Problem Set 2

CS373 - Spring 2011

Due: Thursday Feb 24 at 2:00 PM in class (151 Everitt Lab)

Please follow the homework format guidelines posted on the class web page:

`http://www.cs.uiuc.edu/class/sp11/cs373/`

1. Is it regular?

   [**Category**: Proof, **Points**: 12] Prove that each of the following languages is regular by building a finite automaton or regular expression, or prove non-regularity by using the pumping lemma.

   i) $A = \{w \in \{0,1\}^* \mid |w| \text{ is even and the number of zeros in } w\text{'s first half is even}\}$ (4 Points)

   ii) $A = \{xy \mid x, y \in \{0,1\}^*, \text{the number of 0s in } x \text{ is equal to the number of 1s in } y\}$ (4 Points)

   iii) $A = \{xy \mid x, y \in \{0,1\}^*, \text{the number of 0s in } x \text{ is equal to the number of 0s in } y\}$ (4 Points)

2. Doubled string

   [**Category**: Proof, **Points**: 10] Let $A = \{ww \mid w \in \{0,1\}^*\}$. Prove that A is non-regular using regularity preserving operations. Do not use the pumping lemma. (You may assume that $\{0^n1^n \mid n \geq 0\}$ is a non-regular language.)

3. Missing a third

   [**Category**: Proof, **Points**: 12] Prove or disprove that the following languages are regular for every regular language A.

   i) $B = \{xy \in \Sigma^* \mid \exists z, |x| = |y| = |z|, xyz \in A\}$ (6 Points)

   ii) $B = \{xz \in \Sigma^* \mid \exists y, |x| = |y| = |z|, xyz \in A\}$ (6 Points)

4. Infinite operations

   [**Category**: Proof, **Points**: 12] Consider a countably infinite collection of regular languages $A_1, A_2, A_3, \ldots$.

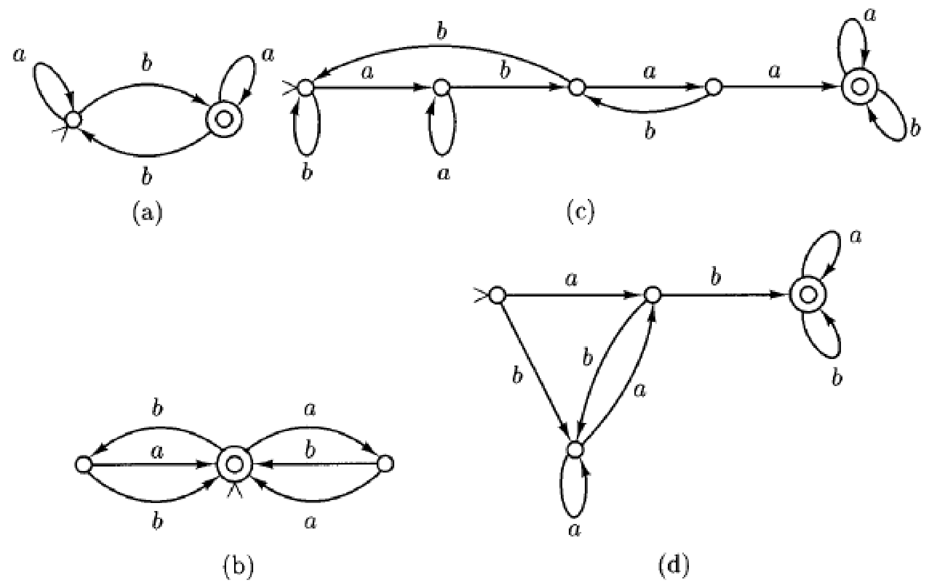
   i) Let $B_1 = \bigcup_{i=1}^{\infty} A_i$. Give an example sequence of A_i such that B_1 is regular. Then give one where B_1 is non-regular. (4 Points)

   ii) Let $B_2 = \bigcap_{i=1}^{\infty} A_i$. Give an example sequence of A_i such that B_2 is regular. Then give one where B_2 is non-regular. (4 Points)

   iii) Let $B_3 = A_1 \circ A_2 \circ A_3 \circ \ldots \circ A_i \circ \ldots$ (i.e. a countably infinite chain of concatenations). Give an example of A_i such that B_3 is regular. Then give one where B_3 is not even a subset of Σ^*. (4 Points)

   In each of the above, prove regularity / non-regularity by the usual means (construction, pumping lemma, regular closure properties, known examples from class, etc).

5. GNFA method
[**Category**: Proof, **Points**: 16] Construct regular expressions for the following four automata using the GNFA method.



6. Pair languages
[**Category**: Proof, **Points**: 18] Define a *pair language* to be a set of pairs of strings where both strings in each pair have the same length. That is, a pair language P is a subset of $\Sigma^* \times \Sigma^*$ where for $(w_1, w_2) \in P$, $|w_1| = |w_2|$. Notice that the elements of a pair language are not strings; hence, pair languages are not subsets of Σ^*. A simple example of a pair language is the pairs of all binary strings that are binary complements: $\{(w_1, w_2) \in \{0,1\}^* \times \{0,1\}^* \mid w_1 \text{ is the binary complement of } w_2\}$.

We can define a type of finite automaton that accepts a pair language. On input (w_1, w_2), the machine reads symbols from both w_1 and w_2 at the same time, making $|w_1| = |w_2|$ moves before stopping. Each state has an outgoing transition for each pair of input symbols $(a_1, a_2) \in \Sigma^2$, and otherwise it behaves exactly like a DFA. Formally, a pair DFA (PDFA) is a 5-tuple $M = (Q, \Sigma, \delta, q_0, F)$, where everything is defined as in a DFA except $\delta : Q \times \Sigma^2 \to Q$.

i) For $\Sigma = \{0,1\}$, construct a PDFA that accepts the following pair language:

$$P = \{(w_1, w_2) \in \Sigma^* \times \Sigma^* \mid \text{number of 1s in } w_1 \text{ is even iff the number of 1s in } w_2 \text{ is even}\}$$

(3 Points)

ii) Given a regular language A, prove that there is a PDFA that accepts $\{(w, w) \in \Sigma^* \times \Sigma^* \mid w \in A\}$. (3 Points)

iii) Consider a function *depair* that translates from pair languages to languages: $depair(P) = \{w_1w_2 \mid (w_1, w_2) \in P\}$. Prove that there exists a pair language P with a PDFA such that $depair(P)$ is not regular. (5 Points)

iv) Consider another function from pair languages to languages:

$$weave(P) = \{a_{1,1}a_{2,1}a_{1,2}a_{2,2}...a_{1,n}a_{2,n} \mid \exists(a_{1,1}a_{1,2}...a_{1,n}, a_{2,1}a_{2,2}...a_{2,n}) \in P \text{ for } a_{i,j} \in \Sigma\}$$

For any pair language P and the corresponding language $weave(P)$, prove that P has a PDFA if and only if $weave(P)$ is regular. (7 Points)

7. De-slip-and-slide
[**Category**: Proof, **Points**: 10] Recall that any NFA can be turned into an equivalent DFA using the subset construction method with a possibly exponential increase in the number of states. Is there a method for removing ε-transitions (aka slip-and-slides) from any NFA without changing the number of states? That is, given $N = (Q, \Sigma, \delta, q_0, F)$, can you always construct another NFA $N' = (Q', \Sigma, \delta', q_0', F')$ where $L(N) = L(N')$, $|Q| = |Q'|$, and δ' has no ε-transitions? If so, give the construction method and prove its correctness. If not, prove that there exists an NFA for which every method will fail.

8. Regular expression length
[**Category**: Proof, **Points**: 10] For some infinite sequence of symbols a_0,a_1,a_2,... from an alphabet Σ, consider the sequence of strings defined recursively by $x_1 = a_0a_0a_1$ and $x_i = x_{i-1}x_{i-1}a_i$ for $i > 1$. For example, $x_2 = x_1x_1a_2 = a_0a_0a_1a_0a_0a_1a_2$. Define $A_n = \{x_n\}$ to be the language containing only the nth string.

i) The *length* of a regular expression is the number of symbols in it, not including operators. If we only allow union, concatenation, and Kleene star in our expressions, what is the length of the smallest regular expression for A_n? Give an exact formula as a function of n in closed form. (3 Points)

ii) The intersection operator ($\cap$) can make our regular expressions more efficient. Write a regular expression with intersections for A_n whose length is $O(n^2)$. (*Hint*: Find n languages to intersect together, where each language has a regular expression of length $O(n)$.) (7 Points)