CSci 435: Formal Languages and Automata

Instructor: Dr. M. E. Kim Date: September 11th, 2018

Due: 12:00 PM, September 21th (Fri.), 2018. (No Extension) Name: **\_\_\_Aaron Johnson\_\_\_**

**Home Assignment 2: 150 points + 30 points (optional)**

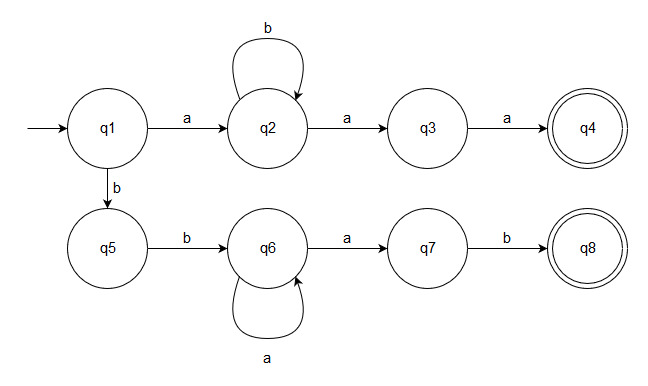
Q1. [10] Find all strings in L((*ab* + *b*)\* b (*a* + *ab*)\*) of length ***less than*** four.

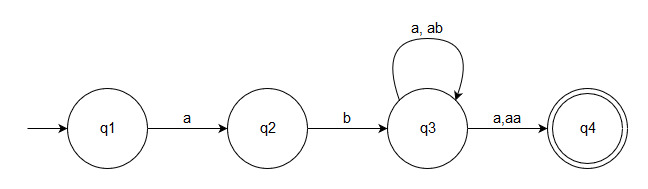
**b, bb, bbb, abb, ba, baa, bab**

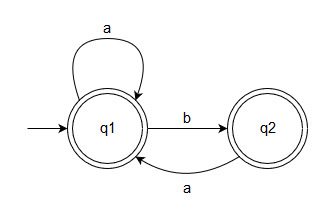
Q2. [10] Give a ***regular expression*** for the language L = {*anbm* | (*n*+*m*) is odd}.

**Not possible because state of n and m is required, making L not regular**

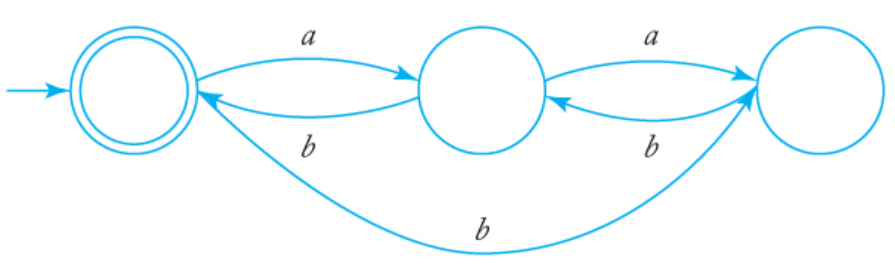
Q3. [10] Using the construction in Theorem 3.1, construct an NFA that accepts the complement of the

Language L(*ab*\**aa* + *bba*\**ab*).

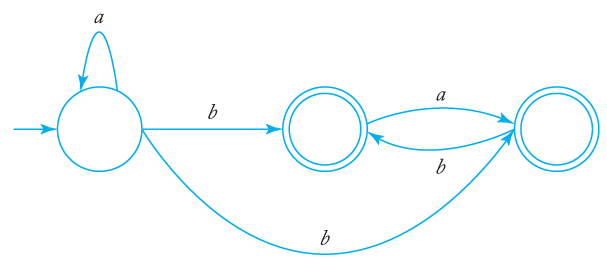
Q4. [20] Construct a ***minimal DFA*** that accepts the following language

1. L(*ab*(*a*+*ab*)\*(*a*+*aa*))
2. L((*aa*\*)\**b*)\*)

Q5. [20] Find ***regular expressions*** for the languages accepted by the following automaton.

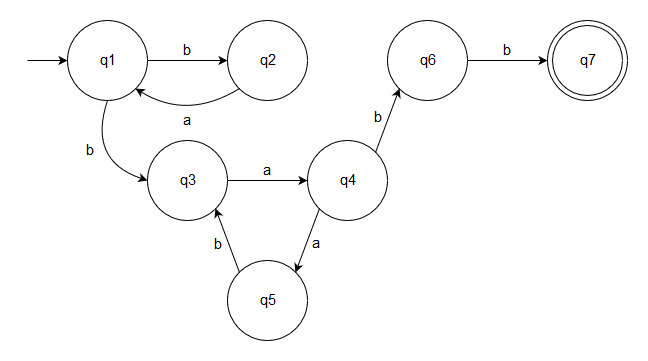


**(aabb)\*+(bbb)\***



**(a\*b (a+(ab)\*)\*+(b+(ba)\*)\*)**

Q6. [10] Construct a ***DFA*** that accepts the language generated by the *grammar*

 S → *ab*S | A, A → *ba*B, B → *a*A | *bb*

Q7. [20] Find a ***regular grammar*** that generates the language on Σ={a, b}

1. *L*(*aa*\*(*ab*+*a*)\*)

**S → aS | A | a**

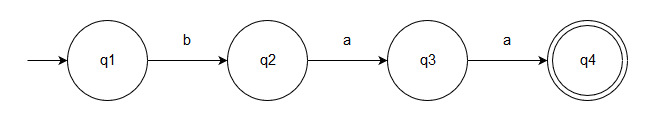
**A → ab | B**

**B → a | A**

1. the language consisting of all strings with no more than two *a*’s.

**S → bS | Sb | AaaA | b**

**A → bA | Ab | b**

Q8. [10]

1. Use the construction in Theorem 4.1 to find an DFA that accept ~~L((~~*~~ab~~*~~)\*~~*~~a~~*~~\*)~~ **L(*b*\**aab*\*)** ∩ L(*baa*\*).
2. [5, Optional] Give the regular expression for the above language ~~L((~~*~~ab~~*~~)\*~~*~~a~~*~~\*)~~ **L(*b*\**aab*\*)** ∩ L(*baa*\*) that is accepted by your DFA.

**L(baa)**

Q9. [10] The ***symmetric difference*** of two sets S1 and S2 is defined as

S1⊝S2 = {*x* | *x* ∈S1 or *x* ∈ S2, but *x* is not in both S1 and S2}.

Show that the family of regular languages is ***closed under symmetric difference.***

**L1**⊝**L2 = (L1** ◡ **L2) - L1** ∩ **L2 only if L1 and L2 don’t have the same words in them. If they did, they would be open under symmetric difference, thus the regular languages are closed under symmetric difference.**

Q10. [10] The family of regular languages are closed under arbitrary ***homomorphism***.

Prove or disprove h(L1 ∩ L2) =h(L1) ∩ h(L2) is a regular language where L1 and L2 are regular.

**Homomorphism is where the rules of one language are preserved when transformed into another language, so the regular languages are closed under arbitrary homomorphism.**

Q11. [10] Show that there exists an ***algorithm*** for determining if L1 **⊆** L2  for any regular languages

L1 and L2.

**L(A)⊆L(B) whenever L(A)∩(L(B))’ is empty. If we construct the automaton that accepts the intersection of A and the complement of B, we can use it to test if it accepts this empty language.**

Q12. [10] Pumping Lemma

1. Prove that the language L = {*anbkcn* | *n* ≥ 0, *k* ≥ *n* } is ***not regular***.

**It is impossible to generate any words using regular expressions for L without knowing the state of n and k, therefor, L is not regular.**

1. [10, Optional] Prove that the language L = {*w* | *na*(*w*) ≠ *nb*(*w*)} is not regular.

**Again, we can’t use state to keep track of the number of any substrings in a word of a regular language, so L is not regular.**

Q13. [15, Optional] Decide whether or not the following claims are true for all regular expressions r1 and r2. The symbol ≡ stands for an equivalence of regular expressions in the sense that both expressions denote the same language. Prove/Justify or disprove your answer.

1. r1\*(r1 + r2)\* ≡ (r1 + r2)\*

**Yes because (r1 + r2)\* already includes r1\* due to the + operator**

1. (r1 + r2)\* ≡ (r1\*r2\*)\*

**Yes because they both produce any number of r1 and r2 in any order**

1. (r1 r2)\* ≡ r1\*r2\*

**No, (r1r2)\* produces any number of r1r2, whereas r1\*r2\* may produce r1 by itself or r2 by itself.**