Lab Manual
Compiler Construction

Name________________________________

Year ____________________________

Batch________________________________

Roll No ____________________________

Department: __________________________

Dept. of Computer Science and Information Technology

Sir Syed University of Engineering & Technology,

Karachi, Pakistan

INTRODUCTION

Programming languages are notations for describing computations to people and to machines. The world as we know it depends on programming languages, because all the software running on all the computers was written in some programming language. But, before a program can be run, it first must be translated into a form in which it can be executed by a computer.
The software systems that do this translation are called compilers

This book is about how to design and implement compilers. We shall discover that a few basic ideas can be used to construct translators for a wide variety of languages and machines.

Besides compilers, the principles and techniques for compiler design are applicable to so many other domains that they are likely to be reused many times in the career of a computer scientist. The study of compiler writing touches upon programming languages, machine architecture, language theory, algorithms, and software engineering.

In this preliminary chapter, we introduce the different forms of language translators, give a high level overview of the structure of a typical compiler, and discuss the trends in programming languages and machine architecture that are shaping compilers. We include some observations on the relationship between compiler design and computer-science theory and an outline of the applications of compiler technology that go beyond compilation. We end with a brief outline of key programming-language concepts that will be needed for our study of compilers.

CONTENTS

Lab Session No .	Objectives	Page No.

.

11. Transform your Grammar in the unambiguous form and perform the left recursion and left factoring from the grammar.

12. Find out the First () and Follow () of the grammar which you have completed in the previous lab.

13. Implement the Non recursive predictive parser by mapping the functions of First () & Follow () and making the Parsing table, perform the stack implementation by taking any program of your own language and shows its output

14. Transform Your Grammar in Augmented form in order to Implement SLR parser and find the closure and Goto operation.

15. Write the complete code of SLR parser with the help of making Transition diagram, also show its ouput. Take any program of your own language as an input and display the output.

LAB # 01

Compiler Basic Operation and IDE

What is IDE?

It is a software application that defines the visual representation of the location of the files easily and makes it more understandable for the user. It contains development tools such as text editors, code libraries, compilers, and test platforms and consists of at least build automation tools, and a debugger.

IDE can develop software applications by using a set of tools, which makes easier to write programs. The main objective of using IDE is that it allows coding quickly and efficiently. IDE includes built-in compilers, which convert the program into machine level code or byte code and saves a lot of time. You can also select multiple programming languages of your choice.

IDE's have some common features as listed below:

- Text editor: It provides a text editor to write and manage source code.
- Debugger: It uses debugging tools to identify the errors in the source code.
- Compiler
- Code Completion
- Programming language support
- Integration and use of plug-ins

Lab Task:

Make a program which recognizes the key strokes as you press different letters.

Example : if you press letter A then it displays the output as
" Letter A is pressed "

After pressing the 10 letters , it counts the # number of occurrences of the letters.

LAB # 02

PATTERN MATCHING CONCEPTS

Filing

Saving & opening a given text in a file.

Pattern Matching

Searching for a given string in a file

Replacement

Replace the searched string with the given string