You have 120 minutes to answer five questions.

Write your answers in the separate answer booklet.

Please return this question sheet and your cheat sheet with your answers.

1. Let compress0s(w) be a function that takes a string w as input, and returns the string formed by compressing every run of 0s in w by half. Specifically, every run of $2n$ 0s is compressed to length n, and every run of $2n+1$ 0s is compressed to length $n+1$. For example:

   5 3 3 2

   $$\text{compress0s}(00000110001) = 00011001$$

   4 1 2 1

   $$\text{compress0s}(11000010) = 110010$$

   $$\text{compress0s}(11111) = 11111$$

   Let L be an arbitrary regular language.

   (a) ***Prove*** that $\{w \in \Sigma^* \mid \text{compress0s}(w) \in L\}$ is regular.

   (b) ***Prove*** that $\{\text{compress0s}(w) \mid w \in L\}$ is regular.

2. Let L be the language of all strings over $\{0, 1\}$ that contain at least 374 consecutive 1s.

   (equiv. to "contains 1^{374} as substr.")

   (a) Give a regular expression that matches L.

   Use the notation R^k to denote the concatenation of k copies of the regular expression R; for example,

   $$(1+01)^5 = (1+01)(1+01)(1+01)(1+01)(1+01)$$

   (b) Describe a DFA whose language is L. *[Hint: Do not try to **draw** your DFA!]*

   (c) ***Prove*** that any DFA whose language is L must have at least 375 states, using a fooling set argument.

   $|F| \geq 375$

3. Consider the following recursive function Bond, which doubles the length of any run of 0s in its input string.

   Induction

   $$\text{Bond}(w) := \begin{cases} \varepsilon & \text{if } w = \varepsilon \\ 00 \cdot \text{Bond}(x) & \text{if } w = 0 \cdot x \text{ for some string } x \\ 1 \cdot \text{Bond}(x) & \text{if } w = 1 \cdot x \text{ for some string } x \end{cases}$$

   (a) ***Prove*** that $|\text{Bond}(w)| \geq |w|$ for all strings w.

   (b) ***Prove*** that $\text{Bond}(x \bullet y) = \text{Bond}(x) \bullet \text{Bond}(y)$ for all strings x and y.

As usual, you can assume any result proved in class, in the lecture notes, in labs, or in homework solutions.

4. Let L be the language $\{0^a 1^b 0^c \mid a = b \text{ or } a = c \text{ or } b = c\}$
   (a) ***Prove*** that L is *not* a regular language. fooling set
   (b) Describe a context-free grammar for L.

5. For each statement below, check "True" if the statement is always true and check "False" otherwise, and give a brief (one short sentence) explanation of your answer. Read these statements very carefully—small details matter!

   For any string $w \in (0+1)^*$, let w^C denote the *bitwise complement* of w, obtained by flipping every 0 in w to a 1, and vice versa. For example, $\varepsilon^C = \varepsilon$ and $000110^C = 111001$.

   (a) If $2+2=5$, then zero is odd.

   (b) $\{0^n 1 \mid n > 0\}$ is the only infinite fooling set for the language $\{0^n 1 0^n \mid n > 0\}$.

   (c) $\{0^n 1 0^n \mid n > 0\}$ is a context-free language.

   (d) The context-free grammar $S \to 00S \mid S11 \mid 01$ generates the language $\{0^n 1^n \mid n \geq 0\}$.

   (e) Every regular language is recognized by a DFA with exactly one accepting state.

   (f) Any language that can be decided by an NFA with ε-transitions can also be decided by an NFA without ε-transitions.

   (g) If L is a regular language over the alphabet $\{0,1\}$, then $\{xy^C \mid x, y \in L\}$ is also regular.

   (h) If L is a regular language over the alphabet $\{0,1\}$, then $\{ww^C \mid w \in L\}$ is also regular.

   (i) The regular expression $(00+11)^*$ represents the language of all strings over $\{0,1\}$ of even length.

   (j) Let L_1 and L_2 be two regular languages. The language $(L_1 + L_2)^*$ is also regular.

CS/ECE 374 A ✧ Spring 2026

Midterm 1 Practice 3

February 21, 2026

Name:	Navid Tajkhorshid
NetID:	navid

- ***Don't panic!*** (7:20)
- You have 120 minutes to answer five questions. The questions are described in more detail in a separate handout.
- If you brought anything except your writing implements, your **hand-written** double-sided 8½" × 11" cheat sheet, and your university ID, please put it away for the duration of the exam. In particular, please turn off and put away *all* medically unnecessary electronic devices.
- Please clearly print your name and your NetID in the boxes above.
- Please also print your name at the top of every page of the answer booklet, except this cover page. We want to make sure that if a staple falls out, we can reassemble your answer booklet. (It doesn't happen often, but it does happen.)
- Proofs or other justifications are required for full credit if and only if we explicitly ask for them, using the word ***prove*** or ***justify*** in bold italics.
- **Do not write outside the black boxes on each page**. These indicate the area of the page that our scanners will actually scan. If the scanner can't see your work, we can't grade it.
- If you run out of space for an answer, please use the overflow/scratch pages at the back of the answer booklet, but **please clearly indicate where we should look**. If we can't find your work, we can't grade it.
- **Only work that is written into the stapled answer booklet will be graded**. In particular, you are welcome to detach scratch pages from the answer booklet, but any work on those detached pages will not be graded. Please let us know if you detach a page accidentally. We will provide additional scratch paper on request.
- Please return ***all*** paper with your answer booklet: your question sheet, your cheat sheet, and all scratch paper. **Please put all loose paper *inside* your answer booklet.**

Is between
between is and between
or between between and is
or between and and or
or between or and between
or between between and and
or between or and and
or between or and or?

CS/ECE 374 A ✧ Spring 2026 Midterm 1 Practice 3 Problem 1	Name:

Consider the function compress0s defined in the question handout. Let L be an arbitrary regular language.

(a) ***Prove*** that $\{w \in \Sigma^* \mid \text{compress0s}(w) \in L\}$ is regular.

(b) ***Prove*** that $\{\text{compress0s}(w) \mid w \in L\}$ is regular.

In both parts, let $M = (Q, \delta, s, A, \Sigma)$ be a DFA accepting L.

(Every 2 0s, give 1 0 to M)

a) Let $N = (Q', \delta', s', A', \Sigma)$ be an NFA where

- $Q' = Q \times \{\varepsilon, 0\}$
- $s' = (s, \varepsilon)$
- $A' = A \times \{\varepsilon\} \cup \{(q, 0) \mid \delta(q, 0) \in A\}$

$$\left.\begin{aligned} \delta'((q,\varepsilon),1) &= \{(\delta(q,1),\varepsilon)\} \\ \delta'((q,\varepsilon),0) &= \{(q,0)\} \\ \delta'((q,0),0) &= \{(\delta(q,0),\varepsilon)\} \\ \delta'((q,0),1) &= \{((\delta(\delta(q,0),1),\varepsilon)\} = \{(\delta^*(q,01),\varepsilon)\} \end{aligned}\right. \quad \forall q \in Q$$

States (q, ε) mean we're at q in original machine & either in an even run of 0s, empty, or run of 1s. $(q, 0)$ means we're at q & in odd run of 0s. N simulates M on compress0s(w). Since N accepts it, the language is regular.

b) See page 7.

CS/ECE 374 A ✧ Spring 2026 Midterm 1 Practice 3 Problem 2	Name:

Let L be the language of all strings over $\{0, 1\}$ that contain at least 374 consecutive 1s.

(a) Give a regular expression that matches L. *[Hint: Use the notation R^k to denote the concatenation of k copies of R.]*

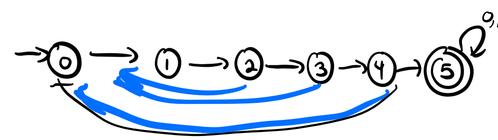
(b) Describe a DFA whose language is L. *[Hint: Do not try to **draw** your DFA!]*

(c) ***Prove*** that any DFA whose language is L must have at least 375 states, using a fooling set argument.

---

a) $(0+1)^* 1^{374} (0+1)^*$ ~~(stuff) 1^{374} (stuff)~~

b) Want to match substring 1^{374}.



Let $M = (Q, s, A, \delta, \{0,1\})$, where

- $Q = \{n \in \mathbb{Z} \mid 0 \le n \le 374\}$
- $s = 0$
- $A = \{374\}$
- $\delta(n, 0) = 0$ for all $n \le 373$
  $\delta(n, 1) = n+1$ for all $n \le 373$
  $\delta(374, c) = 374$ for all $c \in \{0,1\}$

States n for $n < 374$ mean we haven't seen a run of 374 1s, and the current run of 1s has n (or 0 if string empty or last char 0)

State 374: We have seen 374 consecutive 1s.

c) Let $F = \{1^n \mid 0 \le n \le 374\}$. Let x & y be distinct in F; WLOG $x = 1^i, y = 1^j$ for $i < j$. Let $z = 1^{374-j}$. Then $yz = 1^{374} \in L$, while $xz = 1^{374+i-j} \notin L$, since $|xz| < 374$. Thus F is a fooling set of size 375 for L, so no DFA accepting L can have less than 375 states.

CS/ECE 374 A ✧ Spring 2026 **Midterm 1 Practice 3 Problem 3**	Name:

Consider the recursive function Bond defined in the question handout.

(a) ***Prove*** that $|\text{Bond}(w)| \geq |w|$ for all strings w.

(b) ***Prove*** that $\text{Bond}(\underbrace{x}_{w} \bullet y) = \text{Bond}(x) \bullet \text{Bond}(y)$ for all strings x and y.

induct on $|x|$

$$\text{Bond}(w) = \begin{cases} \varepsilon & \text{if } w = \varepsilon \\ 00 \cdot \text{Bond}(x) & \text{if } w = 0x \\ 1 \cdot \text{Bond}(x) & \text{if } w = 1x \end{cases}$$

a) Let w be an arbitrary string. Assume for all strings x shorter than w, that $|\text{Bond}(x)| \geq |x|$.

Case 1: If $w = \varepsilon$, then $|\text{Bond}(w)| = |\varepsilon| = 0 \geq 0 = |w|$.

Case 2: If $w = 0x$ for some string x, then

$$\begin{aligned} |\text{Bond}(w)| = |00 \cdot \text{Bond}(x)| &= 2 + |\text{Bond}(x)| \\ &\geq 2 + |x| \quad \text{(IH)} \\ &\geq 1 + |x| = |0| + |x| = |0x| = w \end{aligned}$$

Case 3: If $w = 1x$ for some x, then

$$\begin{aligned} |\text{Bond}(w)| = |1 \cdot \text{Bond}(x)| &= |1| + |\text{Bond}(x)| \\ &\geq |1| + |x| \quad \text{(IH)} \\ &= |1x| = w \end{aligned}$$

So by induction, $|\text{Bond}(w)| \geq |w| \ \forall w \in \Sigma^*$.

b) See page 6.

CS/ECE 374 A ✧ Spring 2026	Name:
Midterm 1 Practice 3 Problem 4	

Let L be the language $\{0^a 1^b 0^c \mid a = b \text{ or } a = c \text{ or } b = c\}$

1. ***Prove*** that L is *not* a regular language.
2. Describe a context-free grammar for L. You do not need to justify your answer.

---

1. Let $F = \{1^n \mid n \geq 1\}$. Let $x \neq y \in F$, so $x = 1^i$ and $y = 1^j$ for some $i \neq j$. Let $z = 0^i$.
Then, $xz = 0^0 1^i 0^i \in L$, but $yz = 0^0 1^j 0^i$, which is not in L since $i, j > 0$ and $i \neq j$.
Since F is an infinite fooling set for L, L isn't regular.

$\{0^n 1^n\}$?

1110

11110

Closure sketch: $L \cap (0^*1^*) = \{0^a 1^b \mid a = 0, b = 0, \text{ or } a = b\}$
$L \cap (0^*1^*) \setminus (0^* + 1^*) = \{0^a 1^b \mid a, b > 0 \text{ and } a = b\}$.
$L \cap (0^*1^*) \setminus (0^* + 1^*) + \varepsilon = \{0^n 1^n \mid n \geq 0\}$, which isn't regular.

2. $S \to S_{ab} \mid S_{bc} \mid S_{ac}$

$S_{ab} \to AZ$ $\quad \{0^a 1^a 0^c \mid a, c \geq 0\}$

$A \to \varepsilon \mid 0A1$ $\quad \{0^a 1^a \mid a \geq 0\}$

$Z \to \varepsilon \mid 0Z$ $\quad 0^*$

$S_{bc} \to ZB$ $\quad \{0^a 1^b 0^b \mid a, b \geq 0\}$

$B \to \varepsilon \mid 1B0$ $\quad \{1^b 0^b \mid b \geq 0\}$

$S_{ac} \to D \mid 0 S_{ac} 0$ $\quad \{0^a 1^b 0^a \mid a, b \geq 0\}$

$D \to \varepsilon \mid 1D$ $\quad 1^*$

CS/ECE 374 A ✧ Spring 2026 Midterm 1 Practice 3 Problem 5	Name:

For each statement below, check "Yes" if the statement is ***always*** true and check "No" otherwise, and write a ***brief*** (one short sentence) explanation of your answer. Read these statements very carefully—small details matter!

For any string $w \in (\mathtt{0}+\mathtt{1})^*$, let w^C denote the *bitwise complement* of w, obtained by flipping every $\mathtt{0}$ in w to a $\mathtt{1}$, and vice versa. For example, $\varepsilon^C = \varepsilon$ and $\mathtt{000110}^C = \mathtt{111001}$.

(a) If $2+2=5$, then zero is odd.

☒ Yes ☐ No — $P \Rightarrow Q$ true if P is false.

(b) $\{\mathtt{0}^n\mathtt{1} \mid n>0\}$ is the only infinite fooling set for the language $\{\mathtt{0}^n\mathtt{1}\mathtt{0}^n \mid n>0\}$.

☐ Yes ☒ No — $\{0^n \mid n>0\}$ also an infinite foolng set. (or remove one element from the set)

(c) $\{\mathtt{0}^n\mathtt{1}\mathtt{0}^n \mid n > 0\}$ is a context-free language.

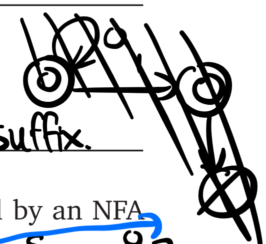
☒ Yes ☐ No — $S \rightarrow 010 \mid 0S0$ generates the language

(d) The context-free grammar $S \rightarrow \mathtt{00}S \mid S\mathtt{11} \mid \mathtt{01}$ generates the language $\{\mathtt{0}^n\mathtt{1}^n \mid n \geq 0\}$.

☐ Yes ☒ No — $S \leadsto 00S \leadsto 0001 \notin \{0^n1^n \mid n \geq 0\}$

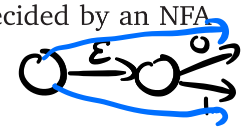
(e) Every regular language is recognized by a DFA with exactly one accepting state. ↙ true for NFA

☐ Yes ☒ No — In $\varepsilon+0$, both ε & 0 accepted, but 0 is dist. suffix.



(f) Any language that can be decided by an NFA with ε-transitions can also be decided by an NFA without ε-transitions.

☒ Yes ☐ No — Accepted by NFA $\Rightarrow$ reg $\Rightarrow$ accepted by DFA



(g) If L is a regular language over the alphabet $\{\mathtt{0},\mathtt{1}\}$, then $\{xy^C \mid x,y \in L\}$ is also regular.

☒ Yes ☐ No — Regular closed under concat & complement.

(h) If L is a regular language over the alphabet $\{\mathtt{0},\mathtt{1}\}$, then $\{ww^C \mid w \in L\}$ is also regular.

☐ Yes ☒ No — Take $L = 0^*$, $\{ww^C \mid w \in L\} = \{0^n1^n \mid n \geq 0\}$.

(i) The regular expression $(\mathtt{00}+\mathtt{11})^*$ represents the language of all strings over $\{\mathtt{0},\mathtt{1}\}$ of even length.

☐ Yes ☒ No — Cannot generate 01.

(j) Let L_1, L_2 be two regular languages. The language $(L_1+L_2)^*$ is also regular.

☒ Yes ☐ No — Regular closed under union & Kleene star.

<u>3b</u> Let x, y be any strings, and suppose for all z shorter than x, $\text{Bond}(z \cdot y) = \text{Bond}(z) \cdot \text{Bond}(y)$.

Case 1: If $x = \varepsilon$, then $\text{Bond}(xy) = \text{Bond}(y) = \varepsilon \cdot \text{Bond}(y)$
$= \text{Bond}(\varepsilon) \cdot \text{Bond}(y) = \text{Bond}(x) \cdot \text{Bond}(y)$

Case 2: If $x = 0z$ for some z,

$$\begin{aligned}
\text{Bond}(x \cdot y) &= \text{Bond}((0 \cdot z) \cdot y) \\
&= \text{Bond}(0 \cdot (z \cdot y)) && (\cdot \text{ is assoc.}) \\
&= 00 \cdot \text{Bond}(z \cdot y) && (\text{def of Bond}) \\
&= 00 \cdot \text{Bond}(z) \cdot \text{Bond}(y) && (\text{IH}) \\
&= \text{Bond}(0z) \cdot \text{Bond}(y) && (\text{def of Bond}) \\
&= \text{Bond}(x) \cdot \text{Bond}(y)
\end{aligned}$$

Case 3: If $x = 1z$ for some z,

$$\begin{aligned}
\text{Bond}(x \cdot y) &= \text{Bond}(1 \cdot (z \cdot y)) = 1 \cdot \text{Bond}(z \cdot y) \\
&= 1 \cdot \text{Bond}(z) \cdot \text{Bond}(y) = \text{Bond}(1z) \cdot \text{Bond}(y) \\
&= \text{Bond}(x) \cdot \text{Bond}(y).
\end{aligned}$$

So by induction, $\text{Bond}(x \cdot y) = \text{Bond}(x) \cdot \text{Bond}(y)$.

(scratch paper)

1b Prove $\{\text{compress0s}(w) \mid w \in L\}$ regular.

Let $N = (Q', \delta', s', A', \Sigma)$ be an NFA where

- $Q' = Q \times \{\varepsilon, \text{next1}\}$
- $s' = (s, \varepsilon)$
- $A' = A \times \{\varepsilon, \text{next1}\}$
- $\forall q \in Q \ldots$

1000

100

1 00 0

$$\delta'((q,\varepsilon),1) = \{(\delta(q,1),\varepsilon)\}$$

$$\delta'((q,\text{next1}),1) = \{(\delta(q,1),\varepsilon)\}$$

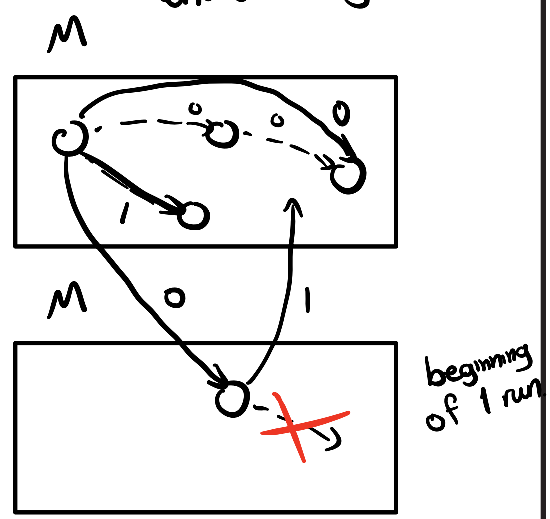
$$\delta'((q,\varepsilon),0) = \{(\delta^*(q,00),\varepsilon), (\delta(q,0),\text{next1})\}$$

$$\delta'((q,\text{next1}),0) = \emptyset$$ ← "all unspecified lead to $\emptyset$"

0010

$(0000+000)1(00+0)$

Guess where runs end... check that next char is 1 (or end of string).



State (q,ε) means we're at q in M and next char can be anything.
State $(q,\text{next1})$ means we're at q in M and next char (if any) must be a 1.
N "uncompresses" each 0 into 00, optionally uncompressing a 0 if it's at the end of a run. Since N accepts it, the language is regular.

(scratch paper)