Benjamin Smith
Email: bxs566@case.edu
Course: CSDS 337 - Compiler Design
Instructor: Dr. Vipin Chaudhary

Problem Set - 4
ID: 3559750
Term: Spring 2024
Due Date: 3^{rd} April, 2024

Number of hours delay for this Problem Set: 0
Cumulative number of hours delay so far: 40

I discussed this homework with: Jackson Schuetzle

Problem 1 - 15 points
Suppose that we have a production $A \to BCD$. Each of the four nonterminals A , B, C, and D have two attributes: s is a synthesized attribute, and i is an inherited attribute. For each of the sets of rules below, tell whether (i) the rules are consistent with an S-attributed definition (ii) the rules are consistent with an L-attributed definition, and (iii) whether the rules are consistent with any evaluation order at all?

a $A.s = B.i + C.s$.

b $A.s = B.i + C.s$ and $D.i = A.i + B.s$.

c $A.s = B.s + D.s$.

d $A.s = D.i, \quad B.i = A.s + C.s, \quad C.i = B.s$, and $D.i = B.i + C.i$.

Solution:

a The rule is S-attributed. The rule is also L-attributed. The rule can be evaluated with a depth-first order.

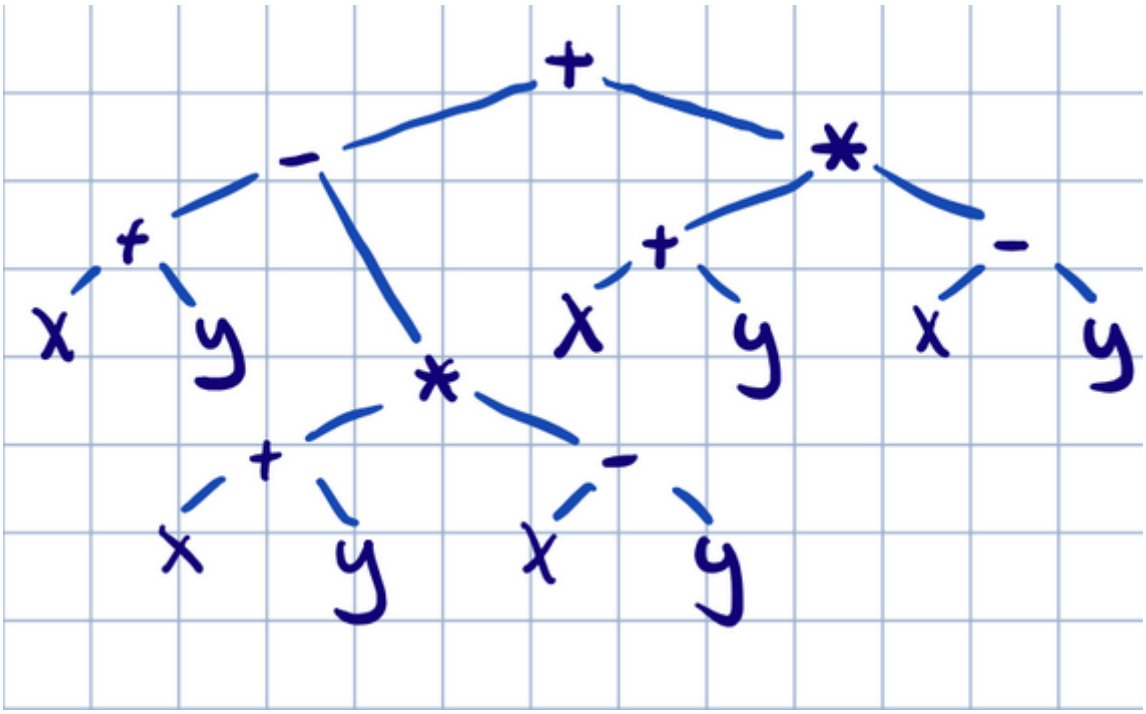
b The rules are not S-attributed because $D.i$ is not in the head of the production and thus $D.i = A.i+B.s$ is an inherited attribute. The rules are L-attributed because there are no cyclic dependencies in the inherited attributes of A. The rules are consistent with a depth-first evaluation order.

c The rule is S-attributed and L-attributed. The rule is consistent with a depth-first evaluation order.

d The rules are not S-attributed. The rules are not L-attributed either because the rules $B.i = A.s+C.s$ and $C.i = B.s$ form a cyclic dependency. The rules are not consistent with any evaluation order because no acyclic dependency graph can be created for some inputs.

Problem 2 - 15 points
Construct the DAG for the expression $((x + y) - ((x + y) * (x - y))) + ((x + y) * (x - y))$

Solution:



Problem 3 - 15 points
Translate the arithmetic expression $a + (b + c)$.

a A syntax tree.

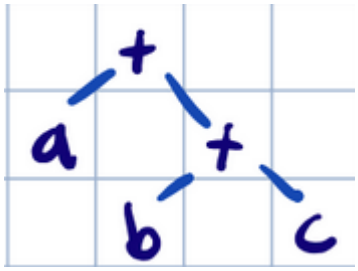
b Quadruples.

c Triples.

d Indirect triples.

Solution:

a



b

#	Op	Arg1	Arg2	Res
0	+	b	c	t1
1	+	a	t1	t2

c

#	Op	Arg1	Arg2
0	+	b	c
1	+	a	(0)

d

#	Statement
0	(10)
1	(11)

#	Op	Arg1	Arg2
10	+	b	c
11	+	a	(10)

Problem 4 - 20 points
A real array $A[i; j; k]$ has index i ranging from 1 to 4, j ranging from 0 to 4, and k ranging from 5 to 10. Reals take 8 bytes each. If A is stored row-major, starting at byte 0, find the location of:

a $A[3; 4; 5]$

b $A[1; 2; 7]$

c $A[4; 3; 9]$.

Repeat the above if A is stored in column-major order.

Solution:
Row-major:

a A row takes 5 * 4 * 8 = 160 bytes of space. The $i = 3$ row starts at 320. Add 5 * 3 * 8 = 120 to get to the $j = 4$ column in the row. Add 0 * 8 = 0 to get to the $k = 5$ index in the column. The location of $A[3; 4; 5]$ is at 440 bytes.

b 40 + 16 = 56 bytes.

c 480 + 80 + 72 = 632 bytes.

Column-major:

a

b

c

Problem 5 - 20 points
Add rules to the syntax-directed definition of Fig. 1 for the following control-flow constructs:

PRODUCTION	SEMANTIC RULES
$P \rightarrow S$	$S.next = newlabel()$ $P.code = S.code \mid\mid label(S.next)$
$S \rightarrow$ **assign**	$S.code =$ **assign**$.code$
$S \rightarrow$ **if** (B) S_1	$B.true = newlabel()$ $B.false = S_1.next = S.next$ $S.code = B.code \mid\mid label(B.true) \mid\mid S_1.code$
$S \rightarrow$ **if** (B) S_1 **else** S_2	$B.true = newlabel()$ $B.false = newlabel()$ $S_1.next = S_2.next = S.next$ $S.code = B.code$ $\mid\mid label(B.true) \mid\mid S_1.code$ $\mid\mid gen('\texttt{goto}'\ S.next)$ $\mid\mid label(B.false) \mid\mid S_2.code$
$S \rightarrow$ **while** (B) S_1	$begin = newlabel()$ $B.true = newlabel()$ $B.false = S.next$ $S_1.next = begin$ $S.code = label(begin) \mid\mid B.code$ $\mid\mid label(B.true) \mid\mid S_1.code$ $\mid\mid gen('\texttt{goto}'\ begin)$
$S \rightarrow S_1\ S_2$	$S_1.next = newlabel()$ $S_2.next = S.next$ $S.code = S_1.code \mid\mid label(S_1.next) \mid\mid S_2.code$

Figure 1: Rules to the syntax-directed definition

- A repeat-statement **repeat** S **while** B.
- A for-loop **for** $(S_1; B; S_2)S_3$.

Solution: $S \rightarrow$ **for** $(S_1; B; S_2)S_3$
$begin = newlabel()$
$B.true = newlabel()$
$B.false = S.next$
$S_3.next = begin$
$S.code = S_1.code \mid\mid label(begin) \mid\mid B.code \mid\mid label(B.true) \mid\mid S_3.code \mid\mid S_2.code \mid\mid gen('goto'\ begin)$

Problem 6 - 15 points
Translate the following expressions using the ifFalse mechanism:

a if $(a == b \quad \&\& \quad c == d \quad || \quad e == f) \quad x = 1;$

b if $(a == b \quad || \quad c == d \quad || \quad e == f) \quad x = 1;$

c if $(a == b \quad \&\& \quad c == d \quad \&\& \quad e == f) \quad x = 1;$

Solution:

a ifFalse $(a! = b \quad || \quad c! = d \quad \&\& \quad e! = f) \quad x = 1;$

b ifFalse $(a! = b \quad \&\& \quad c! = d \quad \&\& \quad e! = f) \quad x = 1;$

c ifFalse $(a! = b \quad || \quad c! = d \quad || \quad e! = f) \quad x = 1;$