CSE 311 : Winter 2024 Final Exam, Form A Solutions

Name:	**NetID:** @uw.edu

Instructions

- You have one-hour and fifty-minutes to complete this exam.
- You are permitted one piece of 8.5x11 inch paper with handwritten notes (notes are allowed on both sides of the paper).
- You may not use a calculator or any other electronic devices during the exam.
- We will be scanning your exams before grading them. Please write legibly, and avoid writing up to the edge of the paper.
- **Problems are printed on both the front and back of each page!**
- You may also use the last page for extra space, but tell us where to find your answer if it's not right below the problem.
- If you want us to grade something you wrote on scratch paper, put your name and netid on the paper and tell us when you turn in your exam that you have an extra sheet.

Advice

- Remember to properly format English proofs (e.g. introduce all your variables).
- All proofs for this exam must be English proofs.
- We give partial credit for the beginning and end of a proof. Even if you don't know how the middle goes, you can write the start of the proof and put the "target" and conclusion at the bottom.
- Remember to take deep breaths.

Question	**Max points**
Translation	13
Sets	12
Induction I	20
Functions	8
Induction II	20
Models of Computation	10
Uncountability/Irregularity	15
Multiple Choice	16
Grading Morale	1
Total	**115**

1. Translation [13 points]

Let your domain of discourse be mammals. Define the following predicates

- $\texttt{Cat}(x)$ returns true if and only if x is a cat.
- $\texttt{Dog}(x)$ returns true if and only if x is a dog.
- $\texttt{Equals}(x, y)$ returns true if and only if $x = y$.
- $\texttt{Human}(x)$ returns true if and only if x is a human.
- $\texttt{Loves}(x, y)$ returns true if and only if x loves y.
- $\texttt{LovesCatnip}(x)$ returns true if and only if x loves catnip.
- $\texttt{LovesToNap}(x)$ returns true if and only if x loves to nap.
- $\texttt{LovesToPlayWith}(x, y)$ returns true if and only if x loves to play with y.

(a) Translate this sentence into predicate logic notation. [3 points]

Every cat has a dog that it loves and a different dog that it does not love.

Solution:

$\forall x \exists y \exists z (\texttt{Cat}(x) \to [\texttt{Dog}(y) \land \texttt{Dog}(z) \land \texttt{Loves}(x, y) \land \texttt{Hates}(x, z) \land \neg\texttt{Equals}(y, z)])$

(b) Translate this sentence into predicate logic notation. [3 points]

There is a cat that does not love catnip.

Solution:

$\exists x (\texttt{Cat}(x) \land \neg\texttt{LovesCatnip}(x))$

(c) State **in English** the negation the following sentence, taking advantage of the domain restriction(s). (negations must be applied to single predicates). [3 points]

$$\forall x \exists y (\texttt{Cat}(x) \to [\texttt{Human}(y) \land \texttt{LovesToPlayWith}(x, y)])$$

Solution:

$\exists x \forall y \neg(\neg\texttt{Cat}(x) \lor [\texttt{Human}(y) \land \texttt{LovesToPlayWith}(x, y)])$

$\exists x \forall y (\neg\neg\texttt{Cat}(x) \land \neg[\texttt{Human}(y) \land \texttt{LovesToPlayWith}(x, y)])$

$\exists x \forall y (\neg\neg\texttt{Cat}(x) \land [\neg\texttt{Human}(y) \lor \neg\texttt{LovesToPlayWith}(x, y)])$

$\exists x \forall y (\texttt{Cat}(x) \land [\texttt{Human}(y) \to \neg\texttt{LovesToPlayWith}(x, y)])$

There is a cat that does not love to play with any human.

(d) State **in English** the contrapositive of the following statement. Your English contrapositive must take advantage of domain restriction(s). [3 points]

$$\forall x ([\neg\texttt{LovesToNap}(x) \land \neg\texttt{LovesCatnip}(x)] \to \neg\texttt{Cat}(x)])$$

Solution:

All cats love to nap or love catnip.

(e) The contrapositive of the statement in (d) has: [1 point]

- ◯ The same truth value as the original statement.
- ◯ The negation of the original statement's truth value.

Solution:

Same truth value.

2. Sets [12 points]

(a) Let A, B, C be arbitrary sets. Prove that $(B \cap C) \setminus A \subseteq B \cap (A \cup C)$ [8 points]

You must format your proof as an English proof, and structure your proof by introducing arbitrary element(s) of sets as appropriate. We recommend drawing a picture of the sets for yourself so you see why the statement is true.

Solution:

Let x be an arbitrary element of $(B \cap C) \setminus A$.

By the definition of set difference, $x \in B \cap C$ and $x \notin A$. For the first expression, the definition of set intersection tells us that $x \in B$ and $x \in C$. Since we have $x \in C$, then $x \in A \cup C$ by the definition of set union. Finally, since $x \in B$ and $x \in A \cup C$, we have $x \in B \cap (A \cup C)$ by the definition of set intersections.

Since x was arbitrary, all elements of $(B \cap C) \setminus A$ must be elements of $B \cap (A \cup C)$, so $(B \cap C) \setminus A \subseteq B \cap (A \cup C)$ by the definition of subsets.

(b) Consider the equation:

$$(A \setminus B) \setminus C = A \setminus (B \cup C)$$

Which best describes this equation? [2 points]

◯ This equation is **always** true (for all A, B, C)
◯ This equation is **sometimes** true (depending on A, B, C)
◯ This equation is **never** true (regardless of A, B, C) **Solution:**

Always true.

(c) Consider the equation:

$$(A \cap B) \setminus C = A \cap (B \cup C)$$

Which best describes this equation? [2 points]

◯ This equation is **always** true (for all A, B, C)
◯ This equation is **sometimes** true (depending on A, B, C)
◯ This equation is **never** true (regardless of A, B, C) **Solution:**

Sometimes true iff $A \cap C = \emptyset$.

3. Induction I [20 points]

Prove that $4 \mid (9^m + 3)$ for all integers $m \geq 1$.

You **must** use induction; be sure to define a predicate $P()$ and use good style. **Solution:**

Let $P(n)$:="$4 \mid 9^m + 3$". We prove $P(m)$ holds for all integers $m \geq 1$ using induction.

Base Case: $m = 1$
$9^{(1)} + 3 = 9 + 3 = 12 = 4(3)$, so $4 \mid 9^m + 3$ by definition of divides, which is $P(1)$.

Inductive Hypothesis: Suppose $P(k)$ holds for some arbitrary integer $k \geq 1$. Therefore $4 \mid 9^k + 3$, or $9^k + 3 = 4q$ for some $q \in \mathbb{Z}$.

Inductive Step:

$$\begin{aligned} 9^{k+1} + 3 &= 9 \cdot 9^k + 3 \\ &= 9(9^k + 3) - 9(3) + 3 \\ &= 9(4q) - 8(3) && \text{IH} \\ &= 4(9q - 6) \end{aligned}$$

Since $9^{k+1} + 3 = 4(9q - 6)$ and $9q - 6 \in \mathbb{Z}$, we have $4 \mid 9^m + 3$ by definition of divides.

Conclusion: We have shown $P(m)$ holds for all integers $m \geq 1$ by the principle of induction.

4. Function Fun [8 points]

Let $A = \{x \in \mathbb{Z} : 4 \mid x\}$, and let $B = \{x \in \mathbb{Z} : 2 \mid x\}$. Consider the function $f : A \to B$ defined by $f(x) = x+2$.

(a) Convince yourself that this function definition makes sense—in particular, that no matter what element of A you give as input, you'll get an element of B as output. Do **not** write anything for this part [0 points]

Solution:

If $x \in A$, then $4 \mid x$, or $x = 4k$ for some $k \in \mathbb{Z}$ by the definition of divides. As a result, $f(x) = x + 2 = 4k + 2 = 2(2k + 1)$. Since $2k + 1 \in \mathbb{Z}$, we have $2 \mid f(x)$ by the definition of divides, so $f(x) \in B$.

(b) Prove that f is **not** onto. [4 points]

Solution:

Consider $0 \in B$. If $f(x) = 0$ for some $x \in \mathbb{Z}$, then $x + 2 = 0$, or $x = -2$. However, $-2 \notin A$, so there does not exist an $x \in A$ such that $f(x) = 0$. Therefore, f is not onto.

Remark: Any concrete counter-example would work here, but some explanation (e.g. calculation of the pre-image) is needed.

(c) Prove that f is one-to-one. Be sure to use good style. [4 points]

Solution:

Assume that $f(x) = f(y)$ for some arbitrary $x, y \in A$. Then, $x + 2 = y + 2$. Subtracting 2 from both sides yields $x = y$. Therefore, since x and y are arbitrary, f is one-to-one by definition.

5. Flip it and invert it (Induction II) [20 points]

We define trees as the recursive set $\mathcal{T}$:
Basis Step: $\texttt{null} \in \mathcal{T}$.
Recursive Step: If $L \in \mathcal{T}$, $R \in \mathcal{T}$, and $x \in \mathbb{Z}$, then $\texttt{tree}(L, x, R) \in \mathcal{T}$.
Intuitively, $\texttt{tree}(L, x, R)$ is a tree with a root node (storing the value x) and subtrees L and R.

Define the following functions on trees:

$$\texttt{sum}(T) = \begin{cases} 0 & \text{if } T = \texttt{null} \\ x + \texttt{sum}(L) + \texttt{sum}(R) & \text{if } T = \texttt{tree}(L, x, R) \end{cases}$$

$$\texttt{inverse}(T) = \begin{cases} \texttt{null} & \text{if } T = \texttt{null} \\ \texttt{tree}(\texttt{inverse}(L), -x, \texttt{inverse}(R)) & \text{if } T = \texttt{tree}(L, x, R) \end{cases}$$

Prove that for all $T \in \mathcal{T}$, $\texttt{sum}(\texttt{inverse}(T)) = -\texttt{sum}(T)$.
You must use structural induction for this problem.

Solution:

Let $P(t) :=$"$\texttt{sum}(\texttt{inverse}(t)) = -\texttt{sum}(t)$". We will prove $P(t)$ holds for all $t \in \mathcal{T}$ by structural induction.

Basis Step: From the definitions,

$$\texttt{sum}(\texttt{inverse}(\texttt{null})) = \texttt{sum}(\texttt{null}) = 0$$

and

$$-\texttt{sum}(\texttt{null}) = -0 = 0$$

Since both are 0, the base case is satisfied.

Inductive Hypothesis: Suppose $P(L)$, $P(R)$ for some arbitrary trees L, R.

Inductive Step: Let $x \in \mathbb{Z}$ be arbitrary. Let $t = \texttt{tree}(L, x, R)$. We will show $P(t)$.

$$\begin{aligned} \texttt{sum}(\texttt{inverse}(t)) &= \texttt{sum}(\texttt{inverse}(\texttt{tree}(L, x, R))) \\ &= \texttt{sum}(\texttt{tree}(\texttt{inverse}(L), -x, \texttt{inverse}(R))) \\ &= -x + \texttt{sum}(\texttt{inverse}(L)) + \texttt{sum}(\texttt{inverse}(R)) \\ &= -x + -\texttt{sum}(L) + -\texttt{sum}(R) && \text{by IH} \\ &= -(x + \texttt{sum}(L) + \texttt{sum}(R)) \\ &= -(\texttt{sum}(\texttt{tree}(L, x, R))) \\ &= -\texttt{sum}(t) \end{aligned}$$

This proves $P(t)$.

Conclusion: Therefore, $P(t)$ holds for all $t \in \mathcal{T}$ by structural induction.

6. Models of Computation [10 points]

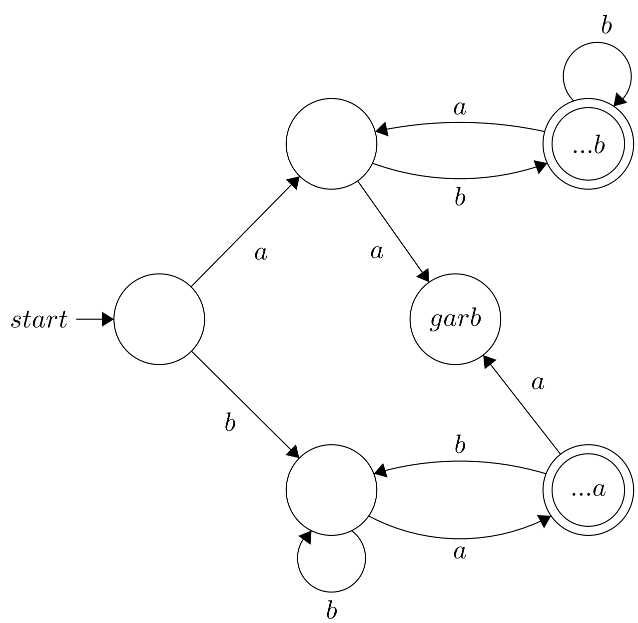
(a) Let L_1 be the following language: L_1 contains all strings x over $\{a, b\}$ such that both of the following are true

- the first and last character of x are different **and**
- x does **not** contain the substring 'aa'

For example: $abb \in L_1$, but $aab \notin L_1$ and $aba \notin L_1$. Note that you should consider ε to not be in the language (as it doesn't have a first character or last character).

Draw a **DFA** that recognizes L_1. Include a brief description of how your machine works (You'll probably want about 2-3 sentences total, or a few words for each state about what it does). [5 points]

Solution:

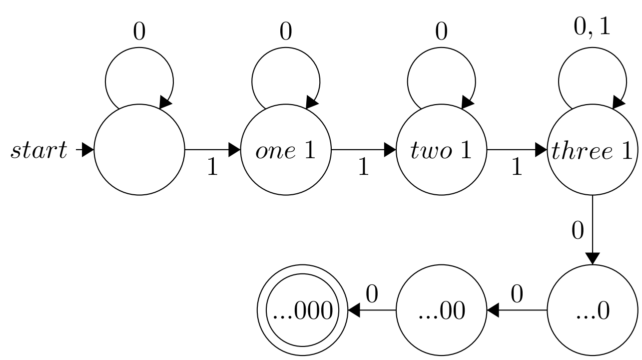


(b) Let L_2 the language containing all binary strings x such that both of the following are true

- x contains at least three 1's **and**
- x ends with 000

Draw an **NFA** that recognizes L_2. Include a brief description of how your machine works (You'll probably want about 2-3 sentences total, or a few words for each state about what it does). [5 points]

Solution:



7. Wait, that's illegal [15 points]

Choose to do exactly one of these two problems.

(a) Let L be the following language:

$$L = \{1^a 0^b 1^c : a, b, c \geq 1 \text{ and } a \geq c\}$$

Prove that L is not regular. You must use the proof technique shown in lecture. **Solution:**

> Let $L = \{1^a 0^b 1^c : a, b, c \geq 1 \text{ and } a \geq c\}$. Suppose for contradiction that a DFA D accepts L.
>
> Consider $S = \{1^n 0 : n \geq 1\}$. Since S contains infinitely many strings and D has a finite number of states, two strings in S must end up in the same state.
>
> Say these strings are $a = 1^i 0$ and $b = 1^j 0$ for some $i, j \geq 1$. Without loss of generality, we know that $i > j$.
>
> Append the string $c = \#1^i$ to both of these strings. The two resulting strings are:
>
> $ac = 1^i 0 1^i$ Note that $ac \in L$ since $i \geq i$.
>
> $bc = 1^j 0 1^i$ Note that $bc \notin L$, since $i > j$.
>
> Since ac and bc end up in the same state, but $ac \in L$ and $bc \notin L$, that state must be both an accept and reject state, which is a contradiction. Thus there is no DFA that recognizes L, so L is not regular.

(b) Let $\Sigma = \{5, 6, 7\}$, call a function "Σ-valued" if $f : \mathbb{N} \to \Sigma$ (i.e., it's a function that takes in a natural number and outputs $5, 6$, or 7).

Prove, by recreating a diagonalization argument, that the set of all Σ-valued functions is uncountable. **Solution:**

> **Updated solution**
>
> Let S be the set of all Σ-valued functions. Suppose for the sake of contradiction that S is countable. Then there exists a surjection $\Phi : \mathbb{N} \to S$. In other words, we can associate each Σ-valued function f_n with a natural number n such that $\Phi(n) = f_n$. (Note that there may be multiple natural numbers that map to f_n under Φ.)
>
> We construct a new function f as follows: for all $n \in \mathbb{N}$,
>
> $$f(n) = \begin{cases} 6 & \text{if } f_n(n) = 5 \\ 7 & \text{if } f_n(n) = 6 \\ 5 & \text{if } f_n(n) = 7. \end{cases}$$
>
> Notice that f differs from each f_n on the input $n \in \mathbb{N}$, so S cannot be in the (infinite) list of Σ-valued functions, which contradicts our assumption that such a list exists. Therefore, S is uncountable.

I am doing

◯ part (a)

◯ part (b)

8. True/False [16 points]

Choose the (one) best answer for each question below. No justification is required, and no partial credit will be given. Each question is worth 2 points.

(a) $p \leftrightarrow q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$

◯ True

◯ False **Solution:**

True

(b) Let $P(x, y)$ be a predicate. Then $\exists y \forall x P(x, y)$ implies $\forall x \exists y P(x, y)$.

◯ True

◯ False **Solution:**

True, the first statement is "stronger" than the second (that is, the first implies the second). For the second, y is allowed to depend on x. For the first, one specific y must work for all x. Thus if the first is true, whatever value of y makes it true, can also be plugged in for y in the second statement for every x. On the other hand, if the second statement is true, it might be that different y's work for the different x's

(c) Let a, b, c, d, n be integers with $c, d, n > 0$. If $a \equiv b \pmod{n}$ and $c \equiv d \pmod{n}$, then $a^c \equiv b^d \pmod{n}$.

◯ True (for all a, b, c, d, n)

◯ False (for at least one choice of a, b, c, d, n) **Solution:**

False. $2 \equiv 5 \mod 3$, and $4 \equiv 1 \mod 3$, but $2^4 = 16 \equiv 1 \mod 3$ and $5^1 = 5 \equiv 2 \mod 3$.

(d) There exists a language that can be recognized by a NFA but cannot be recognized by any DFA.

◯ True

◯ False **Solution:**

False, every NFA can be written as a DFA.

(e) For every regular language L, there is a **unique** DFA that accepts L (and no other strings).

◯ True

◯ False **Solution:**

False

(f) Let L_1 and L_2 be languages over the same alphabet Σ. If L_1 and L_2 are both regular, then $\Sigma^* \setminus (L_1 \cup L_2)$ is

◯ never regular

◯ sometimes regular

◯ always regular **Solution:**

Check! Always regular. $L_1 \cup L_2$ is regular since we can represent it using an NFA with ϵ-transitions to NFAs accepting both L_1 and L_2. The complement of a regular language is regular since we can make an accepting NFA by flipping the accept/reject states.

(g) We need a definition for this question: in a finite automaton (DFA or NFA), we use the term "garbage state" to describe a non-accepting state that has a self-loop for every character in the language (and no other existing transition). Note that if we're in a garbage state, no matter what characters are left to process, the automaton will stay in the garbage state.

Now, to the statement to evaluate: Given an NFA that has a garbage state, if we remove that state and any transitions to it, then the new NFA might not describe the same language.
The statement to evaluate is

◯ True

◯ False **Solution:**

False. Having a garbage state does not change the strings accepted by an NFA.

(h) Let $f : \mathbb{Z} \to A$ be given by the formula $f(x) = x^2$. Depending on the set A, f is

◯ **never** onto

◯ **sometimes** onto

◯ **always** onto **Solution:**

Sometimes onto. If A is the set of perfect squares it is onto; if A is $\mathbb{Z}$, it is not onto.

9. Grading Morale [1 point]

Put something on this page. Depending on your mood, this might be a poem, the statement of your favorite theorem, the statement of your least favorite theorem, or a piece of art.

As long as you make some mark on this page, you will get the point.

Looking at these helps keep the TAs happy while grading.

Additional space for any prior problem. Be sure to tell us to look here on the original problem!