

Assignment 1:

This week's total work:

[A] Based on today's discussion, continue reading 3-addr code generation. You may contact me any clarification.

From Dragon Book, Read Sections 8.1 and 8.6 only (Generate Code through Back-patching only)

[B]

Extend your last Lab Work (An LALR (1) Parser for Tiny-C lang) to generate 3-addr code for a simple c-prog having the following 4 type of statements:

1. Assignment statements : $a = b + c$; $a = 10$;
2. If-else statement: if (a) $a = 10$; else $b = 20$;
3. While statement : Put a while loop in the above two statements .

Submit by Sun, Dec 8, 2019. The students who will submit the above, will get additional bonus marks.

Assignment 2:

Hope most of you have completed the PL Design & Implementation (PLDI) Assignment for parsing a programme written in Tiny-C PL.

{

Only a handful of students had submitted the Design Assignment (DA) I about CFG for Tiny-C. None of you have shown generation of simple C-programs using the CFG.

Pls understand that this is the running assignment, which will continue till end of the semester.

If you have not done so already, then you should do so that you can do DA II.

DA II:

In first part of this assignment, you make the CFG of DA I, suitable for predictive parsing, i.e., you have to remove left-factor and left-recursion.

In second part, write the CFG in pictorial diagrams using EBNF notations.

}

Assignment 3:

You have to submit lex, yacc code for parsing c-code using LALR(1) parse

Assignment 4:

Make a LALR(1) for Tiny - C Prog Lang as designed in DA

Consider the following CFG, in which terminals are in **bold-face**:

stmt_seq	→	stmt_seq ; stmt stmt
stmt	→	if_stmt assign_stmt
if_stmt	→	if exp then stmt_seq end if exp then stmt_seq else stmt_seq end
assign_stmt	→	id = exp
exp	→	0 1 int id

1) Calculate First & Follow sets for every non-terminal. Then deduce w/o making an LL(1) table, is the above CFG suitable for LL(1) parsing? Make an LL(1) table.

2) Modify the above CFG suitable for predictive parsing. Repeat the above with the modified CGF.

Submit by Wed, Oct. 30/10 1200 noon.