

IIT CS536: Science of Programming

Homework 1: Course Basics, Logic

Prof. Stefan Muller

Out: Wednesday, Aug. 23

Due: Thursday, Sept. 7, 11:59pm CDT

Updated Aug. 30

This assignment contains 7 written task(s) and 3 task(s) for which you'll submit logic proofs from the online tool, for a total of 60 points.

Logistics

Submission Instructions

Please read and follow these instructions carefully.

- Submit your homework on Blackboard under the correct assignment by the deadline (or the extended deadline if taking late days).
- You may submit multiple times, but we will only look at your last submission. Make sure your last submission contains all necessary files.
- Email the instructor and TAs ASAP if
 - You submit before the deadline but then decide to take (more) late days.
 - You accidentally resubmit after the deadline, but did not intend to take late days.

Otherwise, you do not need to let us know if you're using late days; we'll count them based on the date of your last submission.

- Submit your written answers in a single PDF or Word document. Typed answers are preferred (You can use any program as long as you can export a .pdf, .doc or .docx; LaTeX is especially good for typesetting logic and math, and well worth the time to learn it), but *legible* handwritten and scanned answers are acceptable as well.
- Your Blackboard submission should contain exactly the file with your written answers and all of the .log files with your proofs (with the names given in each problem); *do not rename these files*. Do not compress or put any files in folders.

Collaboration and Academic Honesty

Read the policy on the website and be sure you understand it.

1 Course Basics

Task 1.1 (Written, 10 points).

For each statement, say whether it is **true or false** and **explain in 1-2 sentences**.

- a) Testing provides more of a guarantee that a program is correct than verification, because you actually run the program.
- b) It is useful to do both testing and verification on a program before deploying it.

Task 1.2 (Written, 10 points).

For each pair of propositions, say whether they are

- (I) syntactically equal (\equiv) (and therefore also semantically equal)
- (II) semantically equal ($=$) but not syntactically equal, or
- (III) neither

- | | | |
|----|---|-----------------------------|
| a) | $p \wedge q \vee r$ | $(p \wedge q) \vee r$ |
| b) | $p \rightarrow q$ | $\neg p \rightarrow \neg q$ |
| c) | $(p \rightarrow q) \leftrightarrow (\neg q \rightarrow \neg p)$ | T |
| d) | $\neg p$ | $p \rightarrow q$ |
| e) | $p \wedge q \vee r$ | $p \wedge (q \vee r)$ |

2 Propositional Logic

Task 2.1 (Written, 5 points).

For the following proposition, **construct a truth table** and **say whether it is** a tautology, contradiction or contingency. Explain your answer briefly.

$$(P \rightarrow Q) \rightarrow (\neg P \rightarrow \neg Q)$$

Task 2.2 (Programming, 5 points).

Prove that $P \rightarrow (Q \vee P)$ is a tautology by proving $T \Rightarrow P \rightarrow (Q \vee P)$.

Important: Submit your proof (and only your proof) in the file `tauto.log`. It should be in a format checkable by the proof checker at <http://www.cs.iit.edu/~smuller/cs536-f23/logic/proofs.html>. That is, *you should be able to copy and paste the entire contents of your file into the text box of the proof checker, click “Check” and have it respond that the proof is correct.* To help you, you can select “HW1 Task 2” from the dropdown box on the proof checker page to load the editor with the statement above, and work on your proof there. When you’re done, click “Download” to download the finished proof.

Task 2.3 (Written, 2 points).

As we saw in the previous task, proving $T \Rightarrow P$ shows that P is a tautology. What does it show if we instead prove $P \Rightarrow T$?

Task 2.4 (Programming, 5 points).

Prove the following implication. **Submit your proof in the file `uncurry.log`, following the same instructions as in Task 2.**

$$P \rightarrow (Q \rightarrow R) \Rightarrow (P \wedge Q) \rightarrow R$$

(Functional programmers may recognize this as *uncurrying*.)

3 Predicate Logic

Task 3.1 (Written, 4 points).

Let \mathbb{Z} be the set of integers and let M be the set of prime numbers (clearly $M \subset \mathbb{Z}$). Goldbach’s Conjecture states that every even integer greater than 2 is the sum of two prime numbers. *Updated 8/30: Clarified that this holds only for even numbers greater than 2.* **State** Goldbach’s Conjecture formally in our predicate logic notation^a You can quantify universally and existentially over M in the same way that you do over \mathbb{Z} ($\forall x \in M. P(x)$, $\exists y \in M. Q(y)$, ...). You can use standard mathematical symbols and operators ($<$, $>$, \leq , \geq , $=$, $+$, etc.).

^aYou do not need to prove this, and I would suggest not trying since it’s been an open problem since 1742. However, if you do happen to prove it, it would be worth substantial extra credit and probably a Fields Medal.

Task 3.2 (Written, 9 points).

Are the following statements true (provable) or false (have a counterexample)? Explain briefly (1-2 sentences).

- a) $\forall x \in \mathbb{Z}. \forall y \in \mathbb{Z}. x < y$
- b) $\forall x \in \mathbb{Z}. \neg(\exists a \in \mathbb{Z}. \exists b \in \mathbb{Z}. a > 1 \wedge b > 1 \wedge a * b = x)$
- c) $\forall x \in \mathbb{Z}. \exists y \in \mathbb{Z}. y = x * 2.$

Task 3.3 (Programming, 10 points).

Prove the following implication. **Submit your proof in the file `pred.log`, following the same instructions as in Task 2.**

$$(\forall x. P(x) \rightarrow Q(x)) \Rightarrow \neg(\exists x. P(x) \wedge \neg Q(x))$$

4 One more wrap-up question

Task 4.1 (Written, 0 points).

How long (approximately) did you spend on this homework, in total hours of actual working time?
Your honest feedback will help us with future homeworks.