This work is to be done before Exam 1.

Assigned Reading on the Syntax of Expressions

Read Sections 2.1 and 2.2 on pp. 28 – 33 of Sethi (up to and including the figure at the top of p. 33).

Homework Exercises on the Syntax of Expressions [Not for credit]

A. Do problems 2.1, 2.2, 2.3, 2.7, and 2.8 on pp. 49 – 50 of Sethi.

B. Do the following exercises on infix, prefix, and postfix syntax, and abstract syntax trees:

1. In a certain language expressions are written in infix syntax. The language has binary, prefix, and postfix operators that belong to the following precedence classes:

	binary ops	**prefix ops**	**postfix ops**	**associativity**
1st Class:	# ~	~	[none]	right
2nd Class:	@	[none]	$	left
3rd Class:	% ^	@	[none]	right

1st class operators have ***highest*** precedence and 3rd class operators have ***lowest*** precedence.

(a) Say which operator is applied last in the following expression, and then draw the abstract syntax tree of the expression. [To help you, subscripts have been attached to each operator to indicate its precedence class and whether that class is left- or right-associative, even though this information can also be obtained from the above table.]

$$(@_{3R}\ a\ \#_{1R}\ u)\ @_{2L}\ w\ \$_{2L}\ \%_{3R}\ (5\ \hat{}_{3R}\ b\ \sim_{1R}\ c)\ \hat{}_{3R}\ d$$

(b) Rewrite the expression in prefix syntax.

(c) Rewrite the expression in postfix syntax.

2. Draw the AST of the following postfix syntax expression, and rewrite the expression in prefix syntax. A subscript has been attached to each operator that shows the operator's arity. [The operators in this question and the next are ***not*** related to the operators in question 1.]

$$a\ b\ \$_2\ @_1\ 3\ u\ *_1\ v\ w\ \$_2\ \sim_2\ 5\ @_1\ \hat{}_3\ \#_2$$

3. Draw the AST of the following prefix syntax expression, and rewrite the expression in postfix syntax. A subscript has been attached to each operator that shows the operator's arity.

$$\hat{}_3\ x\ y\ \#_2\ \hat{}_3\ \sim_2\ u\ v\ *_1\ w\ \#_2\ -_1\ 5\ a\ b$$

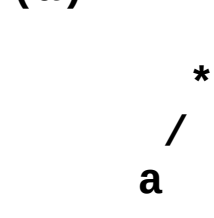
Solutions

Section A

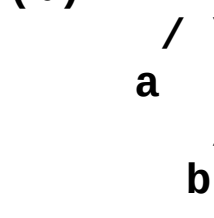
2.1 (a) + * a b c (b) * a + b c (c) + * a b * c d
(d) * * a + b c d (e) / + / b 2 SQRT - * / b 2 / b 2 * a c a

2.2 (a) a b * c + (b) a b c + * (c) a b * c d * +
(d) a b c + * d * (e) b 2 / b 2 / b 2 / * a c * - SQRT + a /

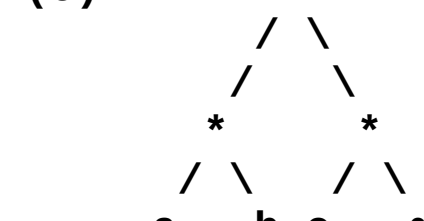
2.3 (a)



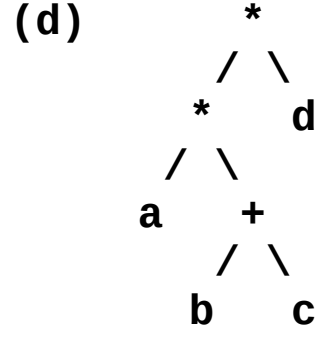
(b)



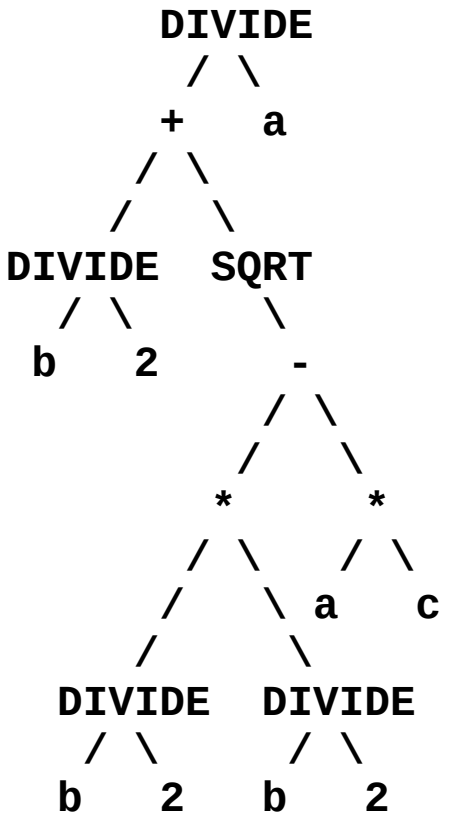
(c)



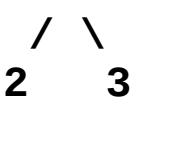
(d)



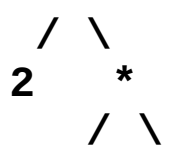
(e)



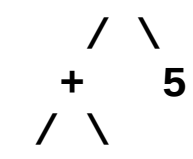
2.7 (a,b)



(c,e)



(d)



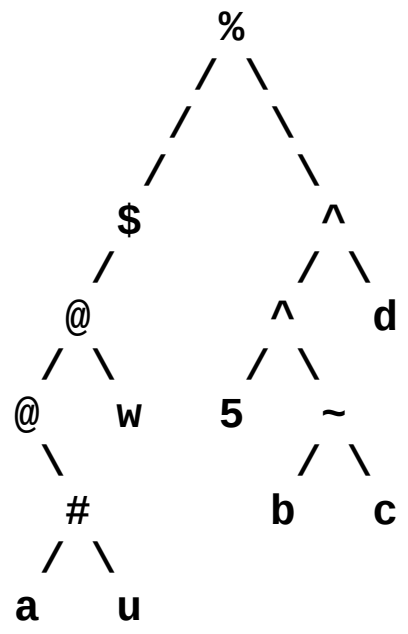
2.8

stack	remaining input
	7 7 * 4 2 * 3 * -
7	7 * 4 2 * 3 * -
7 7	* 4 2 * 3 * -
49	4 2 * 3 * -
49 4	2 * 3 * -

```
--------------------------
49 4 2                    * 3 * -
--------------------------

--------------------------
49 8                        3 * -
--------------------------

--------------------------
49 8 3                        * -
--------------------------

--------------------------
49 24                           -
--------------------------

--------------------------
25
--------------------------
```

Section B

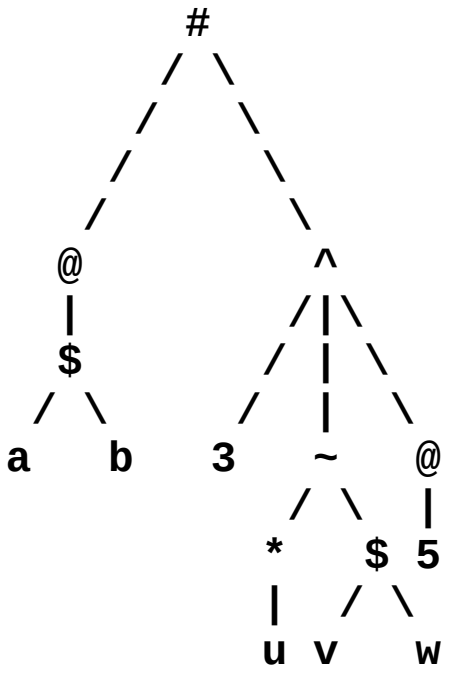
1.(a) % is applied last.



1.(b) % $ @ @ # a u w ^ ^ 5 ~ b c d

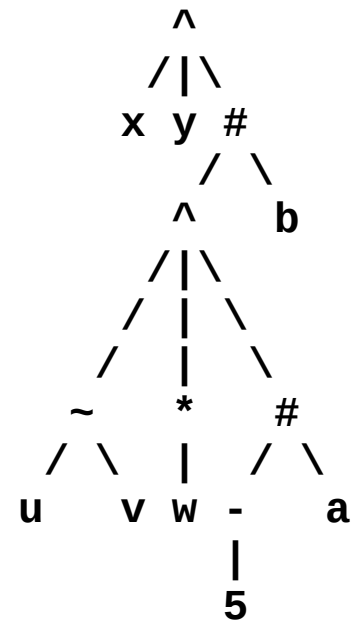
1.(c) a u # @ w @ $ 5 b c ~ ^ d ^ %

2.



prefix syntax: `# @ $ a b ^ 3 ~ * u $ v w @ 5`

3.



postfix syntax: x y u v ~ w * 5 - a # ^ b # ^