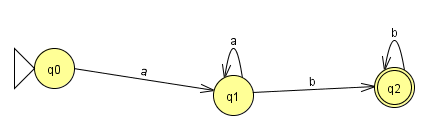
Nicholas Skinner

CS321

Fall 2019

Homework 3

1. **Prove that the following languages are not regular.**
   1. L = { 0^n : n=2^k for some k > 1}
      1. W contains L, |W|>=m w=xyz, |xy| < m |y|>= 1
      2. x y^i z contains L, |w| = 2^m >= m
      3. So w=o^(2^(m)) which contains L
      4. y will equal 0^(k) with 1 <= k <= m
      5. X y^(2) z = o^(2^(m) +k)
      6. According to the pumping lemma, this string must be in L, this tells us that 2^(m) +k = 2q
      7. 2^(m) + k < 2^(m+1)
      8. 2^(m) + k < 2^(m) + m, k<m and 2^(m) + m < 2^(m) + 2^(m) for m>1
      9. 2^(m) + k < 2^(m) + 2^(m) < 2\*2^(m) < 2^(m+1)
      10. Now we have 2^(m) + k < 2^(m+1)
      11. Our w2 is not within our L
      12. Our q value cannot be made from 2^(m) + k = 2q
      13. Thus, the language is not regular
   2. **L = { w : na(w) nb(w), w {a, b}\* }**
      1. For this one, I believe this represents the number of a’s in w must not equal the number of b’s in w.
      2. This could be rewritten to something like:
      3. {a^n b^p | n=/=p}
      4. |w| >=m, w=xyz, |xy|<m, |y|>=1
      5. w=(a^m b^(m+1))which contains L
      6. This means that m>= 0, and m+1 >=1
      7. Which can be used to determine n>=0, and p>= 1
      8. So we can have a^(m+p) b^(m+1) which does not contain our language, as it is in a state where m+p can equal m+1
      9. But this does not disprove our claim, so instead we can take the compliment of this language, which would be:
      10. {a^n b^p | n=/=p}
      11. |w| >=m, w=xyz, |xy|<m, |y|>=1, y = a^(k)
      12. W0 = a^(m-k) b^(m), which is a contradiction, as the values are unequal.
2. **Determine whether or not the following languages are regular. If the language is regular then give an NFA or regular expression for the language. Otherwise, use the pumping lemma for regular languages to prove the language is not regular.**
   1. { a^n b^n : n > 0} ∪ { a^k b^m : k > 0, m > 0}
      1. This appears to be the union of a regular and non-regular language, which depending on how it is structured, it is possible. This is regular, and would look like:

****

* 1. **L = { a^n b^m : n ≤ m ≤ 2n}**
     1. We’re going to use the pumping lemma, so we will assume that L is a regular language, and have m be the pumping lemma constant
     2. We will choose w = a^m b^m which is in L according to the definition
     3. Factor w = xyz where 0 < |xy| <= m and |y|>= 1 and xy^(i) z for all i
     4. W = a^m b^m since |xy| <= m we know that xy is all 0’s now let y = a^k where 0<k<=m
     5. We must now find an i such that w(i) = xy^(i)z should be in L by the pumping lemma
     6. Now w(2) will give us xy^(2)z = xyyz = a^(m+k) b^m since y=a^k, and by the PL: a^(m+k) b^m ∈ L.
     7. The condition m+k <=m <= 2(m+k) does not hold, so w(2) contradicts the pumping lemma proving that L is not a regular language.
  2. **L = { 0^n : n=2k for some k > 1}**

1. **Prove or disprove the following statement: If L1 and L2 are nonregular languages, then L1 L2 is also nonregular. A counterexample is sufficient to disprove the statement.** 
   1. The union of two non regular
   2. languages may or may not be a regular language.

For example: the union of {a^n b^m | n>=m} and {a^n b^m | n<=m} may themselves be irregular, but together they form the regular language L = {a\*,b\*}

1. **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 𝑆1and 𝑆2 } Show that the family of regular languages is closed under symmetric difference.**
   1. This would be closed under the symmetric difference if we assume that they are regular sets, and then use (S1 ∪ S2) ∩