COP 5555 Programming Language Principles

Homework 2

Assigned: May 25, 2011
Deadline: 11:59pm June 1, 2011

*Note: Please submit your solution to the e-learning system by the deadline.
No late submission is allowed.

1. Problem 1 (25 Points)

Consider the following grammar:
A -> B (', B)+ => "a"
  -> B
B -> B ' & ' C => "b"
  -> C
C -> D '#' C => "c"
  -> D
D -> <identifier>

Write the skeleton of a recursive descent parser for this grammar, including 'BuildTree' statements that will build the AST bottom-up, for the original grammar.

*Hint: See Powerpoint lecture 6.*

2. Problem 2 (25 Points)

The following RPAL program is overly parenthesized; remove all superfluous parentheses, i.e. minimize the parentheses without changing the meaning of the program.

Also, format the program by writing it on several lines with the appropriate indentation.

let (x =((a * b)-c)) in ((e & f) -> (g -> x ** (x ** 2)) | y)
| (f(x) where ((f(y) = (y * 2)) and (x = (2 + (x ** 2))))))
3. Problem 3 (25 Points)

Write, test, and debug an RPAL program that computes the "tuple reverse" function:

Rev(4,'hello',(3,4),true) = (true,(3,4),'hello',4).

Please write them on several lines with the appropriate indentation.

4. Problem 4

Explain the behavior of the following RPAL program, and the type of each identifier which appeared:

Let Prod N = P 1 N where rec P Cum N = N eq 1 -> Cum | P (Cum*N)
in Print (Prod 5 4 3 2 1)

Make an honest attempt at figuring it out BEFORE asking the RPAL interpreter.