BRAC UNIVERSITY
Inspiring Excellence

Department of Computer Science and Engineering

Course Code: CSE 420	Credits: 1.5
Course Name: Compiler Design	Semester: Summer' 20

Lab 02

Introduction

I. Topic Overview:

The lab is designed to introduce the students to the basics concept of a compiler Design. As part of this activity students will write code for a fixed set of regular expression without using any built-in libraries. Basic techniques of coding and required tools will also be shown to students.

II. Lesson Fit:

The lab gives a hand on experience of the knowledge of theory class of Lexical Analysis.

III. Learning Outcome:

After this lecture, the students will be able to:

a. Understand and visualize the Lexical Analysis phase.

b. Converting regular expression to DFA.

c. Creating own version of Lexical recognizer.

IV. Anticipated Challenges and Possible Solutions

a. Mapping the regular expression to DFA will be challenging.

Possible Solutions:

a. Use regular expression to guide the DFA.

b. Use methods of java switch case construct.

V. Acceptance and Evaluation

If a task is a continuing task and one couldn't finish within time limit, he/she will continue from there in the next Lab, or be given as a home work. He/ she have to submit the code and have to face a short viva. A deduction of 30% marks is applicable for late submission. The marks distribution is as follows:

Code: 0%

Viva: 100%

VI. Activity Detail

Activity Detail

a. Hour: 1, 2

Discussion: Converting Regular Expression to Transition Diagram or DFA.

Problem Task: Task 1 (page 3-4)

b. Hour: 3

Discussion: Code the equivalent DFA for the RE.

Problem Task: Task 2 (page 3-4)

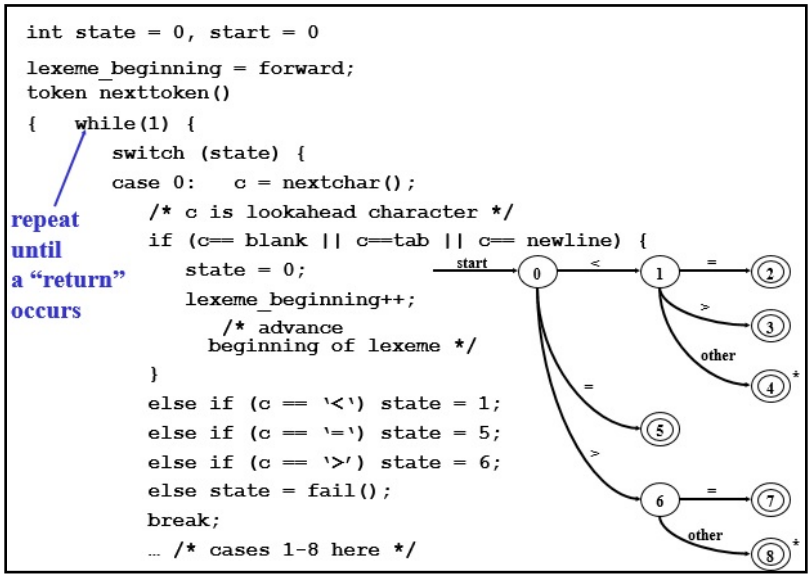
Assignment 3: Problem Description

In this assignment, you will work on regular expression. For simplicity, we will assume that there is a fixed set of regular expressions. We will not consider out of these. But you must not use any built-in method or package in your implementation. If you need any method, you will write that. In Regular Expression (RE), '*' means occurrence of zero of more characters, '+' indicates happening of one or more characters, '?' means only once or not at all occurrence, '[ ]' indicates happening of inclusive characters, '^' indicates that next characters will not be used in the pattern, '[a-d]{3}' indicates that valid string will be exactly of length 3 inclusively using a, b, c, d. The following table contains a fixed set of RE that will be used in our assignment.

Description	RE	Valid	Invalid
Email Address	Find yourself	abc@gmail.com	123abc@gmail.com
Web Address	Find yourself	www.abc.com	123.abc.com

Lab 3: Activity List

Task 1: The best way to approach this problem is to draw DFA and translate the DFA in code. Consider the following Transition Diagram for relational operators.



Task 2: User will be asked first to input an integer value n followed by n lines of Strings. You have to find out whether it is email or web address along with its line number. **Remember, in no way you can use any kind of built in Regular Expression for this task.**

Input:

2
dilrubashowkat@gmail.com
www.dilrubashowkat.com

Output:

Email, 1
Web, 2