Middle East Technical University Department of Computer Engineering

CENG 280

Formal Languages and Abstract Machines

Spring 2022-2023

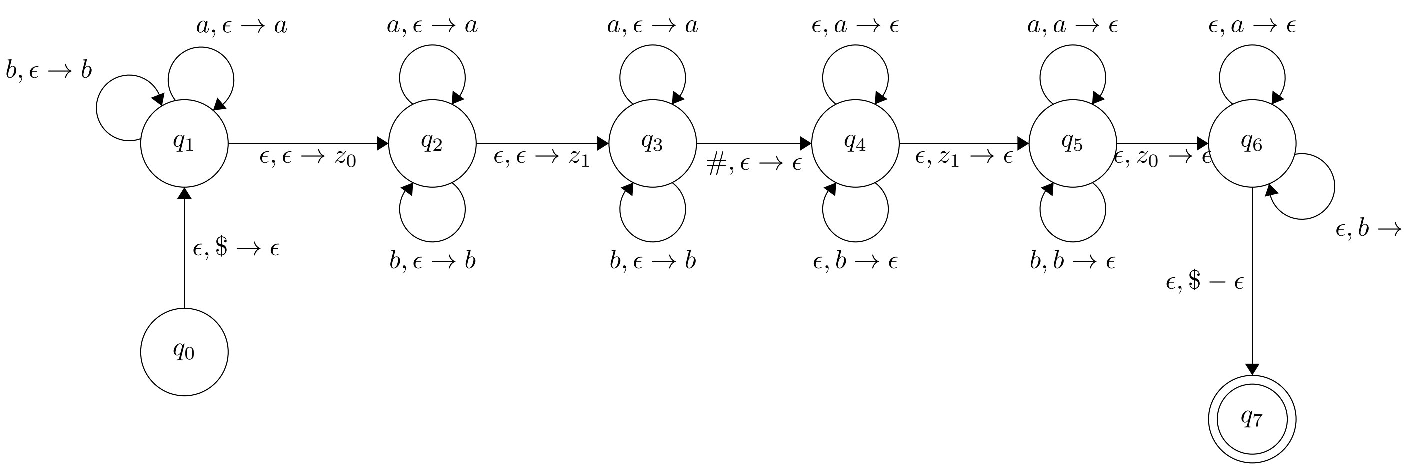
Homework 4

Name Surname: BAŞAR YILMAZ
Student ID: 2644409

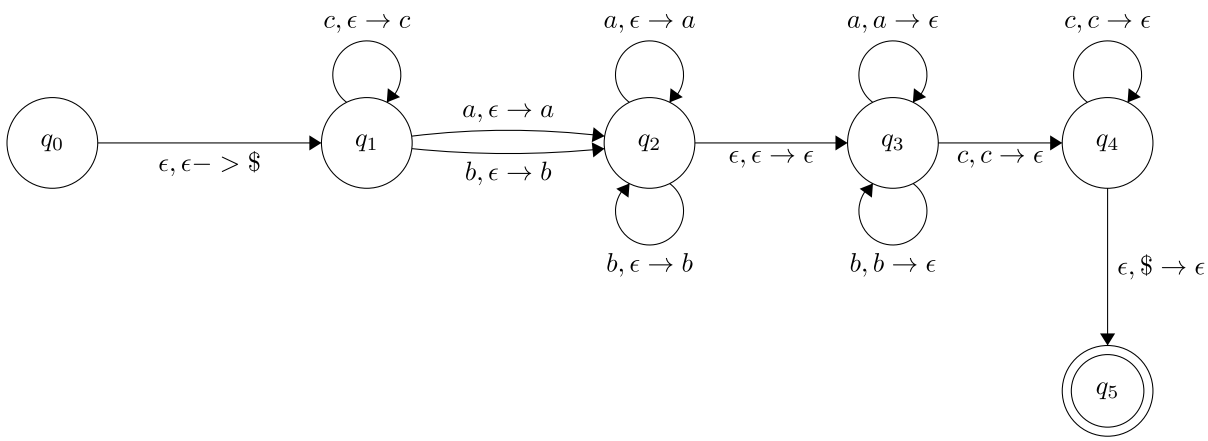
Answer for Q1

1. $M = (K, \Sigma, \Gamma, \Delta, S, F)$

$K = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7\}$
$\Sigma = \{a, b\}$
$\Gamma = \{a, b, \$, z_0, z_1\}$
$s = q_0$
$F = \{q_7\}$
$\Delta =$
$((q_0, \epsilon, \epsilon), (q_1, \$)),\ ((q_1, a, \epsilon), (q_1, a)),$
$((q_1, b, \epsilon), (q_1, b)),\ ((q_1, \epsilon, \epsilon), (q_2, z_0)),$
$((q_2, b, \epsilon), (q_2, b)),\ ((q_2, a, \epsilon), (q_2, a)),$
$((q_2, \epsilon, \epsilon), (q_3, z_1)),\ ((q_3, a, \epsilon), (q_3, a)),$
$((q_3, b, \epsilon), (q_3, b)),\ ((q_3, \#, \epsilon), (q_4, \epsilon)),$
$((q_4, \epsilon, a), (q_4, \epsilon)),\ ((q_4, \epsilon, b), (q_4, \epsilon))$
$((q_4, \epsilon, z_1), (q_5, \epsilon)),\ ((q_5, b, b), (q_5, \epsilon))$
$((q_5, a, a), (q_5, \epsilon)),\ ((q_5, \epsilon, z_1), (q_6, \epsilon))$
$((q_6, \epsilon, a), (q_6, \epsilon)),\ ((q_6, \epsilon, b), (q_6, \epsilon))$
$((q_6, \epsilon, \$), (q_7, \epsilon))$



2. $M = (K, \Sigma, \Gamma, \Delta, S, F)$
$K = \{q_0, q_1, q_2, q_3, q_4, q_5\}$
$\Sigma = \{a, b, c\}$
$\Gamma = \{a, b, c, \$\}$
$s = q_0$
$F = \{q_5\}$
$\Delta =$
$((q_0, \epsilon, \epsilon), (q_1, \$)),\ ((q_1, c, \epsilon), (q_1, c)),$
$((q_1, a, \epsilon), (q_2, a)),\ ((q_1, b, \epsilon), (q_2, b)),$
$((q_2, b, \epsilon), (q_2, b)),\ ((q_2, a, \epsilon), (q_2, a)),$
$((q_2, \epsilon, \epsilon), (q_3, \epsilon)),\ ((q_3, a, a), (q_3, \epsilon)),$
$((q_3, b, b), (q_3, \epsilon)),\ ((q_3, c, c), (q_4, \epsilon)),$
$((q_4, c, c), (q_4, \epsilon)),\ ((q_4, \epsilon, \$), (q_5, \epsilon))$



Answer for Q2

1. Assume we have a CFG such that $G = (V, \Sigma, R, S)$.
$V = \{S\}$
$\Sigma = \{a, b\}$
$S = S$
$R = S \to aSa \mid bSb \mid aa \mid bb$

If we conducted Kleene Star operation on this CFG its R would be changed as follows:
$R = S \to aSa \mid bSb \mid aa \mid bb \mid SS \mid \epsilon$

As it can be easily observed, adding only $S \to SS$ operation to CFG will not be enough to find the Kleene Star of the grammar. ϵ should be included in every case if it does not exist.

Answer for Q3

1. L_1 is an S-CFL, whenever we read an a push a symbol to the stack, and whenever we read b pop the symbol from the stack. If we have our stack empty and the string is read completely, the PDA accepts..

L_2 is not an S-CFL, we need to keep track of the number of occurrences of the a's and b's.
For example, let's say we push a symbol when we read a a and pop one when we read a b. Given the string $abbaa$ which is in the language. When we read the second b there won't be any symbol to pop, so

we cannot count the number of occurrences accurately. We need an extra stack symbol in this case.

L_3 is not an S-CFL. We need to keep track of the number of occurrences. Such that we have to recognize between $a^n b^n$ and $b^m c^m$, if we can push another symbol when we are done with the first part, $a^n b^n$, it can be accepted; yet, in this case, it is not an S-CFL.

2. $L = \{a^{n+m} b^n c^m \mid m, n \in N\}$
$G = \{V, \Sigma, R, S\}$
$V = \{S, X\}$
$\Sigma = \{a, b\}$
$S = S$
$R = S \rightarrow aSc \mid X$
$X \rightarrow aXb \mid \epsilon$

3. A simple "counter" would work. Basically, it has the same purpose as what we achieve with the one-symbol stack. Such that, pushing a symbol to the stack increases the counter while popping a symbol from the stack decreases the counter.

4. We added a counter to finite automata, which keeps track of the difference between the number of occurrences of two different symbols in the input string. This counter is necessary to recognize some context-free languages that require comparing the number of occurrences of two different symbols in the input. Adding a counter makes the machine more powerful and able to recognize some context-free languages. The machine reads the input from left to right, and at each step, it reads a symbol and either increments or decrements the counter based on the symbol. If the counter becomes negative, the machine rejects the input. If the input is entirely read and the counter is zero, the machine accepts the input. The machine has no output; it only accepts or rejects the input.

5. Let's say we have an S-CFL which is the L_1 in the first part. In order to take it's complement $L_1' = \{w \mid w$ has different number of a's and b's or its placement different than $a^n b^n$ $w \in \{a, b\}^*\}$. In this case, we cannot say L_1' is an S-CFL as we need two different stack symbols to keep track of the number of a's and b's, similar to L_2. So S-CFL is not closed under complementation.