**CYCLE 1**

1. To Implement a lexical analyzer in C.
2. To write a program for implementing a Lexical analyser using LEX tool
3. LEX Program to count the number of lines and number of characters in an input.
4. LEX Program to remove space, tab or newline
5. LEX Program to count the number of vowels and consonants
6. Generate a YACC specification to recognize a valid identifier which starts with a letter followed by any number of letters or digits.
7. Generate a YACC specification to recognize a valid arithmetic expression that uses operators +, – , \*,/ and parenthesis.
8. Implementation of Calculator using LEX and YACC
9. Write a LEX Program to convert the substring abc to ABC from the given input string

**CYCLE2**

1. Implementation of DFA
2. Program to find the Epsilon closure of all states of

NFA

1. Implementation of Top down Parser
2. Implementation of Bottom up Parser
3. Simulation of code optimization Techniques
4. Implement Intermediate code generation for simple expressions
5. Implement Backend of the Compiler