = E.val$
7)	$F \to \mathbf{digit}$	$F.val = \mathbf{digit}.\text{lexval}$

Figure 1: Syntax-directed definition of a simple desk calculator

Exercise 2: What are all the topological sorts for the dependency graph of Figure 2? One sort mentioned during lecture is $1, 2, 3, \ldots, 9$ (slide #16 of Chapter 4). [20 points]

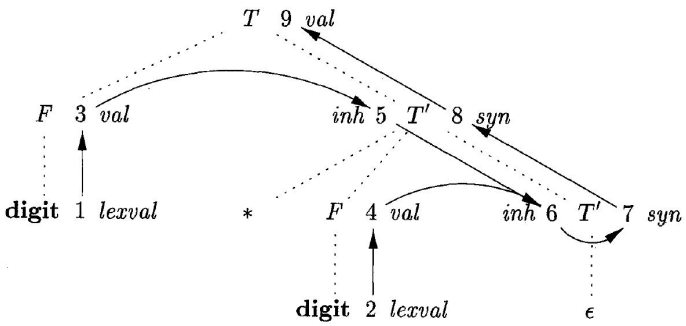

Figure 2: A dependency graph

Exercise 3: Below is an SDD introduced during our lecture for computing the structure of a type.

- Is the SDD S-attributed? Why? [10 points]
- Is the SDD L-attributed? Why? [10 points]
- Given the input `float[3][4][5]`, please give an annotated parse tree and the evaluation order of the attributes (you may refer to the slide #39 of Chapter 4). [20 points]

PRODUCTION	SEMANTIC RULES
$T \rightarrow B\ C$	$T.t = C.t$
	$C.b = B.t$
$B \rightarrow$ **int**	$B.t = integer$
$B \rightarrow$ **float**	$B.t = float$
$C \rightarrow$ [**num**] C_1	$C.t = array(\mathbf{num}.val,\ C_1.t)$
	$C_1.b = C.b$
$C \rightarrow \epsilon$	$C.t = C.b$

Figure 3: An SDD for computing the structure of a type

Exercise 4: Below is a grammar for expressions involving operator + and integer or floating-point operands. Floating-point numbers are distinguished by having a decimal point. **digit** is a terminal representing a number in $[0, 9]$.

$$E \rightarrow E + T \mid T$$
$$T \rightarrow D \cdot D \mid D$$
$$D \rightarrow \mathbf{digit}$$

1. Give an L-attributed SDD to compute the value of the expression E. [15 points]
2. Is possible to evaluate all the attributes during bottom-up parsing process? [5 points]