

CEN405

FORMAL LANGUAGES AND AUTOMATA THEORY

Spring2018

LAB 10: DFA, NFA, RegEx, CFG, PDA and TM Design using JFLAP

In this LAB you are given two executable JARs that you can use for designing various FAs, RegEx, CFG, PDA and TMs. Before starting you have to download the following this JAR:

- [regexChecker.jar](#)
- [JFLAP.jar](#)

You can run the **regexChecker.jar** program with the command

```
java -jar regexChecker.jar
```

The program allows you to type in a regular expression and then match it against some strings. For each string, the program will tell you whether or not the string matches the regular expression. A sample usage is given as follows:



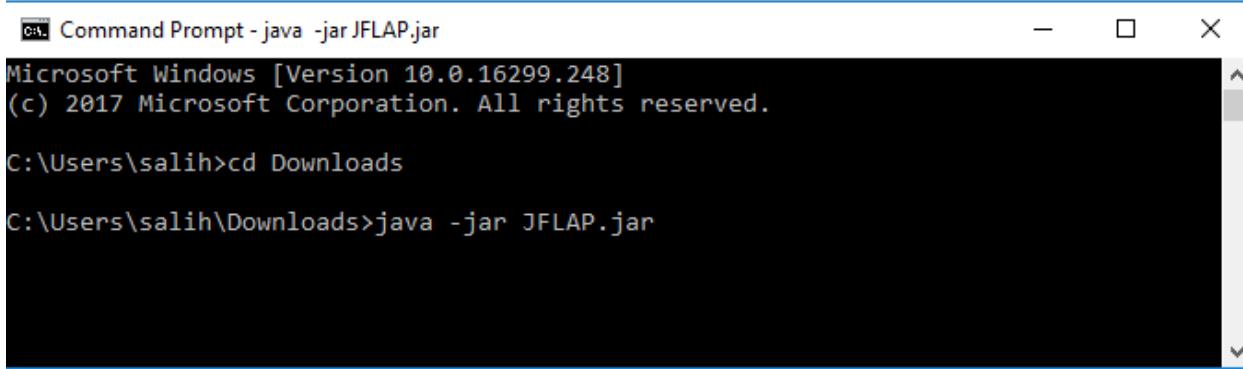
A screenshot of a Windows Command Prompt window titled "Command Prompt - java -jar Downloads\regexChecker.jar". The window shows the following interaction:

```
C:\Users\salih>java -jar Downloads\regexChecker.jar
Enter a regular expression (use 'e' for epsilon): a*bc*
Enter strings (type 'quit' when done)
abc
ACCEPT
acc
REJECT
abccccccccc
ACCEPT
bbccccc
REJECT
bccc
ACCEPT
```

You can run the **JFLAP.jar** program with the command

```
java -jar JFLAP.jar
```

A sample usage is given as follows:



A screenshot of a Windows Command Prompt window titled "Command Prompt - java -jar JFLAP.jar". The window shows the following text:
Microsoft Windows [Version 10.0.16299.248]
(c) 2017 Microsoft Corporation. All rights reserved.
C:\Users\salih>cd Downloads
C:\Users\salih\Downloads>java -jar JFLAP.jar

Your Task

Design Turing Machines recognizing the following languages:

- 1) Give a DFA for $\Sigma = \{0, 1\}$ and strings that have an odd number of 1's and any number of 0's.
- 2) Give a DFA for $\Sigma = \{a, b\}$ that accepts any string with aababb as a substring.
- 3) Give an NFA for $\Sigma = \{0, 1\}$ and that accepts all binary strings that end with 101.
- 4) Give an NFA for RegEx: $a^* | (ab)^*$
- 5) Give an NFA for The language $\{w \in \Sigma^* \mid w \text{ contains at least two 0s, or exactly two 1s}\}$ with six states.
- 6) Write regular expression (RegEx) that define the language L. L=All strings of 0's and 1's such that the third position from the right end is 1.

7) Write regular expression (RegEx) that define the language L. $\Sigma = \{a, b, c, A, B, C\}$. L=All strings of lower-case letters that are in sorted order.

8) Write CFG for the languages L_1 , L_2 , and L_3 .

- $L_1 = \{a^i b^j c^k : i = k\}$
- $L_2 = \{a^i b^j c^k : i = j\}$
- $L_3 = \{a^i b^j c^k : i = j + k\}$

9) Design PDAs for the following languages:

- $L_1 = \{a^i b^j c^k : i = k\}$
- $L_2 = \{a^i b^j c^k : i = j\}$
- $L_3 = \{a^i b^j c^k : i = j + k\}$

10) Design a Turing Machine (TM) recognizing the following language:

$$L = \{ a^i b^j c^k \mid i \neq j \text{ and } i, j, k \geq 1 \}$$