Middle East Technical University Department of Computer Engineering

CENG 280

Formal Languages and Abstract Machines

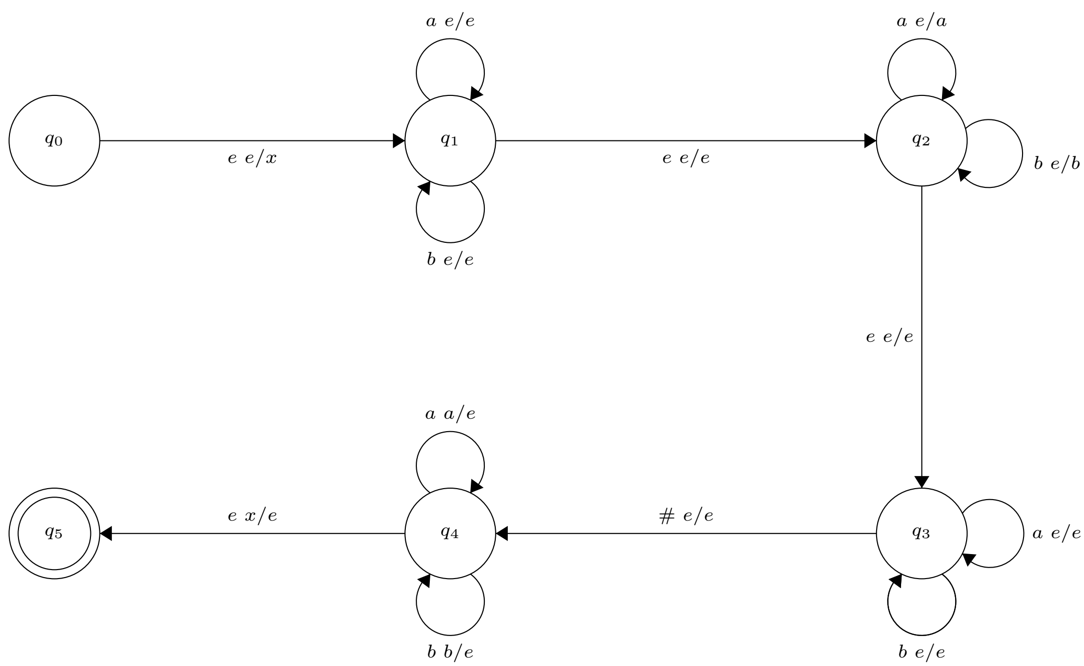
Spring 2022-2023

Homework 4

Name Surname: Burak YILDIZ
Student ID: 2449049

Answer for Q1

1)



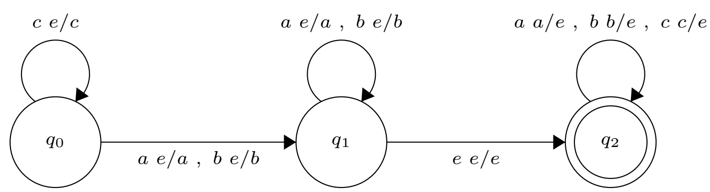
Let $M = (K, \Sigma, \Gamma, \Delta, s, F)$ then:

$K = \{q_0, q_1, q_2, q_3, q_4, q_5\}$
$\Sigma = \{a, b, \#\}$
$\Gamma = \{a, b, x\}$
$s = \{q_0\}$
$F = \{q_5\}$
Δ contains the following transitions:

$((q_0, e, e)(q_1, x))$

$((q_1, a, e)(q_1, e))$
$((q_1, b, e)(q_1, e))$
$((q_1, e, e)(q_2, e))$
$((q_2, a, e)(q_2, a))$
$((q_2, b, e)(q_2, b))$
$((q_2, e, e)(q_3, e))$
$((q_3, a, e)(q_3, e))$
$((q_3, b, e)(q_3, e))$
$((q_3, \#, e)(q_4, e))$
$((q_4, a, a)(q_4, e))$
$((q_4, b, b)(q_4, e))$
$((q_4, e, x)(q_5, e))$

2)



Let $M = (K, \Sigma, \Gamma, \Delta, s, F)$ then:
$K = \{q_0, q_1, q_2\}$
$\Sigma = \{a, b, c\}$
$\Gamma = \{a, b, c\}$
$s = \{q_0\}$
$F = \{q_2\}$
Δ contains the following transitions
$((q_0, c, e)(q_0, c))$
$((q_0, a, e)(q_1, a))$
$((q_1, b, e)(q_1, b))$
$((q_1, a, e)(q_1, a))$
$((q_1, b, e)(q_1, b))$
$((q_1, e, e)(q_2, e))$
$((q_2, a, a)(q_2, a))$
$((q_2, b, b)(q_2, b))$
$((q_2, c, c)(q_2, c))$

If the automaton reaches in this way the configuration (q_2, e, e) final state, end of input, empty stack then the input was indeed of the form of $c^n ww^R c^n$, and the automaton accepts.

Answer for Q2

Let L be the language $L = \{x\}$. The grammar for L is $G = (\{S\}, \{x\}, \{S \to x)\}, S)$

Now, let us add the rule $S \to SS$ to the grammar G, generating the new grammar $G' = (\{S\}, \{x\}, \{S \to x), (S \to SS)\}, S)$

It can be shown that G' does not generate L^*. Because L^* contains the empty string but $L(G')$ does not.

Answer for Q3

1)

a) Yes, we can push an 'x' symbol onto the stack for each ‘a’ that it reads. When it reads a 'b', it would pop an 'x' symbol from the stack. If the stack is empty at the end of the input, then the input is in L1.

b)No, if the numbers of b's are greater than numbers of a's the stack will eventually be empty but the input will not be finished since we cannot pop from an empty stack this is not an S-CFL.

c)No, we can push an 'x' symbol onto the stack for each 'a' then pop the 'x' symbol when b is read but when the stack is empty we can not pop anymore or decide do push again because of this it is not an S-CFL

2) An example is $L = \{a^n b^{2n} \mid n \geq 0\}$
The grammar for L is $G = (\{S\}, \{a, b\}, \{S \rightarrow aSbb\}, S)$

3) Counter

4) A counter will work like a stack, it will count +1 for push operation and -1 for the pop operations. S-CFL's only has one input symbol thats why counter works.

5) No it is not lets consider the L_1 in part 1 of this question. We can think this as the language that the numbers of a's and b's are equal. Complement of this language can be said as the language that the number of a's and b's are not equal as proven already if the numbers differ the language is not accepted by S-CFL.