VIDYA JYOTHI INSTITUTE OF TECHNOLOGY

(An Autonomous Institution)

(Accredited by NAAC & NBA, Approved by AICTE New Delhi & Permanently Affiliated to JNTUH)

Aziznagar Gate, C.B. Post, Hyderabad – 500 075

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

III B.Tech II Semester

COMPILER DESIGN & DATA MINING LAB (A16590)

Course outcomes: After completing this course the student must demonstrate the knowledge and ability to:

1. Understand the role of lexical analyzer.
2. Identify the working of compiler construction tools – LEX, YACC and Parser
3. Derive machine code from intermediate code.
4. Able to understand WEKA tool.
5. Ability to add mining algorithms as a component to the existing tools.
6. Able to apply mining techniques for realistic data.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

III B.Tech II Semester

COMPILER DESIGN & DATA MINING LAB (A16590)

LIST OF EXPERIMENTS

Part - A: Compiler Design

<program> ::= <block>
<block> ::= { <variabledefinition> <slist> } | { <slist> }
<variabledefinition> ::= int<vardeflist>;
<vardeflist> ::= <vardec> | <vardec>, <vardeflist>
<vardec> ::= <identifier> | <identifier> [<constant>]
<slist> ::= <statement> | <statement>; <slist>
<statement> ::= <assignment> | <ifstatement> | <whilestatement> | <block> | <printstatement> | <empty>
<assignment> ::= <identifier> = <expression> | <identifier> [<expression>] = <expression>
<ifstatement> ::= <bexpression> then <slist> else <slist> endif | if <bexpression> then <slist> endif
<whilestatement> ::= while <bexpression> do <slist> enddo
<printstatement> ::= print (<expression>)
<expression> ::= <expression> <additionop> <term> | <term> | addingop> <term>
<bexpression> ::= <expression> <relop> <expression>
<relop> ::= < | <= | == | >= | > | !=
<addingop> ::= + | -
<term> ::= <term> <mulitop> <factor> | <factor>
<multop> ::= * | /
<factor> ::= <constant> | <identifier> | <identifier> [<expression>] | (<expression>)
<constant> ::= <digit> | <digit> <constant>
<identifier> ::= <identifier> <letterordigit> | <letter>
<letterordigit> ::= <letter> | <digit>
<letter> ::= a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z
<digit> ::= 0|1|2|3|4|5|6|7|8|9
<empty> has the obvious meaning
Comments (zero or more characters enclosed between the standard C / Java style comment brackets /*...*/) can be inserted.
A simple program written in this language is:
{
  int a[3], t1, t2;
  t1 = 2;
  a[0] = 1; a[1] = 2; a[t1] = 3;

```
t2 = -(a[2] + t1 * 6)/ a[2] - t1);
if t2 > 5 then
print(t2);
else
{
  int t3;
  t3 = 99;
  t2 = -25;
  print(-t1 + t2 * t3); /* this is a comment on 2 lines */
}
end if
}
```

Week 1: Design a Lexical analyzer for the above language. The lexical analyzer should ignore redundant spaces, tabs and newlines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value.

Week 2: Implement the lexical analyzer using JLex, flex or lex or other lexical analyzer generating tools.

Week 3: Design Recursive decent parser for the given language.

Week 4: Design LALR bottom up parser for the given language.

Week 5: Convert the BNF rules into YACC from and write code to generate abstract syntax tree.

Week 6: Design a Machine code for the Given Intermediate code.

Part - B: Data Mining

Week-1: Analyze a sample dataset using classification technique in WEKA Tool.

Week-2: Demonstrate Apriori based Association Rule Mining.

Week-3: Demonstrate FP –growth based Association Rule Mining.

Week-4: Demonstrate K-means Clustering Algorithm.

Week-5: Demonstrate PAM based Clustering.

Week-6: Analyze German Credit Data Assessment using various Data mining Techniques**.**

LIST of EQUIEMENTS

S.No	Name Of the equipment	Configuration	QTY
1	Desktop	DELL OPTIPLEX 3050 i3 @ 3.20 G.Hz 8GB RAM 500 GB HDD	60
2	LCD PROJECTOR	Panasonic	1
3	D-LINK Switch	24 ports	3
4	DATAMINIG Software	WEKA 3.7 , Pentaho 4.4,	
5	COMPLIER DESIGN Software	Flex for windows LEX and Yacc, TURBO C	