**North Western University**



**LAB REPORT**

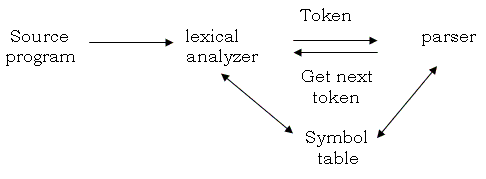
**Compiler Design(User Manual)**

**Course Code: CSE-4104**

|  |  |
| --- | --- |
| **SUBMITTED BY:**  Name : Sharmin Ara Shamila  ID :20201082010  Name : Richi Jerald  ID :20201083010  Name : Asif Salman  ID :20201079010  Section : “C”  Year : 4th  Semester: 1st  Department: CSE | **SUBMITTED TO:**  Md. Shymon Islam  Lecturer  Dept. Of CSE  North Western University  Khulna |

***Introduction***

Lexical analysis is the process of converting a sequence of characters (such as in a computer program of web page) into a sequence of tokens (strings with an identified “meaning”). A program that perform lexical analysis may be called a lexer, tokenize or scanner.



***Objectives***

**Token**

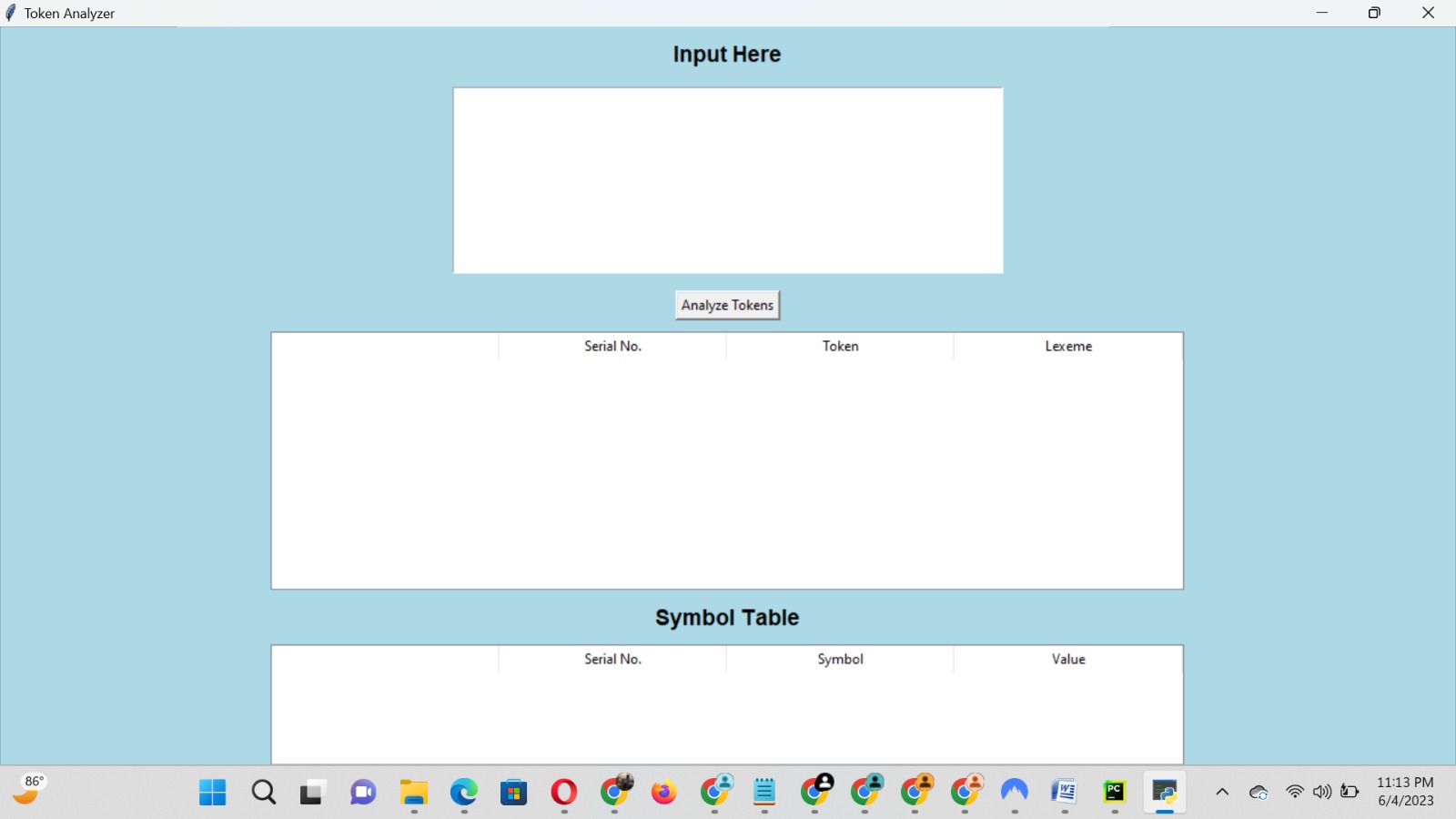
A token is a structure representing a lexeme that explicitly indicates its categorization for the Purpose of parsing. A category of token is what in linguistics might be called a part-of- speech. Examples of token categories may include “identifier” and “integer literal”, although the set of Token differ in different programming languages. The process of forming tokens from an input stream of characters is called tokenization. Consider this expression in the C programming language: Sum=3 + 2;

**Symbol Table Generation**

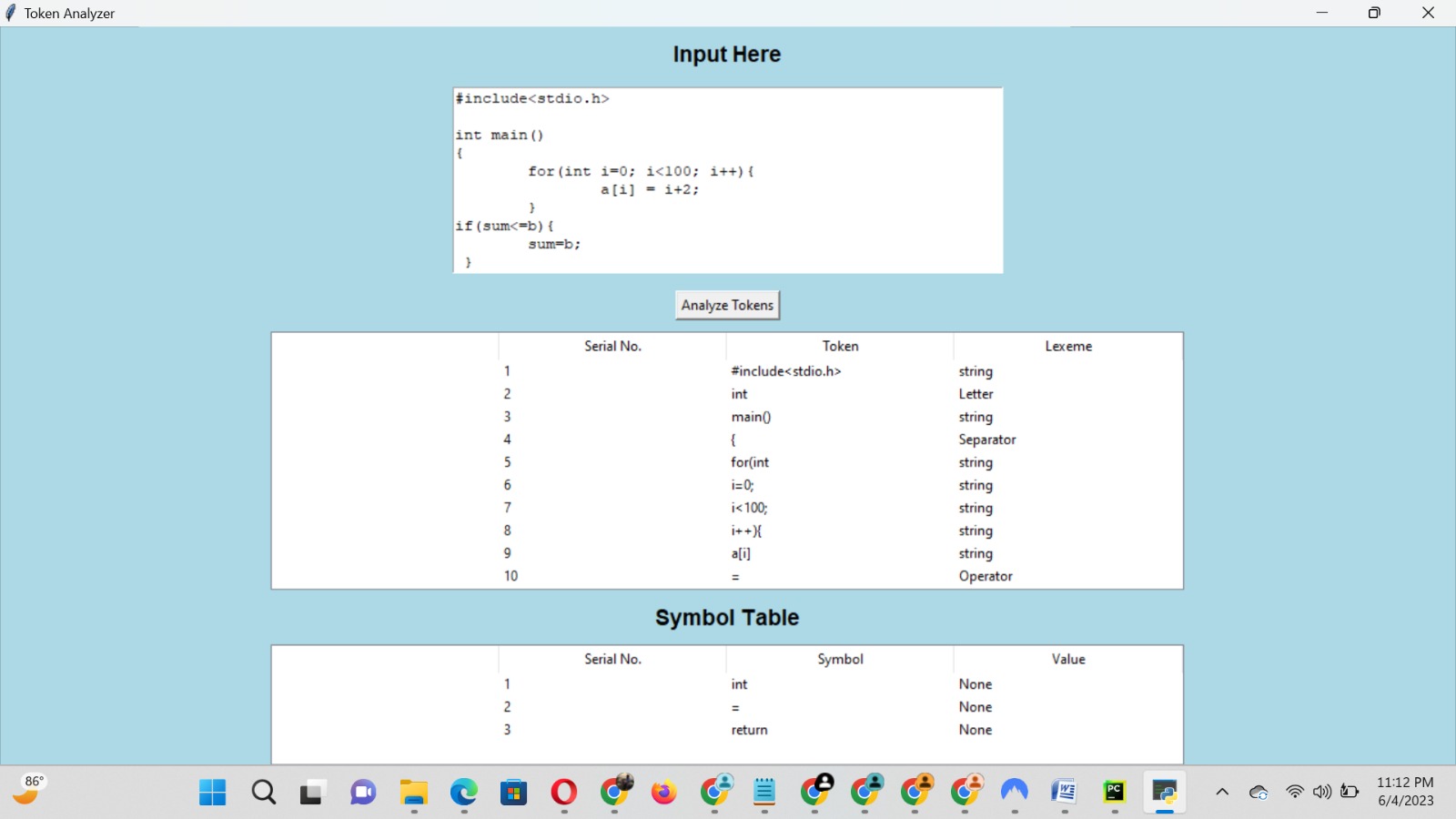
The lexer may generate a symbol table, which is a data structure used to store information about identifiers (variables, functions, classes) encountered during tokenization.

***Description***

Lexical analysis is the starting phase of the compiler. It gathers modified source code that is written in the form of sentences from the language preprocessor. The lexical analyzer is responsible for breaking these syntaxes into a series of tokens, by removing whitespace in the source code.



Project interface



The source code has been taken c language in input and Tokenization