Practice Midterm Exam

This practice exam is closed-book and closed-computer but open-note. You may have a double-sided, 8.5” × 11” sheet of notes with you when you take this exam. Please hand-write all of your solutions on this physical copy of the exam.

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 24 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	(6)	/ 6	
(2) Graph Theory	(6)	/ 6	
(3) Induction	(6)	/ 6	
(4) Mathematical Logic	(6)	/ 6	
	(24)	**/ 24**	

Best of luck on the exam!

Problem One: Set Theory (6 Points)

In this question, we're going to introduce a special type of set called a ***hereditary set*** and then ask you to work with that definition.

Let's begin with the definition of hereditary sets:

A set S is a ***hereditary set*** if all its elements are hereditary sets.

This definition might seem strange because it's self-referential – it defines the hereditary sets in terms of other hereditary sets! However, it turns out that this is a perfectly reasonable definition to work with, and in this problem you'll explore properties of these types of sets.

i. **(1 Point)** Given the self-referential nature of the definition of hereditary sets, it's not even clear that hereditary sets even exist. As a starting point, prove that there is at least one hereditary set. *(Hint: Think about vacuous truths.)*

Here's the definition of hereditary sets from the previous page:

A set S is a ***hereditary set*** if all its elements are hereditary sets.

It's possible to use some of the standard set operations to transform hereditary sets into new hereditary sets. This question explores one example of this.

ii. **(5 Points)** Prove that if S is a hereditary set, then $\wp(S)$ is also a hereditary set.

Problem Two: Graph Theory (6 Points)

In this question, we're going to return to tournament graphs and tournament winners (which you first explored in Problem Set Two) and see some of their other strange properties.

Here's a quick refresher of some definitions you'll need in this problem. A ***tournament*** is a contest between some number players in which each player plays each other player exactly once. We assume that no games end in a tie, so each game ends in a win for one of the players. A ***tournament winner*** in a tournament is a player w where, for each other player p, either player w beat player p, or player w beat some other player who in turn beat player p.

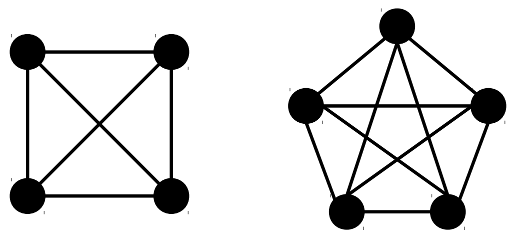
Now, let's introduce a new definition. We'll say that a tournament G is an ***egalitarian tournament*** if G has an odd number of players (say, $2n + 1$ players) and each player won exactly n games. In other words, G is egalitarian if it contains an odd number of players and each player won exactly half the games she played.

Prove that if G is an egalitarian tournament, then every player in G is a tournament winner.

Problem Three: Induction (6 Points)

In this problem, you'll see a new type of graph called the ***k-clique***, then will prove a useful property of k-cliques with applications to social network analysis.

A ***k-clique*** is a graph with k nodes where each node is connected to the k-1 other nodes in the graph. For example, here's a 4-clique and a 5-clique:



Now, suppose that you take a k-clique and color each edge either red or blue. Prove the following result by induction: if the k-clique contains an odd-length cycle made only of blue edges, then it must contain a cycle of length three with an odd number of blue edges (that is, a cycle of length three with exactly one blue edge or exactly three blue edges.) This result might seem pretty strange, but trust me, it's meaningful. We'll put details in the solution set. ☺

As a hint, try doing induction on the length of the cycle rather than the number of nodes in the graph.

(Extra space for Problem Three, if you need it.)

Problem Four: Mathematical Logic (6 Points)

There's a close connection between mathematical logic and set theory. The first part of this question explores this connection.

For any sets A and B, consider the set S defined below:

$$S = \{ x \mid \neg(x \in A \to x \in B) \}$$

i. **(2 Points)** Write an expression for S in terms of A and B using the standard set operators (union, intersection, etc.), but without using set-builder notation. Briefly justify why your answer is correct.

Two candidates X and Y are running for office and are counting final votes. Candidate X argues that more people voted for him than for Candidate Y by making the following claim: "For every ballot cast for Candidate Y, there were two ballots cast for Candidate X." Candidate X states this in first-order logic as follows:

$$\forall b.\ (BallotForY(b) \rightarrow \exists b_1.\ \exists b_2.\ (BallotForX(b_1) \wedge BallotForX(b_2) \wedge b_1 \neq b_2))$$

However, it is possible for the above first-order logic statement to be true even if Candidate X didn't get the majority of the votes.

ii. **(4 Points)** Give an example of a set of ballots such that

1. every ballot is cast for exactly one of Candidate X and Candidate Y,
2. the set of ballots obeys the rules described by the above statement in first-order logic, but
3. candidate Y gets strictly more votes than Candidate X.

You should justify why your set of ballots works, though you don't need to formally prove it. Make specific reference to the first-order logic statement in your justification.