**CSE2260**

**Principles of Programming Languages**

**Project 1 – Lexical Analyzer’s Report**

**Group Members:**

Feyzullah Asıllıoğlu – 150121021

Mohamad Nael Ayoubi – 150120997

Kadir Bat – 150120012

The project has been implemented in Java. It will prompt the user to enter the file name to be opened and start reading its contains character by character using BufferedReader class, while this process continues it will buffer the characters for efficient reading of characters, that is the reason we preferred using BufferReader instead of a scanner. Then since we are gonna print on both console and an output text file, we used PrintWriter class, before that it will create a text file in the project folder to be written on.

The reader will continue reading in an infinite loop till it reaches the end of text file i.e., stream. In each iteration it will check if character is either single quote or double quotes if not it will start initializing three strings that are tempTokenId, token, and tokenNumber based on these variables and if character is a whitespace will go through the other if else statements that will check either of them are valid Number, Keyword or a valid Identifier.

In isNumber(string)’s we were struggling a little bit in the regular expression of floating point number that contains a point, but we have solved it using a for loop and till we find a point it will save its index and continue with the regular expression.

We had no difficulty in making the keyword and boolean parts. We made the boolean part with the keyword. The boolean part is done inside (isKeyword(token)).

The bracket part was pretty easy. We did it with isBracket() and printBracket() methods. It was one of the parts we had no problems.

The string part was where we had the most difficulty. We were examining the given input one by one in our code. When we got to the String part, we couldn't do anything at first. We thought if we should examine the input line by line instead of examining the input one by one. We worked on this and still couldn't find a solution. We were constantly getting errors when different inputs were given. Then, we decided to examine the input one by one again. After much effort, we were able to do this part. We also had some problems in the char part, but we overcame it with the knowledge we experienced while making the string part.

Running the program: program prompts user for input file name.



After the program is run, both output is printed to the terminal and a file named output.txt is created.



Sample inputs and outputs:

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Output in the terminal:

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu metin içeren bir resim

Açıklama otomatik olarak oluşturuldu metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Output.txt file:

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu takvim içeren bir resim

Açıklama otomatik olarak oluşturuldu metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Another input:

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Output in the terminal:

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Output.txt file:

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu metin içeren bir resim

Açıklama otomatik olarak oluşturuldu