CS320: Review Problems for second midterm, Fall 2024

Problem 1 Let L be the set of all strings over alphabet $\{a, b, c\}$ whose first letter occurs at least once again in the string.

Write a regular expression that defines L. If such a regular expression does not exist, prove it.

Problem 2 Write a regular expression that represents the set of all strings over alphabet $\{a, b, c\}$ that contain the substring ac and the substring bc. If such a regular expression does not exist, prove it.

Problem 3 Let L be the set of strings over alphabet $\{a, b, c\}$ with at most three a's.

(a) Write a regular expression that defines L. If such regular expression does not exist, prove it.

(b) Is $\overline{L}$ (the complement of L) context-free? Explain your answer briefly.

Problem 4 Let L be the set of strings over alphabet $\{a, b, c\}$ that have even length and contain exactly one c.

(a) Write a regular expression that defines L. If such regular expression does not exist, prove it.

(b) Write a regular expression that defines $\overline{L}$ (the complement of L). If such regular expression does not exist, prove it.

Problem 5 Let L_1 be the language defined over alphabet $\Sigma = \{a, b\}$ by the regular expression:

$$(\boldsymbol{a} \cup \boldsymbol{bb})^*$$

Let L_2 be the language generated by the context-free grammar $G = (V, \Sigma, P, S)$, where $\Sigma = \{a, b\}$, $V = \{S\}$, and the production set P is:

$$S \to aSbb \mid \lambda$$

1. Write a complete formal definition of a context-free grammar G_1 that generates language L_1. If such grammar does not exist, explain why.
2. Write a complete formal definition of a context-free grammar G_2 that generates language L_2L_2. If such grammar does not exist, explain why.
3. List six different strings that belong to $L_1 - L_2$. If this is impossible, explain why.
4. List six different strings that belong to $L_2 - L_1$. If this is impossible, explain why.
5. List six different strings that belong to $L_2L_2 - L_2$. If this is impossible, explain why.

Problem 6 Let L be the language generated by the context-free grammar $G = (V, \Sigma, P, S)$, where $\Sigma = \{a, b, c\}$, $V = \{S, A, B\}$, and P is:

$$\begin{aligned}
&S \to AB \mid BA \\
&A \to ab \\
&B \to BB \mid \lambda \mid a \mid b \mid c
\end{aligned}$$

(a) Write a regular expression that defines L. If such regular expression does not exist, prove it.

(b) Is $\overline{L}$ (the complement of L) context-free? Explain your answer.

Problem 7 **(a)** Let L be the language generated by the context-free grammar $G = (V, \Sigma, P, S)$, where $\Sigma = \{a, b, c\}$, $V = \{S, B, D, E\}$, and P is:

$$\begin{aligned}
&S \to SS \mid \lambda \mid B \\
&B \to cD \mid bbD \mid aE \mid b \\
&D \to aD \mid cD \mid \lambda \\
&E \to EE \mid \lambda \mid b
\end{aligned}$$

Write a regular expression that defines L. If such a regular expression does not exist, prove it.

(b) Let $\mathcal{R}$ be the class of languages that can be represented by a regular expression, and let $\mathcal{C}$ be the class of languages that can be represented by a context-free grammar. State the cardinalities of $\mathcal{R}$ and $\mathcal{C}$, and compare them.

Problem 8 **(a)** Let L be the set of all strings over alphabet $\{a, b\}$ that have the same symbol in the first and last positions.

Write a complete formal definition of a context-free grammar that generates L. If such grammar does not exist, prove it.

(b) Let L_1 be the set of all strings of odd length over alphabet $\{a, b\}$ that have the same symbol in the first, last, and middle positions.

Write a complete formal definition of a context-free grammar that generates L_1. If such grammar does not exist, prove it.

(c) Let L_2 be the set of all strings of odd length over alphabet $\{a, b\}$ that have the same symbol in the first and middle positions.

Write a complete formal definition of a context-free grammar that generates L_2. If such grammar does not exist, prove it.

Problem 9 Let L be the set of strings over alphabet $\{a, b, c\}$ in which no two adjacent symbols are equal.

(a) Write a complete formal definition of a context-free grammar that generates L. If such grammar does not exist, prove it.

(b) Write a complete formal definition of a context-free grammar that generates $\overline{L}$ (the complement of L). If such grammar does not exist, prove it.

Problem 10 **(a)** Let:

$$L = \{a^i b^j c^k d^m \mid i = j \text{ and } j = 2k,\ i, j, k, m \geq 0\}$$

Write a complete formal definition of a context-free grammar that generates L. If such grammar does not exist, prove it.

(b) Is every countable language context-free? Explain your answer briefly.

Problem 11 **(a)** Let L_1 be a language over alphabet $\{a, b, c, d, e\}$, defined as follows:

$$L_1 = \{a^{2n} d^{\ell} b^m c^k d^{2m+1} e^{n+3} \mid k, n, m, \ell \geq 0\}$$

Write a complete formal definition of a context-free grammar G_1 that generates language L_1. If such grammar does not exist, explain why.

(b) Let L_2 be a language over alphabet $\{a, b, c, d, e\}$, consisting of those strings that have an even number of e's.

Write a complete formal definition of a context-free grammar G_2 that generates language L_2. If such grammar does not exist, explain why.

Problem 12 Let:

$$L = \{a^i b^k c^{2i+1} d^{k+2} h^{2i} \mid i, k \geq 0\}$$

(a) Write a complete formal definition of a context-free grammar that generates L. If such grammar does not exist, prove it.

(b) Is L uncountable? Explain your answer briefly.

Problem 13 Let:

$$L = \{a^{\ell} b^j c^k d^m \mid m = 2k \text{ and } k = 2\ell,\ \ell, j, k, m \geq 0\}$$

(a) Write a complete formal definition of a context-free grammar that generates L. If such grammar does not exist, prove it.

(b) Write a regular expression that defines L. If such regular expression does not exist, prove it.

Problem 14 Let L be the set of all strings over alphabet $\{a, b, c\}$ whose length is even and two middle symbols are equal.

(a) Write a complete formal definition of a context-free grammar that generates L. If such a grammar does not exist, prove it.

(b) Draw a state-transition graph of a finite automaton M that accepts L. If such an automaton does not exist, prove it.

Problem 15 Let L be the set of all strings over $\{a, b\}$ with twice as many a's as b's. Write a complete formal definition or a state-transition graph of a finite automaton M that accepts L. If such automaton does not exist, prove it.

Problem 16 Let L be the set of all strings over alphabet $\{a, b, c\}$ in which at least one of the letters appears at least twice.

(a) Write a complete formal definition of a context-free grammar G that generates L. If such a grammar does not exist, explain why.

(b) Construct a state transition graph of a finite automaton that accepts L. If such an automaton does not exist, explain why.

Problem 17 Let:

$$\Sigma = \{a, b, c\}$$

and let L be the set of all strings over Σ ending with the substring $bacb$.

(a) Construct a state-transition graph of a finite automaton M that accepts L. If such automaton does not exist, prove it.

(b) Construct a state-transition graph of a deterministic finite automaton M' that accepts L. If such automaton does not exist, prove it.

Problem 18 **(a)** Let:

$$\Sigma = \{a, b, c\}$$

and let L_1 be the set of all strings over Σ in which every a is either immediately preceded or immediately followed by b.

Construct a state-transition graph of a finite automaton M_1 that accepts L_1. If such automaton does not exist, prove it.

(b) Let:

$$\Sigma = \{a, b, c\}$$

and let L_2 be the set of all strings over Σ with an even number of a's or an odd number of b's.

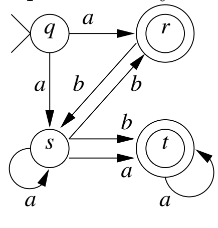
Write a regular expression that defines L_2. If such expression does not exist, prove it.

Problem 19 Let L be the language accepted by the finite automaton $M = (Q, \Sigma, \delta, q, \{f\})$, where $\Sigma = \{a\}$, $Q = \{p, q, r, s, t, v, w, x, y, z, f\}$,
and δ is given by the following table:

	a	λ
p	$\{z\}$	$\emptyset$
q	$\{t, r\}$	$\{s\}$
r	$\emptyset$	$\{q, t\}$
s	$\emptyset$	$\{w\}$
t	$\{z, y\}$	$\{p, w\}$
v	$\{x\}$	$\{r\}$
w	$\{y\}$	$\emptyset$
x	$\{p\}$	$\{v\}$
y	$\{p\}$	$\{f\}$
z	$\emptyset$	$\{v\}$
f	$\emptyset$	$\emptyset$

Compute the λ-closure of state v.

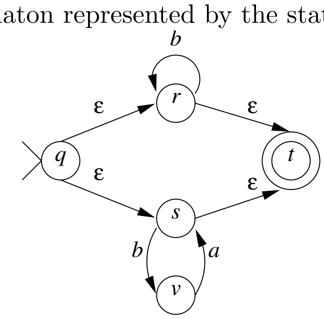
Problem 20 Let M be the finite automaton represented by the state diagram



and let L be the language accepted by M.

Write a complete formal definition or a state-transition graph of a deterministic finite automaton M' that accepts L and show your work. If such automaton does not exist, prove it.

Problem 21 Let M be the finite automaton represented by the state diagram

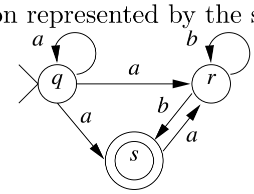


and let L be the language accepted by M. (In this diagram ϵ is used where we write λ.)

(a) Is the finite automaton M deterministic? Justify briefly your answer.

(b) If M is not deterministic, construct a deterministic finite automaton M' that accepts L and show your work. If such an automaton M' does not exist, explain why.

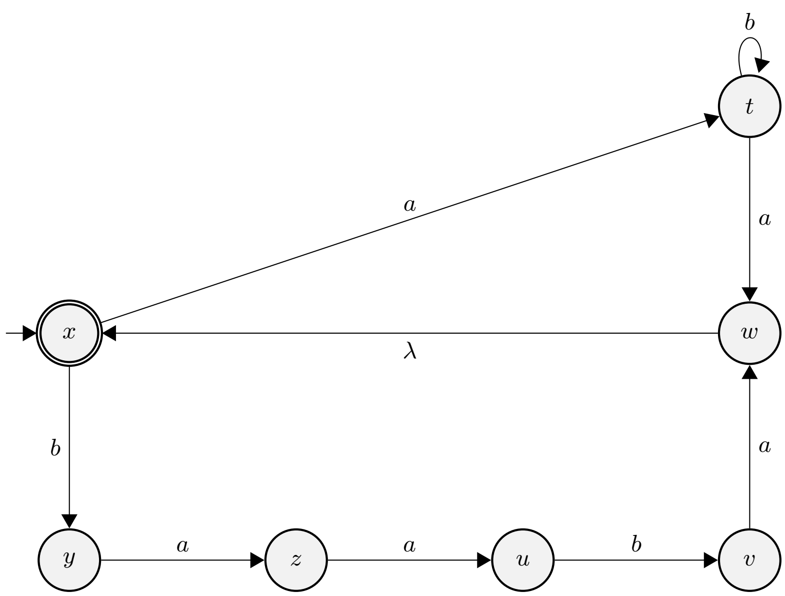
Problem 22 Let M be the finite automaton represented by the state diagram



and let L be the language accepted by M.

Construct a deterministic finite automaton M' that accepts L and show your work. If such M' does not exist, explain why.

Problem 23 Let M be the finite automaton represented by the state diagram:



and let L be the language accepted by M.

Construct a regular expression that defines L and show your work. If such regular expression does not exist, prove it.

Problem 24 Write a complete formal definition of a Turing machine:

$$M = (Q, \Sigma, \Gamma, \delta, q_0, accept, reject)$$

over input alphabet $\{0, 1\}$, such that M halts on every input, after making exactly 5 moves. If such machine does not exist, explain why.

Problem 25 Consider the Turing machine:

$$M = (Q, \Sigma, \Gamma, \delta, p, accept, reject)$$

such that: $Q = \{p, q, s, accept, reject\}$; $\Sigma = \{a, b, c\}$;
$\Gamma = \{B, X, Y, Z, a, b, c\}$.
and δ is defined by the following transition set:

$$[p, a, q, X, R]$$
$$[p, b, q, Y, R]$$
$$[p, c, q, Z, R]$$
$$[p, B, reject, B, R]$$

$$[q, a, q, X, R]$$
$$[q, b, q, Y, R]$$
$$[q, c, q, Z, R]$$
$$[q, B, s, B, L]$$

$$[s, X, accept, B, R]$$
$$[s, Y, s, B, R]$$
$$[s, Z, reject, B, R]$$
$$[s, B, s, B, R]$$

(where B is the designated blank symbol.)

Let L_1 be the set of strings *accepted* by M, let L_2 be the set of strings *rejected* by M, let L_3 be the set of strings on which M *diverges.*

(a) Draw a state transition diagram for M.

(b) What task is performed while M operates in states p and q?

(c) Write a regular expression that defines L_1. If such a regular expression does not exist, prove it.

(d) Write a regular expression that defines L_2. If such a regular expression does not exist, prove it.

(e) Write a regular expression that defines L_3. If such a regular expression does not exist, prove it.

(f) Which (if any) of the languages: L_1, L_2 and L_3 are decidable? Explain your answer.

Problem 26 Consider a Turing machine:

$$M = (Q, \Sigma, \Gamma, \delta, q_0, accept, reject)$$

such that:

$$Q = \{q_0, accept, reject\}$$
$$\Sigma = \{a, b, c\}$$
$$\Gamma = \{a, b, c, B\}$$

and δ is defined by the following transition set:

$$[q_0, a, q_0, a, R]$$
$$[q_0, b, q_0, b, R]$$
$$[q_0, c, accept, c, R]$$
$$[q_0, B, q_0, B, R]$$

(where B is the designated blank symbol.)

(a) Write a complete formal definition of a Turing machine M_1 such that M_1 accepts η if M halts on η, and M_1 rejects η if M does not halt on η, for all $\eta \in \Sigma^*$. In short:

$$(M(\eta) \searrow) \rightarrow (M_1(\eta) \searrow \text{ and accepts })$$

and also:

$$(M(\eta) \nearrow) \rightarrow (M_1(\eta) \searrow \text{ and rejects })$$

If such Turing machine M_1 does not exist, prove it.

(b) Is the language accepted by M decidable? Explain your answer.

(c) Is the language accepted by M Turing recognizable? Explain your answer.

Problem 27 Consider the Turing machine:

$$M = (Q, \Sigma, \Gamma, \delta, q, accept, reject)$$

such that:
$Q = \{q, r, s, accept, reject\}$;
$\Sigma = \{a, b\}$;
$\Gamma = \{B, a, b, X\}$;
and δ is defined by the following transition set:

$$[q, a, q, a, R]$$
$$[q, b, q, X, R]$$
$$[q, B, r, B, L]$$
$$[q, X, reject, B, L]$$

$$[r, a, r, a, L]$$
$$[r, b, reject, b, L]$$
$$[r, B, reject, B, L]$$
$$[r, X, s, X, L]$$

$$[s, a, s, a, L]$$
$$[s, b, reject, b, L]$$
$$[s, B, reject, B, L]$$
$$[s, X, accept, X, R]$$

(where B is the designated blank symbol.)
M accepts by final state.

Note that: This Turing machine terminates abnormally if it is instructed to move to the left of the left end of its tape.

(a) Write a regular expression that defines the set of strings on which M diverges. If such regular expression does not exist, prove it.

(b) Write a regular expression that defines the set of strings on which M halts and accepts. If such regular expression does not exist, prove it.

(c) Write a regular expression that defines the set of strings on which M halts and rejects. If such regular expression does not exist, prove it.

(d) Write a regular expression that defines the set of strings on which M terminates abnormally (attempts to move the head to the left of the leftmost cell.) If such regular expression does not exist, prove it.

Problem 28 Let L be the language of strings over alphabet $\{a, b\}$ that contain at least three occurrences of letter a.

(a) Write a regular expression that defines L. If such regular expression does not exist, prove it.

(b) Describe a Turing machine M that decides the following problem:

INPUT: A representation of a Turing machine M.
QUESTION: Is $L(M) = L$?

If such Turing machine does not exist, prove it.

Problem 29 Let $\Sigma = \{a, b\}$. Construct a Turing recognizable language L over Σ that is not decidable, such that its complement $\overline{L}$ is Turing recognizable but not decidable. Explain your answer. If such a language does not exist, prove it.

Problem 30 Let L_1 be a Turing recognizable language which is not decidable; and let L_2 be a decidable language.

(a) Is $L_1 - L_2$ a decidable language?
If your answer is "yes", prove it by describing an appropriate Turing machine. If your answer is "no", prove it by showing that such a Turing machine does not exist.

(b) Is $L_1 - L_2$ a Turing recognizable language?
If your answer is "yes", prove it by describing an appropriate Turing machine. If your answer is "no", prove it by showing that such a Turing machine does not exist.

(c) Is $L_2 - L_1$ a decidable language?
If your answer is "yes", prove it by describing an appropriate Turing machine. If your answer is "no", prove it by showing that such a Turing machine does not exist.

(d) Is $L_2 - L_1$ a Turing recognizable language?
If your answer is "yes", prove it by describing an appropriate Turing machine. If your answer is "no", prove it by showing that such a Turing machine does not exist.

Problem 31 Let L_1 be a Turing recognizable language, and let L_2 be a decidable language. Describe a Turing machine M that accepts $L_1 - L_2$. If such M does not exist, explain why.