Discrete Structures	Prelim 2 sample questions
CS2800	Questions selected for spring 2017

1 Structural induction, finite automata, regular expressions

1. We define a set S of functions from $\mathbb{Z}$ to $\mathbb{Z}$ inductively as follows:

   Rule 1. For any $n \in \mathbb{Z}$, the translation (or offset) function $t_n : x \mapsto x + n$ is in S.

   Rule 2. For any $k \neq 0 \in \mathbb{Z}$, the scaling function $r_k : x \mapsto kx$ is in S.

   Rule 3. If f and g are elements of S, then the composition $f \circ g \in S$.

   Rule 4. If $f \in S$ and f has a right inverse g, then g is also in S.

   In other words, S consists of functions that translate and scale integers, and compositions and right inverses thereof.

   > **Note:** This semester, we made a bigger distinction between the elements of an inductively defined set and the meaning of an inductively defined set. We probably would have phrased this question as follows: Let S be given by
   >
   > $$s \in S ::= t_n \mid r_k \mid s_1 \circ s_2 \mid rinv\ s$$
   >
   > and inductively, let the function defined by s (written $F_s : \mathbb{Z}$) be given by the rules $F_{t_n}(x) ::= x + n$, $F_{r_k}(x) ::= ks$, $F_{s_1 \circ s_2}(x) ::= F_{s_1} \circ F_{s_2}$ and let $F_{rinv\ s} ::= g$ where g is a right inverse of F_s.

   (a) [1 point] Show that the function $f : x \mapsto 3x + 17$ is in S.

   (b) Use structural induction to prove that for all $f \in S$, f is injective. You may use without proof the fact that the composition of injective functions is injective.

   (c) Give a surjection ϕ from S to $\mathbb{Z}$ (proof of surjectivity not necessary). Remember that this surjection must map a *function* to an *integer*, and for every integer there must be a function that maps to it.

2. Seeking to avoid the limitations of regular expressions, a student designs a set S of "super expressions". Just as with regular expressions, each super expression $s \in S$ matches a set of strings, denoted by $L(s)$. The set S and the function L are defined inductively as follows:

   - A "character class" is denoted $[a_1 a_2 \ldots a_n]$ (where the a_i are single characters, i.e. elements of Σ). It matches a single character which is any of the a_i. For example, [012] matches "0", "1", and "2", but does not match "12".

     Formally, $[a_1 a_2 \ldots a_n] \in S$ and $L([a_1 a_2 \ldots a_n]) = \{a_1, a_2, \ldots, a_n\}$.

   - If $s \in S$, a "positive repetition" of s is denoted $s+$. It matches one or more strings, each of which is matched by s. For example, [45]+ matches "4", "54", and "555" but does not match ϵ.

     Formally, $s+ \in S$ and $L(s+) = \{x_1 x_2 \cdots x_n \mid n \geq 1 \text{ and } x_i \in L(s)\}$.

   - If $s_1 \in S$ and $s_2 \in S$ then the "difference" of s_1 and s_2 is denoted by $s_1 \setminus s_2$. It matches all strings that s_1 matches but s_2 does not. For example, ([123]+) \ [123] matches "1322" and "11" but not "1", "2", or "3".

     Formally, $s_1 \setminus s_2 \in S$ and $L(s_1 \setminus s_2) = L(s_1) \setminus L(s_2)$.

   - If $s_1 \in S$ and $s_2 \in S$ then the "concatenation" of s_1 and s_2 is denoted by $s_1 s_2$. It matches any string that is formed by concatenating a string matched by s_1 with a string matched by s_2. For example [012][34] matches "03" and "24" but not "0" or "32".

Formally, $s_1 s_2 \in S$ and $L(s_1 s_2) = \{xy \mid x \in L(s_1) \text{ and } y \in L(s_2)\}$.

(a) Let $\Sigma = \{0, 1, 2\}$. Let s be the super expression "([01][12])+ \ ([012][012])". Write a (normal) regular expression r such that $L(r) = L(s)$.

(b) Prove by structural induction that every super expression $s \in S$ has an equivalent regular expression. You may use without proof the fact that the union, intersection, and complement of regular languages are regular.

3. Given DFAs $M_1 = (Q_1, \Sigma, \delta_1, q_{01}, F_1)$ and $M_2 = (Q_2, \Sigma, \delta_2, q_{02}, F_2)$, we can construct a machine M_{12} with $L(M_{12}) = L(M_1) \cap L(M_2)$ as follows:

- Let $Q = Q_1 \times Q_2 =$ the set of all ordered pairs (q_1, q_2), where $q_1 \in Q_1$ and $q_2 \in Q_2$.
- Let $q_0 \in Q = (q_{01}, q_{02})$.
- Let $F = F_1 \times F_2 = \{(q_1, q_2) \mid q_1 \in F_1 \text{ and } q_2 \in F_2\}$.
- Let $\delta_{12}((q_1, q_2), a) = (\delta_1(q_1, a), \delta_2(q_2, a))$.
- Let $M_{12} = (Q, \Sigma, \delta_{12}, q_0, F)$.

Use structural induction to prove that for all $x \in \Sigma^*$, $\widehat{\delta}_{12}((q_1, q_2), x) = \left(\widehat{\delta}_1(q_1, x), \widehat{\delta}_2(q_2, x)\right)$.

4. (a) Draw a finite automaton (DFA, NFA or ϵ-NFA) with alphabet $\{0, 1\}$ to recognize the language

$$\{x \in \{0,1\}^* \mid x \text{ contains the substring } 010\}$$

(b) Draw a finite automaton (DFA, NFA or ϵ-NFA) with alphabet $\{a, b\}$ to recognize the same language as the regular expression $(ab|ba)^*$.

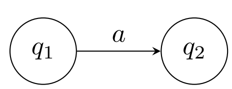
5. Prove that $L = \{0^n 1 0^n \mid n \in \mathbb{N}\}$ is not regular.

6. Let r be a regular expression. Show that there exists a regular expression r' with $L(r') = \overline{L(r)}$ (the complement of $L(r)$). If your proof involves the construction of a regular expression or automaton, you must prove that the language of the regular expression/automaton is what you claim it is (using the definitions).

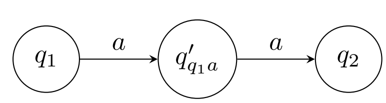
7. Build a deterministic finite automaton that recognizes the set of strings of 0's and 1's, that only contain a single 0 (and any number of 1's). Describe the set of strings that lead to each state.

8. Given a string x, we can define the "character doubling" of x to be x with every character doubled: for example $cd(abc) = aabbcc$. Formally, $cd(\epsilon) = \epsilon$, and $cd(xa) = cd(x)aa$. We can then define the "character doubling" of a language L to be the set of all strings formed by doubling the characters of strings in L; formally $cd(L) = \{cd(x) \mid x \in L\}$.

Given a DFA $M = (Q, \Sigma, \delta, q_0, F)$, we can construct a new DFA M_{cd} that recognizes $cd(L(M))$ by adding a new state q'_{qa} to the middle of every transition from q on character a:



becomes



(a) Formally describe the components $(Q_{cd}, \Sigma_{cd}, \delta_{cd}, q_{0cd}, F_{cd})$ of M_{cd} in terms of the components of M. Be sure to describe δ_{cd} on all inputs (you may need to add one or more additional states).

(b) Use structural induction on x to prove that for all x, $\widehat{\delta}(q_0, x) = \widehat{\delta}_{cd}(q_{0cd}, cd(x))$.

9. We can also define the "string doubling" of x to be xx. For example, $sd(abc) = abcabc$. Show that the set of regular languages is *not* closed under string doubling. In other words, give a regular language L and prove that $sd(L) = \{sd(x) \mid x \in L\}$ is not regular.

You can use any theorem proved in class to help prove this result.

10. Happy Cat has been shown the following proof, and has promptly turned into Grumpy Cat. Briefly but clearly identify the error which has induced grumpiness.

To prove: The language of the regular expression 0^*1^* is, in fact, not DFA-recognizable.

Proof. Let L be the language of 0^*1^*. Assume there is some DFA M with n states that recognizes L. Let $x = 0^{n-1}11$. Clearly, $x \in L$ and $|x| \geq n$. Therefore according to the Pumping Lemma, we can split x into three parts u, v and w, such that $|uv| \leq n$, $|v| \geq 1$, and $uv^iw \in L$ for all natural numbers i. Let $|v| = n$. Since $|uv| \leq n$, it must be the case that $u = \epsilon$, and $v = 0^{n-1}1$. Then $uv^2w = 0^{n-1}10^{n-1}11$, which is clearly not in L. This contradicts our assumption that there is a DFA which recognizes the language.

2 Probability problems

1. (a) Give the definition of variance in terms of expectation.

   (b) Let X and Y be random variables with $\mathrm{E}(X) = \mathrm{E}(Y) = 0$. Prove that $\mathrm{Var}(X+Y) = \mathrm{Var}(X) + \mathrm{Var}(Y)$. Make (and clearly state) additional assumptions if necessary.

2. (a) The average human height is 5 feet and 4 inches, and the variance is 2 squared inches. How large a sample must I take so that my estimate of the average will (with 90% probability) be correct to within a half inch?

   (b) A certain high school is divided into two teams: 35% of the students are "beliebers", and the remaining 65% are "directioners".

   90% of the songs on a belieber's playlist will be Justin Bieber songs, while the other 10% will be by One Direction. Directioners are a bit more broad-minded: 80% of their songs will be One Direction songs, while the remaining 20% will be Justin Bieber songs.

   A student is selected at random, and a random song is selected from their playlist. It turns out to be "Baby" by Justin Bieber. What is the probability that the student was a directioner?

3. Suppose that a coin has probability .6 of landing heads. You flip it 100 times. The coin flips are all mutually independent.

   (a) What is the expected number of heads?

   (b) What upper bound does Markov's Theorem give for the probability that the number of heads is at least 80?

   (c) What is the variance of the number of heads for a *single* toss? Calculate the variance using either of the equivalent definitions of variance.

   (d) What is the variance of the number of heads for 100 tosses? You may use the fact that if $X_1, \ldots, X_n$ are mutually independent, then $\mathrm{Var}(\sum X_i) = \sum \mathrm{Var}(X_i)$; you don't need to prove this.

   (e) What upper bound does Chebyshev's Theorem give for the probability that the number of heads is either less than 40 or greater than 80?