CSc 144–002 — Discrete Mathematics for Computer Science I — Fall 2024 (McCann)
https://cs.arizona.edu/classes/cs144/fall24-002/

Homework #4

(50 points)

Due Date: October 18th, 2024, at the beginning of class

Directions

1. **This is an INDIVIDUAL assignment; do your own work! Submitting answers created by computers or by other people is NOT doing your own work.**
2. Start early! Getting help is much easier n days before the due date/time than it will be n hours before. Help is available from the class staff via piazza.com and our office hours.
3. Write complete answers to each of the following questions, in accordance with the given directions. Create your solutions as a PDF document such that each answer is clearly separated from neighboring answers, to help the TAs easily read them. Show your work, when appropriate, for possible partial credit.
4. When your PDF is ready to be turned in, do so on gradescope.com. Be sure to assign pages to problems after you upload your PDF. Need help? See "Submitting an Assignment" on https://help.gradescope.com/.
5. **Solutions submitted more than five minutes late will cost you a late day. Submissions more than 24 hours late are worth no points.**

Topic: Arguments

1. (8 points) Each of the following short arguments is a fallacy. For each, give two "real world" examples: With the first example, show that it is possible that the conclusion is true when the hypotheses are true, With your second example, show that it is possible that the conclusion is false even though the hypotheses are true. Be aware that these fallacies may or may not be fallacy types presented in class.

   (a) Hypotheses: $\neg c$. $c \to d$.
   Conclusion: $\neg d$.

   (b) Hypotheses: $\exists x\,(A(x) \wedge B(x))$. $\exists x\,(B(x) \wedge C(x))$.
   Conclusion: $\exists x\,(A(x) \wedge C(x))$.

Topic: Direct Proofs

2. (6 points) Prove, using a direct proof, that the difference of a multiple of nine and a multiple of fifteen is a multiple of three. (*Hint:* The even numbers are multiples of two; you know how to represent those in a proof. Extend that idea to these multiples.)
3. (6 points) Prove, using a direct proof, that negating a value that is one larger than a multiple of five produces a value that is one smaller than a multiple of five.
4. (6 points) Prove, using a direct proof, that the square of the sum of two rational numbers is also rational.
5. (6 points) Prove, using a direct proof, that $(a+b)^2 \geq a^2 + b^2,\ a, b \in \mathbb{Z}^*$.

Topic: Additional Set Concepts

6. (2 points) What is the result of $\mathcal{P}(\{w, x, y, z\}) - \mathcal{P}(\{x, y, z\})$?
7. (2 points) The Bell numbers tell us the number of partitions that can be formed from a set, assuming that the set itself is one of the partitions. For a set of three values, there are 5 possible partitions. What are the five partitions of $\{j, k, l\}$?

(Continued on the back . . .)

Topic: Prolog

Background: On D2L is a video (in the Content – Assignments – Homework #4 area) that explains how to access our main instructional machine (`lectura.cs.arizona.edu`), and demonstrates how to run and use a version of Prolog (`gprolog`), including how to create a text file containing a Prolog database, load a Prolog database into `gprolog`, and ask queries. On the class web page is a five–page tutorial document titled "Quick–'n'–Dirty Prolog Tutorial" to help you understand Prolog.

The first thing you need to do is find the email you received when you started taking CS classes. It was sent from `it-admin@cs.arizona.edu` with the subject line of "Welcome of the Department of Computer Science." It contains information about your CS account. If you can't find that email, or changed your CS password at that time and forgot it, you can reset it by visiting `https://helpdesk.cs.arizona.edu` and clicking on the red link that reads "Click here to self manage your CS account/password." When you have your account ready to go, have read the tutorial, and have watched that video, you should be ready for the following questions. Start early so that you can get help if you need it!

8. (9 points) It is playoff time for Major League Baseball (MLB)! For this question, we will focus on the National League (NL) and its three divisions of five team each (which we'll abbreviate to nleast, nlcentral, and nlwest). Each team started playing under their current name and city in a specific year, and during the 2024 season, each won a specific quantity of games.

   Create a Prolog database that contains three collections of facts and two rules, all based on the above setting. The first set of facts has facts of the form `firstyear(team,year).` For example, `firstyear(brewers,1970).` says that the Milwaukee Brewers started playing as the Brewers in Milwaukee in 1970. The second set of facts has the form `division(team,divabbrev).` (e.g., `division(brewers,nlcentral).`) to show that the Brewers are in the NL Central, and the third set has facts of the form `wins2024(team,total).` `wins2024(brewers,93).` says that the Brewers won 93 regular–season games in 2024. You'll have fifteen of each fact. To get you started, on the class web page is a page with the fifteen `firstyear()` facts. You'll have to create the other two sets on your own (search Google for "mlb standings 2024" to get the data). There are also two rules to write: `divisionyear(X,Y).` and `divisionwins(X,Y).` The first tests to see if a division abbreviation X has any teams that started playing in year Y, and the second checks if division abbreviation X has any teams with Y wins in 2024. For example, the query `divisionwins(nlwest,90).` will report "no," because no NL West teams won exactly 90 games.

   When your database is created on lectura, load it into Prolog and use it to answer these four queries:

   (a) `wins2024(diamondbacks,89).` (In English: Did the Diamondbacks win 89 games in 2024?)
   (b) `firstyear(cubs,1900).` (In English: Did the Cubs starting playing in Chicago in 1900?)
   (c) `divisionyear(nlcentral,1891).` (In English: Did any NL Central teams starting playing in 1891?)
   (d) `divisionwins(X,89).` (In English: In which divisions did a team win 89 games?)
   (For the last two queries, remember to press the semicolon key (`;`) when you see an answer followed by a question mark, to allow Prolog to find and display additional answers, if any.)

   As your answer to this question in your PDF, include screenshot(s) that show: (a) The facts and rules in your database, and (b) the results gprolog produced when you ran the four queries.

9. (5 points) In class I showed you the "Has Hugo had a programming class?" example. We completed the argument ourselves, but Prolog can answer queries like that for us. Your task: Get Prolog to answer this question: Anyone who can ride a motorcycle can ride a bicycle. Hugo can ride a motorcycle. Can Hugo ride a bicycle?

   You'll need one fact (named `ridesmotorcycle(Hugo)`, stating that Hugo can ride one, and one rule (explaining to Prolog that $\forall x(\text{ridesmotorcycle}(x) \rightarrow \text{ridesbicycle}(x))$ means "if anyone can ride a motorcycle, then they can ride a bicycle"). When this very small database is ready, use it to answer the above question: Can Hugo ride a bicycle?

   As your answer to this question in your PDF, include screenshot(s) that show: (a) The fact and the rule in your database, and (b) the result gprolog produced when you ran the query to answer the question.