© 2023 Sudarshan S. Chawathe

Name:		

1. (1 pt.)

- Read all material carefully.
- If in doubt whether something is allowed, ask, don't assume.
- You may refer to your books, papers, and notes during this test.
- E-books may be used *subject to the restrictions* noted in class.
- Computers are not permitted, except when used strictly as e-books.
- Network access of any kind (cell, voice, text, data, ...) is not permitted.
- Write, and draw, carefully. Ambiguous or cryptic answers receive zero credit.
- Use class and textbook conventions for notation, algorithmic options, etc.

Write your name in the space provided above.

WAIT UNTIL INSTRUCTED TO CONTINUE TO REMAINING QUESTIONS.

Do not write on this page below this point.

Q	Full	Score
1	1	
2	9	
3	10	
4	10	
5	10	
total	40	

2.	(9 pts.)	Answer	the	following	briefly,	in	\mathbf{the}	${\bf context}$	of	\mathbf{the}	PLY	system	as
	discusse	d in cla	SS.										

(a) What is the main difference between literal and non-literal tokens?

(b) Provide a code snippet that defines the literal tokens + and *.

(c) Provide a code snippet that defines the non-literal tokens node and edge.

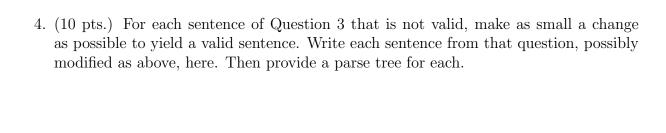
3. (10 pts.) Consider the following context-free grammar.

$$\begin{array}{ccc} S & \rightarrow & S F S \mid \mathtt{i} \mid \mathtt{n} \\ F & \rightarrow & + \mid - \mid * \mid / \end{array}$$

(a) For each symbol used above $(S, F, \rightarrow, |, i, n, +, -, *, /)$, indicate whether it belongs to the *language* (defined by the grammar) or the *metalanguage* or the *meta-metalanguage*. Provide *brief* explanations **iff** you wish to qualify for any partial credit.

- (b) For each of the following *sentences*, state clearly whether the sentence is *valid* (belongs to the language of the grammar). If it does then provide a leftmost derivation for it; else explain (as precisely as possible) why it does not. Ignore all whitespace.
 - (1) i + n * i / n i
 - (2) i * n + i n / i / n

[additional space for earlier material]



[additional space for earlier material]

5.	(10 pts.) trees.	Repeat Question 4 but provide abstract syntax trees (ASTs) instead of parse

[additional space for earlier material]