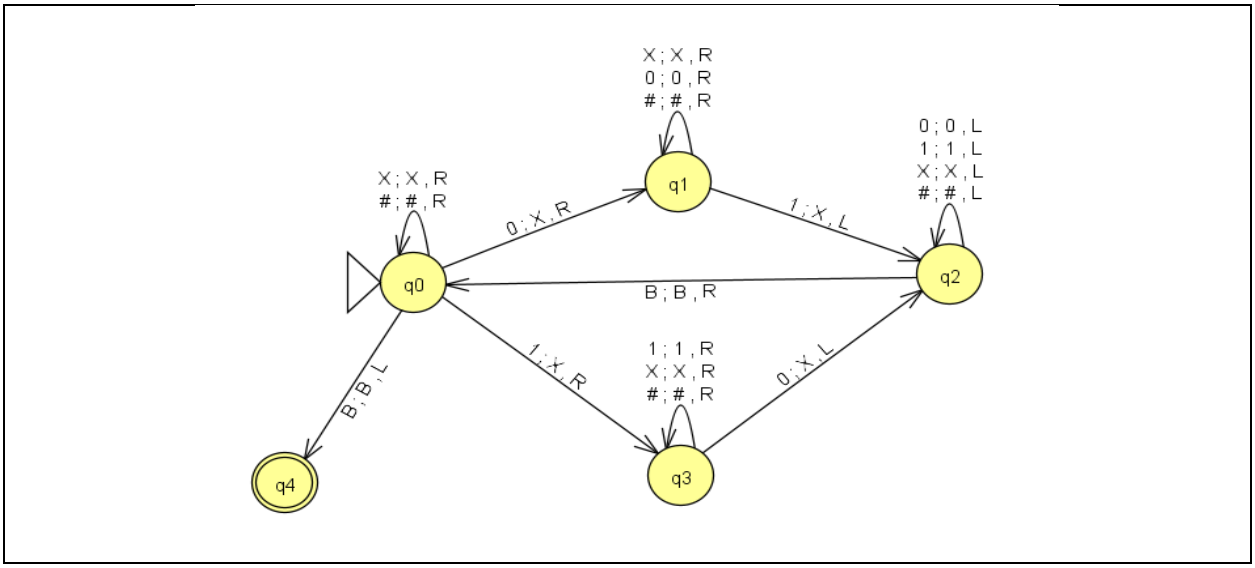
CS 361 – Homework 7

Total possible points: 75

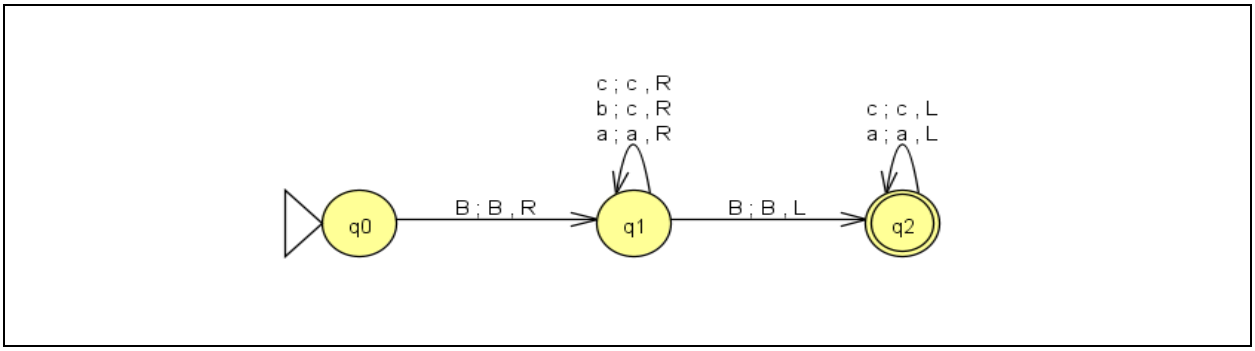
1. (20 points) Create a **state-diagram** representation of a **Turing Machine M_1** that accepts the language A = {w over {0 , 1, #}* | w contains an equal number of 0's and 1's}. 10# ∈ A, 01#10 ∈ A, and #001 ∉ A.



2. Consider the **TM M_2** defined by the set of transitions shown below. Here we use B to indicate the tape's "blank symbol."

δ	B	a	b	c
q_0	q_1,B,R			
q_1	q_2,B,L	q_1,a,R	q_1,c,R	q_1,c,R
q_2		q_2,a,L		q_2,c,L

a. (10 points) Give the **state diagram** of M_2 (assuming q_2 is a final state)



b. (4 points) **Describe** the result of a computation in M_2 (i.e., explain in your own words what M_2 does given a string). Do *not trace the computation*, instead provide a high level overview of the language associated with M_2

B changes to c and accepts every string.

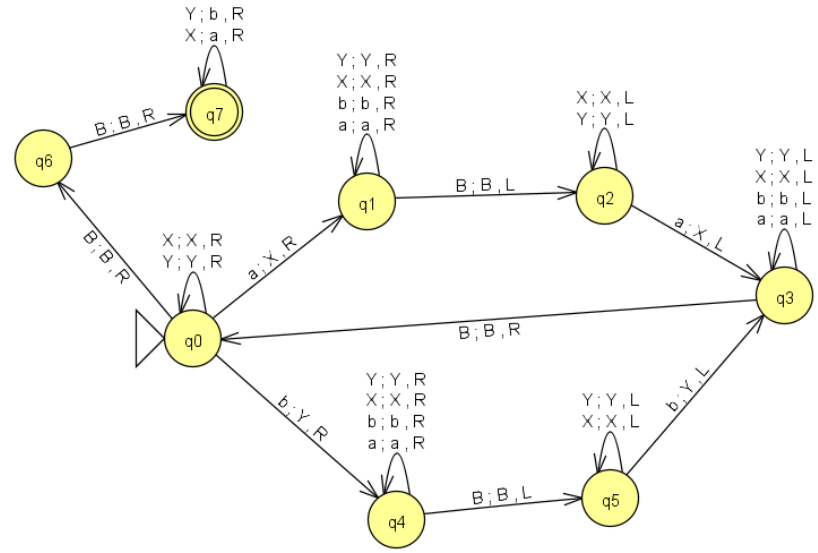
c. (3 points) **Trace** the computation of "BaabcaB"

q0 B a a b c a B
B a q1 a b c a B
B a a q1 b c a B
B a a c q1 c a B
B a a c c q1 a B
B a a c c a q1 B
B a a c c a q2 B
B a a c c q2 a B
B a a c q2 c a B
B a a q2 c c a B
B a q2 a c c a B
B q2 a a c c a B
Accept!

d. (3 points) **Trace** the computation of "BbcbB"

q0 B b c b B
B c q1 c b B
B c c q1 b B
B c c c q1 B
B c c c q2 B
B c c q2 c B
B c q2 c c B
B q2 c c c B
Accept!

3. (20 points) Construct a **state-diagram** representation of a **Turing Machine M_3** with input alphabet {a, b} that accepts strings of the form ww^R (where w is a string over {a, b}). The **final configuration** should be $q_f\ ww^R\ B$.



4. (15 points) Give a **description**[1] of a **TM M_4** that decides language B = {w over {0, 1}* | w contains twice as many 1's as 0's}. For example: 101 ∈ B, 011 ∈ B, and 001 ∉ B

There are three ways to get TM accepted that w contains twice as many 1's as 0's.

1. The first one is '011'. If I find the string 0, I pass the state until I find one string 1 and then I pass again until I find the second string 1. Other strings will stay before they are. And I go back to the beginning of the tape and find another string which accepted that w contains twice as many 1's as 0's.
2. The second string is '101'. If I find the string 1, I pass the state until I find the string 0 and then I pass again the state until I find the second string 1. Other string will stay before they are. And I go back to the beginning of the tape and find another string which accepted that w contains twice as many 1's as 0's.
3. The third string is '110'. If I find the string 1, I pass the state until I find the second string 1 and then I pass the state again until I find the string 0. Other strings will stay before they are. And I go back to the beginning of the tape and find another string which accepted that w contains twice as many 1's as 0's.

If I can't find any string left, its done with final state.