

Practice Midterm Exam 3

This exam, which was the actual midterm given in Spring 2017, is closed-book and closed-computer. You may have a double-sided, 8.5" × 11" sheet of notes with you when you take this exam. You may not have any other notes with you during the exam. You may not use any electronic devices (laptops, cell phones, etc.) during the course of this exam. Please write all of your solutions on this physical copy of the exam.

You are welcome to cite results from the problem sets or lectures on this exam. Just tell us what you're citing and where you're citing it from. However, please do not cite results that are beyond the scope of what we've covered in CS103.

On the actual exam, there'd be space here for you to write your name and sign a statement saying you abide by the Honor Code. We're not collecting or grading this exam (though you're welcome to step outside and chat with us about it when you're done!) and this exam doesn't provide any extra credit, so we've opted to skip that boilerplate.

You have three hours to complete this practice midterm. There are 32 total points. This practice midterm is purely optional and will not directly impact your grade in CS103, but we hope that you find it to be a useful way to prepare for the exam. You may find it useful to read through all the questions to get a sense of what this practice midterm contains before you begin.

Question	Points	Grader
(1) Set Theory	/ 8	
(2) Mathematical Logic	/ 8	
(3) Proofwriting I	/ 8	
(4) Proofwriting II	/ 8	
	/ 32	

You can do this. Best of luck on the exam!

Problem One: Set Theory**(8 Points)**

Consider the following two sets:

$$\mathbb{E} = \{ n \in \mathbb{N} \mid n \text{ is even} \}$$

$$\mathbb{O} = \{ n \in \mathbb{N} \mid n \text{ is odd} \}$$

Remember that 0 is a natural number.

Answer each of the following questions *without* using set-builder notation and *without* using any of the set operators (union, intersection, etc.). **Briefly** (with at most one sentence) justify each of your answers. No proofs are required.

- i. **(1 Point)** What is $\mathbb{E} \cup \mathbb{O}$?

- ii. **(1 Point)** What is $\mathbb{E} \cap \mathbb{O}$?

- iii. **(1 Point)** What is $\mathbb{E} - \mathbb{O}$?

- iv. **(1 Point)** What is $\mathbb{E} \Delta \mathbb{O}$?

- v. **(2 Points)** What is $\{ k \mid \exists m \in \mathbb{E}. \exists n \in \mathbb{O}. k = m + n \}$?

- vi. **(2 Points)** What is $\{ k \mid \exists m \in \mathbb{E}. \exists n \in \mathbb{O}. k = mn \}$?

Problem Two: Mathematical Logic**(8 Points)**

When we discussed first-order logic, we spent a decent amount of time talking about how to translate statements from English into first-order logic. You practiced this skill on Problem Set Two. In this problem, we'd like you to show us what you've learned about the art of first-order translation.

Arthur C. Clarke's short story *The Nine Billion Names of God* is a piece of speculative fiction. In the story, a team of computer scientists is commissioned to write a program that will print out every possible name. The people who commission this program claim, for various reasons, that when the program finishes running, the universe will come to a close. The following scene transpires as the (skeptical) computer scientists anticipate that the program has nearly completed:

“Wonder if the computer's finished its run. It was due about now.”

Chuck didn't reply, so George swung round in his saddle. He could just see Chuck's face, a white oval turned toward the sky.

“Look,” whispered Chuck, and George lifted his eyes to heaven. (There is always a last time for everything.)

Overhead, without any fuss, the stars were going out.

Your job is to translate the crux of this story into first-order logic.

- i. **(5 Points)** Given the predicates

Program(x), which states that x is a program;

Name(x), which states that x is a name;

Prints(x, y), which states that x prints y ;

Star(x), which states that x is a star; and

WillGoOut(x), which states that x will go out,

write a statement in first-order logic that says “if there is a program that prints out every name, then all the stars will go out.”

We've talked a lot about negating and simplifying statements in first-order logic, a useful skill with applications to disproofs, proofs by contradiction, and proofs by contrapositive. You practiced this on Problem Set Two, and in this question we'd like you to demonstrate what you've learned.

ii. (3 Points) Consider the following statement in first-order logic:

$$\begin{aligned} &\forall x. (\text{Person}(x) \wedge (\exists y. \text{FriendOf}(x, y)) \rightarrow \\ &\quad \exists z. (\text{Person}(z) \wedge \\ &\quad \quad \forall w. (\text{WorkTogether}(x, z, w) \leftrightarrow \\ &\quad \quad \quad \exists u. \text{AllReportTo}(x, z, w, u) \\ &\quad \quad \quad) \\ &\quad \quad) \\ &\quad) \end{aligned}$$

Give a statement in first-order logic that is the negation of this statement. As in Problem Set Two, your final formula must not have any negations in it, except for direct negations of predicates. Do not introduce any new predicates, functions, or constants.

While you're not required to show every step of your work, it does make it a bit easier for us to give you partial credit in the event that you accidentally make a mistake. If you do show each step, please clearly indicate what your final answer is.

Problem Three: Proofwriting I**(8 Points)**

On Problem Set One and Problem Set Two, you gained experience writing rigorous mathematical proofs about sets and set theory. In this problem, we're going to ask you to prove two results about sets. The results here are actually quite useful and show up in later CS courses, like CS243 (where they're used in the design of optimizing compilers) and CS258 (where they're used to mathematically model computer programs).

- i. **(4 Points)** Prove that for any set S , if $A \in \wp(S)$ and $B \in \wp(S)$, then $A \cup B \in \wp(S)$.

In the course of writing up your proof, please call back to the formal definitions of the relevant set relations and operations.

The following interaction is often used in conjunction with the previous result.

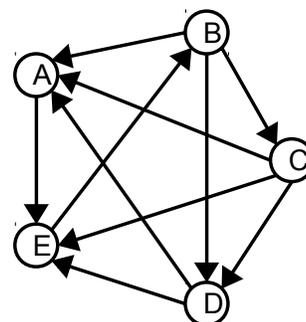
ii. **(4 Points)** Prove that for any sets A and B , if $A \cup B \subseteq B$, then $A \subseteq B$.

In the course of writing up your proof, please call back to the formal definitions of the relevant set relations and operations.

Problem Four: Proofwriting II**(8 Points)**

On Problem Set Two, you explored tournaments, contests between groups of n players. This problem explores another property of tournaments.

Let's begin by refreshing some definitions. A *tournament* is a contest among n players. Each player plays a game against each other player, and either wins or loses the game (let's assume that there are no draws). We can visually represent a tournament by drawing a circle for each player and drawing arrows between pairs of players to indicate who won each game. For example, in the tournament to the right, player A beat player E , but lost to players B , C , and D .



Now, let's introduce a new definition. A tournament is called a *transitive tournament* if for any players x , y , and z in the tournament, if x won her game against y and y won his game against z , then x won her game against z as well. (The tournament to the right is *not* transitive – do you see why?)

Prove that if T is a transitive tournament, then no two players in T can have won exactly the same number of games.

(Extra space for your answer to Problem Four, if you need it.)