MAHARAJA AGRASEN INSTITUTE OF TECHNOLOGY

Department of Computer Science and Engineering

(Academic Session 2023-24)

Subject Name: compiler Design
Subject Code: CIC- 303
Assignment No.-1.

Q1. Write a grammar to declare int or float type variables in C. **[CO2] Marks (5)**

Remove left recursion, if any
Left factored it, if required.
Construct Predictive parser
Check whether the resultant grammar is LL(1) or not
Check whether the following statements follow the rules or not:
int a,b;
float a, int b;

Q2. Given the following CFG grammar **[CO1] Marks(5)**

G = ({S,A,B}, S, {a, b, x), P) with

P: (1) S → A (2) S → xb (3) A → aAb (4) A → B (5) B → x

For this grammar answer the following questions:

a) Compute the set of LR(1) items for this grammar and the corresponding DFA.

b) Construct the corresponding LR parsing table.

c) Would this grammar be LR(0)? Why or why not

d) Show the stack contents, the input and the rules used during parsing for the input string w = axb$

e) Would this grammar be suitable to be parsed using a top-down LL parsing method? Why?

Q3: Parse Table **(CO1) Marks 5**

Consider the following grammar which is not SLR, i.e. the SLR parsing table will have conflicts:
$S \rightarrow M\ a$
$S \rightarrow b\ M\ c$
$S \rightarrow d\ c$
$S \rightarrow b\ d\ a$
$M \rightarrow d$
Construct the SLR parse table and answer the following:

a. How many conflicts are there in the resulting parsing table?
b. Show each pair of items that cause a conflict.
c. What are the viable prefixes for the conflicting states.
d. Explain in a clear language the reason behind each of the conflicts

Q4: SLR vs CLR vs LALR **CO2 Marks 5**

For the grammar
$S \rightarrow id[\ E\ ]\ :=\ E$
$E \rightarrow E\ +\ T \mid T$
$T \rightarrow T\ *\ F \mid F$
$F \rightarrow (\ E\ )\ \mid\ id$
answer the following questions:

a. In the automaton for the SLR parser, there is a state with a self-loop. Identify the state through all the viable prefixes that takes the automaton to this state. You may represent the set of viable prefixes by regular expressions over the grammar symbols.
b. Show an example string for which the SLR and CLR parsers for the grammar behave differently and, in not more than five lines, explain the difference.
c. Show an example string for which the CLR and LALR parsers for the grammar behave differently and, in not more than five lines, clearly explain the difference.