# INDEX

| S.N. | Labs | Date | Signature |
|---|---|---|---|
| 1. | **Implementing Lexical Analyzer:** Implement a lexical analyzer to recognize identifiers, keywords, comments, strings, operators, and constants. Display token type and lexeme. | 2082/05/11 | |
| 2. | **Implementing Symbol Table Operations:** Implement a symbol table to demonstrate the operations: insert, lookup and display. Maintain attributes such as identifier name, type, and scope. | 2082/05/12 | |
| 3. | **Recursive Descent Parser:** Implement a recursive descent parser for the grammar. $S \rightarrow a\,A\,b$ $A \rightarrow a \mid \varepsilon$ | 2082/05/13 | |
| 4. | **Implementation of Shift-Reduce Parsing:** Implement Shift-Reduce parsing for the following grammar and input string: a +a*a. $E \rightarrow E+E$ $E \rightarrow E*E$ $E \rightarrow (E)$ $E \rightarrow a$ | 2082/05/16 | |
| 5. | **Write a program to generate closure set on LR(0) items for the grammar:** $S \rightarrow A\,B$ $A \rightarrow a$ $B \rightarrow b$ | 2082/05/17 | |
| 6. | **Intermediate Code Generation:** Write a program to generate three-address code for arithmetic assignment statement. | 2082/05/18 | |
| 7. | **Target Code Generation:** Write a program to generate target code for a simple register-based machine. | 2082/05/19 | |