PES UNIVERSITY, Bangalore

(Established under Karnataka Act No. 16 of 2013)

Department of Computer Science & Engineering

Automata Formal Languages & Logic

Q&A - Properties of Regular Language and Pumping lemma

1) Prove that if L is regular then Prefix(L) is regular. Prefix(L) is the set of all strings which are a proper prefix of a string in L.

   We can construct a DFA to decide Prefix(L) by taking the DFA for L and marking all states from which an accept state is reachable as accept states. So, Prefix(L) must be regular.

2) Prove that Regular Sets are closed under MIN. MIN(R), where R is a regular set, is the set of all strings w in R where every proper prefix of w is in not in R. (Note that this is not simply the complement of PREFIX).

   We can construct a DFA to decide MIN(R) by taking the DFA for R and redirecting all outgoing arrows from all the accept states to a dead state. So, MIN(R) must be regular.

3) Prove that Regular Sets are NOT closed under infinite union. (A counterexample suffices).

   Consider the sets {0}, {01}, {0011}, etc. Each one is regular because it only contains one string. But the infinite union is the set $\{0^i1^i \mid i>=0\}$ which we know is not regular. So the infinite union cannot be closed for regular languages.

4) Prove that Regular Sets are NOT closed under infinite intersection.

   Solution:We know that

   $\{0^i1^i \mid i>=0\} = \{0\} \cup \{01\} \cup \{0011\} \cup ...,$

   Taking complements and applying DeMorgan's law gives us

   $\{0^i1^i \mid i>=0\}^c = \{0\}^c \wedge \{01\}^c \wedge \{0011\}^c \wedge ...,$

   Where we are using U to deonte union and ^ to denote intersection. Recall the complement of a regular language is regular, and hence the complement of a not-regular language is not regular. So we can conclude that the left hand side of the equation is not-regular, and each term in the intersection is regular. Therefore infinite intersection does not preserve regularity.

Automata Formal Languages & Logic

5) Are the following statements true or false? Explain your answer in each case. (In each case, a fixed alphabet is assumed.)

1. Let L`= L1 ∩L2. If L is regular and L2 is regular, L1 must be regular.
   Let L = L1∩L2. If L` is regular and L2 is regular, L1 must be regular. FALSE. We know that the regular languages are closed under intersection. But it is important to keep in mind that this closure lemma (as well as all the others we will prove) only says exactly what it says and no more. In particular, it says that:

   If L1 is regular and L2 is regular .Then L` is regular.

   Just like any implication, we can't run this one backward and conclude anything from the fact that L is regular. Of course, we can't use the closure lemma to say that L1 must not be regular either. So we can't apply the closure lemma here at all. A rule of thumb: it is almost never true that you can prove the converse of a closure lemma. So it makes sense to look first for a counterexample. We don't have to look far. Let L = Ø . Let L2 =Ø. So L`and L2 are regular. Now let L1 = {$a^i$: i is prime}. L1 is not regular. Yet L` = L1∩ L2. Notice that we could have made L2 anything at all and its intersection with Øhave been Ø . When you are looking for counterexamples, it usually works to look for very simple ones such as Øor Σ*, so it's a good idea to start there first. Øworks well in this case because we're doing intersection.Σ * is often useful when we're doing union.

2. Every subset of a regular language is regular.
   Every subset of a regular language is regular. FALSE. Often the easiest way to show that a universally quantified statement such as this is false by showing a counterexample. So consider L = a*. L is clearly regular, since we have just shown a regular expression for it. Now consider L = a^i: i is prime.L` ⊆L .But we showed in class that L` is not regular.

3. If L is regular, then so is L` = {xy : x ∈L and y∉ L}
   If L is regular, then so is L` = {xy : x∈L and y∉ L}. TRUE. Proof: Saying that y ∉L is equivalent to saying that y∈complement of L.Since the regular languages are closed under complement, we know that L is also regular. L is thus the concatenation of two regular

languages. The regular languages are closed under concatenation. Thus L` must be regular.

4. If L is a regular language, then so is L= {w : w∈L and w^R ∈L}.
If L is a regular language, then so is L= {w : w∈L and w^R ∈L}. TRUE. Proof: Saying that w^R ∈L is equivalent to saying that w∈L^R. If w must be in both L and L^R, that is equivalent to saying that L`= L ∩L^R. L is regular because the problem statement says so. L^R is also regular because the regular languages are closed under reversal. The regular languages are closed under intersection. So the intersection of L and L^R must be regular.

6) We know that the concatenation of two regular languages is a regular language. Consider the language $L = 0^n1^n$ over {0, 1}; L is not regular. Now consider, the language $L_1 = \{0^n\} = 0^*$ and $L_2 = \{1^n\} = 1^*$. L_1 and L_2 are obviously regular. Explain why although L_1 and L_2 are regular, L which could be seen as a concatenation of L_1 and L_2 is not regular.
Solution: Because the concatenation of the two languages $L_1.L_2$ includes any string from L_1 concatenated with any string from L_2, not just corresponding strings of equal length. The given language L is a proper subset of this language and includes only corresponding pairs. Identifying matching pairs of equal length is not possible in an infinite regular language and therefore it is not regular. This is not a violation of the closure property of regular languages under the concatenation operator.

7) What happens if we apply the Pumping Lemma to show that a formal language such as $((a + b)(a + b)(a + b))^*$ that actually regular is not regular? Explain.

Solution: We will not be able to establish a contradiction. We assume that the language is regular and in fact it is. As such, no matter which string and its partitions are chosen, pumping the string up or down always results in strings that do belong to the language. With no contradiction, we will not be in a position to conclude (wrongly, in this case) that the language is not regular.

For the given example string, if we choose $w = a^m b^m a^m$ the opponent can always choose a partition so that y contains any three symbols. Pumping y up or down will always keep the length of string divisible by 3 and there is no contradiction at all.

PES UNIVERSITY, Bangalore

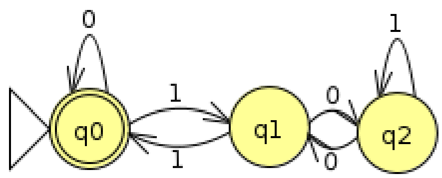
(Established under Karnataka Act No. 16 of 2013)

Department of Computer Science & Engineering

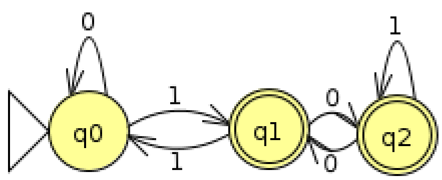
Automata Formal Languages & Logic

Using closure properties of regular languages, construct a finite automaton (NFA or DFA) for:

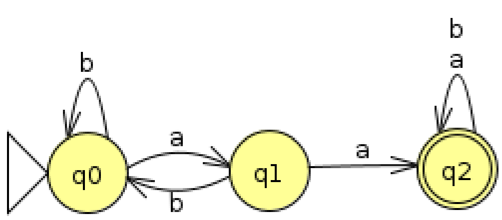
8) Binary strings which when interpreted as positive integers are not divisible by 3. Solution: We know how to construct a DFA for binary strings which when interpreted as positive integers are divisible by 3.



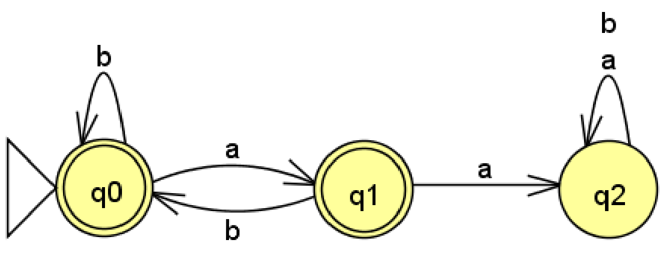
We also know that a complement of a regular language is a regular language. Therefore the complement of the language of this DFA, that is, binary strings that are not divisible by 3 is also regular. We can construct a DFA for it by simply interchanging the accepting and non-accepting states of the DFA



9)Strings over {*a*,*b*} that do not contain two consecutive *a* s. Solution: Let us construct a DFA for strings that do contain two consecutive *a* s



and then complement its language by interchanging accepting and non-accepting states:



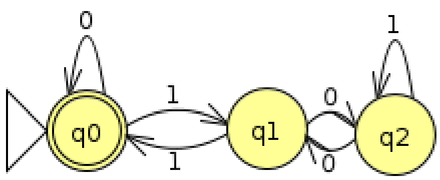
Using closure properties of regular languages, show that the following languages are regular:

10)Binary strings that do not contain the substring 101.

Solution: We know that binary strings that do contain 101 are regular since we can construct a RegEx for them: (0 + 1)*101(0 + 1)*. Therefore its complement, the set of binary strings that do not contain the substring 101 must also be regular.

11)Binary strings made up of two parts; the first part begins with a 1 and ends with a 0; the second part begins and ends with a 1.
Solution: A RegEx for the first part is:1(0 + 1)*0
A RegEx for the second part is: 1 + 1(0 + 1)*1
Both are regular. The concatenation of two regular languages is also regular. Therefore the given set of binary strings is also regular.

12)Binary strings which when reversed represent positive integers that are divisible by 3.
Solution: We know that binary strings divisible by 3 are regular . We also know that the reverse of a regular language is regular.



Therefore the given language is regular. In fact, the above automaton is fully symmetrical; reversing it does not change it in any way at all. In other words, the set of binary strings which when reversed represent positive integers that are divisible by 3 are the same as those that are divisible by 3.

13) Strings over $\{a, b, c\}$ whose length is neither an even number nor divisible by 3 or 5.

Solution: Strings whose length is even is a regular language:

$$((a + b + c)(a + b + c))^*$$

Strings whose length is divisible by 3 are also regular:

$$((a + b + c)(a + b + c)(a + b + c))^*$$

Strings whose length is divisible by 5 are also regular:

$$((a + b + c)(a + b + c)(a + b + c)(a + b + c)(a + b + c))^*$$

The complement of each of the above is also a regular language. If we take the intersection of the three complements, we get the given language which must also be regular since regular languages are closed under complementation and intersection.

14) To show that Language contains equal numbers of a and b, if we select the string w as follows, what could the adversary do in each case?

I) w = $(ab)^m$

ii) w = $a^{m/2}b^{m/2}$

ans:

I) w = xyz such that x and z are Lambda and y = $(ab)^k$, k >= m / 2. Adversary wins as we cannot pump this string out of the language. This is a bad choice of w.

ii) w = xyz such that x and z are lambda and y = $a^{m/2}b^{m/2}$. Adversary wins as we cannot pump this string out of the language. This is a bad choice of w.

15) What is the reversal of the given language L be defined by regular expression 01^*+10^*.

Solution: Let E = 01^*+10^*

$$E^R = (01^*+10^*)^R = (01^*)^R + (10^*)^R$$
$$= (1^*)^R0^R + (0^*)^R1^R$$
$$= (1^R)^*0 + (0^R)^*1$$
$$= 1^*0 + 0^*1$$

16) Is the class of languages recognized by NFAs closed under complement? Explain your answer.

Answer:The class of languages recognized by NFAs is closed under complement, which we can prove as follows. Suppose that C is a language recognized by some NFA M,i.e.,C=L(M). Since every NFA has an equivalent DFA there is a DFA D such thatL(D) =L(M) =C.We know that there is another DFA complement of D that recognizes the complement of the language L(D). Since every DFA is also an NFA, this then shows that there is an NFA, in particular complement of D,that recognizes the complement of language C=L(D) .Thus, the class of languages recognized by NFAs is closed under complement.

17) True or False: Regular expressions that do not contain the star operator can represent only finite languages.

Solution: True ; the star operator in regular expressions is the equivalent of a loop in DFAs.If a deterministic finite automaton with n states does not contain a loop,then at most it can recognise strings of length less than n. The set of such strings is finite.

18) True or False: Define *EVEN*(w) , for a finite string w , to be the string consisting of the symbols of w in even numbered positions.For example ,*EVEN*(1011010) =0011

Automata Formal Languages & Logic

If L is regular language, then {*EVEN*(w) : w∈L} must be regular.

Solution: True; given a DFA M=(Q,∑,δ,q_0F) that recognizes L, we can build an NFA M_1=(Q,∑,δ_1,q_0,F) that recognizes L_1 = {*EVEN*(w): w∈L}. Intuitively, the transition function δ_1 is set so that from every state A there are outgoing transitions to any state that is two hops away from A, where the new transition symbol M_1 is the second of the two symbols in M.

19) True or False: If L_1 and L_2 are languages such that L_2,L_1L_2 and L_2L_1 are all regular, then L_1 must be regular.

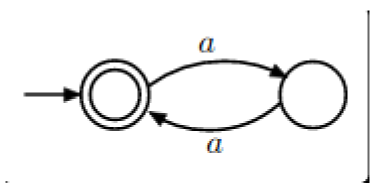
Solution:False; Consider

$$L_1 = \{0^{2^i} : i > 0\}$$

and $L_2 = \{0\}^*$.

20) Determine whether A = {a^{2n}|n>=0} is regular.

Solution : Since the regular expression $(aa)^*$ is a representation of the language A the opponent must have a winning strategy.



$(aa)^*$

Opponent : Choose n = 2(no of states in the DFA)

Me: Choose any w>=2 ,w = a^{2n}

Opponent : choose x=lambda y = a^2 and z=$a^{2(n-1)}$.

Me: choose any i>=0.

Result is that the opponent has a winning strategy for all w>=2 and i>=0 w = $(a^2)^i a^{2(n-1)}=a^{2(n+i-1)}$ belongs to A.

Automata Formal Languages & Logic

21)Let $\sum = \{a,b\}$ and let

D = {w|w contains an equal number of occurrences of the substring 01 and 10}

Thus 101 ∈ D because 101 contains a single 01 and a single 10, but 1010 ∉D because 1010 contains two 10s and only one 01. Show that D is a regular language.

Solution: This language is regular because it can be described by a RegEx and a FA

RegEx is

$(1^+0^*1^+)^*+(0^+1^*0^+)^*$

22) Determine whether each of the following languages is regular.

a) $\{a^na^na^n \mid n>0\}$
b) $\{www \mid w\in\{x,y,z\}^*, |w|<10^{100}\}$
c) $\{vw \mid v,w \in \{a,b\}^*\}$
d) $\{ww \mid w \in \{a\}^*\}$

23) Let $L = \{0^n1^n \mid n >=0\}$. Is complement of L a regular language?

24) Consider the following statement: “If A is a nonregular language and B is a language such that B⊆A, then B must be nonregular.” If the statement is true, give a proof.If it is not true, give a counterexample showing that the statement doesn’t always hold.
Answer:The statement is not always true. For example, we know that the languageA={0j1j|j≥0}is nonregular. Define the languageB={01}, and note thatB⊆A. However,Bis finite, so we know that it is regular

25) Prove that if we add a finite set of strings to a regular language, the result is regular language.
Answer:Let A be a regular language, and let B be a finite set of strings. We know that finite languages are regular, so B is regular. Thus, AUB is regular since the class of regular languages is closed under union.

26)Prove that if we remove a finite set of strings from a regular language, the result is a regular language.

Answer:Let A be a regular language, and let B be a finite set of strings with B⊆A. Let C be the language resulting from removing B from A, i.e.,C=A−B. Note that C=A−B=A∩B.Since B is regular,B is regular since the class of regular languages is closed under complement. We proved in an earlier homework that the class of regular languages is closed under intersection, so A∩B is regular since A and B are regular. Therefore,A−B is regular.

27)Consider the following statement: “If A is a non regular language andBis a language such thatB⊆A, then B must be non regular.” If the statement is true, give a proof.If it is not true, give a counterexample showing that the statement doesn't always hold.
Answer:The statement is not always true. For example, we know that the languageA={0j1j|j≥0}is nonregular. Define the languageB={01}, and note thatB⊆A. However,Bis finite, so we know that it is regular.

28)The pumping lemma says that every regular language has a pumping length p, such that every string in the language can be pumped if it has length p or more. If p is a pumping length for the language A,so is any length p` >= p.The minimum pumping length for A is the smallest p that is a pumping length for A.
For example, if A = 01^*, the minimum pumping length is 2 . The reason is that the string s=0 is in A and has length 1 yet s cannot be pumped , but any string in A of length 2 or more contain a 1 and hence can be pumped by dividing it so that x = 0 ,y=1 and z is the rest. For each of the following languages,give the minimum pumping length and justify your answer.

a) ε
b) Σ^*
c) 10(11*0)*0