HW 5 – First-Order Logic and Axiomatic Semantics

CS 477 – Spring 2025

Assigned April 12, 2025, 12:00 AM
Due April 19, 2025, 11:59 PM

Objectives and Background

The purpose of this HW is to test your understanding of

- First-order logic (Semantic Arguments, Alpha Equivalence)
- Axiomatic Semantics (proof programming using Dafny)

Policy

For all theory questions, including proofs, only typewritten solutions in LaTeX are allowed. Handwritten solutions will not be considered and will be awarded zero marks.
This homework is an individual assignment. In addition, directly copying from LLMs, including ChatGPT and Gemini, is strictly forbidden. In this homework, you can use these tools to gain insights; however, stringent measures will be implemented to detect any instances of direct replication from these sources. Any identified instances of suspicion will prompt an invitation to an oral examination.
Late policy: You have a total of four additional days throughout the term to accommodate late submissions for homework assignments. This cumulative allowance can be used at your discretion across different assignments. If you submit more than 5 minutes late for an assignment, then it is considered as using one full day of the late submission allowance. Once you exhaust the four-day late submission allowance, any further late homework submissions will result in a penalty of 33% grade reduction/day.
Please do not submit any code or use code to accomplish the non-programming problems unless we specifically instruct you to do so.

Turn-In Procedure (PLEASE READ CAREFULLY)

Use `https://www.gradescope.com/` to submit your homework. There are two assignments on Gradescope for this homework; one is designated for theory problems, and the other is for programming assignments. For the theory section of this assignment, you can submit a single PDF file or a set of photos just like HW 1.
For the coding problems, submit **exactly one file** to Gradescope, named <*net_id*>_interval.py, with <*net_id*> replaced by your own NETID. Failure to name your code files correctly will result in a 0 grade for the problem. Some tutorials on how to use Gradescope are available here `https://www.gradescope.com/get_started#student-submission`.

Substitution in Formulae (10 pts)

1. For a given first-order formula ψ, $\psi[t/x]$ is a formula where the variable x is replaced by the term t according to the substitution rules discussed in the lectures. Let P, Q be relations and f, g be functions. Compute $\psi[t/x]$ for the first-order formulae ψ shown below (2.5 pts each):

1. $\exists x.(x < y)[x/y]$
2. $\forall x.(P(x) \implies Q(x, f(y)))[g(y)/x]$
3. $\forall x.(P(x) \implies Q(x, f(y)))[g(y)/y]$
4. $\forall x.(P(x) \implies Q(x, f(y)))[g(x)/y]$

Alpha Equivalence (15 pts)

2. Prove that the following first-order formulae are alpha equivalent (5 pts each):

   1. $(x > 3 \land (\exists y.(\forall z.z \geq (y - x)) \lor (z \geq y))) \equiv^{\alpha} (x > 3 \land (\exists w.(\forall z.z \geq (w - x)) \lor (z \geq w)))$
   2. $(x > 3 \land (\exists y.(\forall z.z \geq (y - x)) \lor (z \geq y))) \equiv^{\alpha} (x > 3 \land (\exists w.(\forall y.y \geq (w - x)) \lor (z \geq w)))$
   3. $(x > 3 \land (\exists y.(\forall y.y \geq (u - x)) \lor (y \geq w))) \equiv^{\alpha} (x > 3 \land (\exists z.(\forall w.w \geq (u - x)) \lor (z \geq w)))$

Semantic Argument Method (20 pts)

3. Let P, Q be any relations. For each of the following formulas, either use the semantic argument method to prove validity or provide a counterexample for which the formula does not hold. (5 pts each)

   1. $(\exists x.P(x)) \implies (\forall y.P(y))$
   2. $(\exists x.\forall y.P(x, y)) \implies (\forall y.\exists x.P(x, y))$
   3. $(\exists x.P(x) \implies \forall x.Q(x)) \implies \forall x.(P(x) \implies Q(x))$
   4. $(\exists x.P(x) \land \exists x.Q(x)) \implies (\exists x.(P(x) \land Q(x)))$

Proving Soundness of Abstract Transformers using Z3 (55 pts)

We need to use Z3 solver for this section (used in Homework 2). It is a tool that you can use to declare symbolic variables and check the satisfiability of a formula defined over those variables. We recap the installation process here. You can install z3py using pip by the command: `pip install z3-solver`. You can refer to https://ericpony.github.io/z3py-tutorial/guide-examples.htm for a short introduction to Z3.
For this exercise questions, please **download the starter files here**. You will use Z3 to write and prove the soundness of transformers. Please follow the submission format for the previous assignments. In the submission, DO NOT print anything except the final answers.

4. We have provided the transformer for the example function $f(x) = x + b$ for some fixed b. Write the soundness check for this transformer in 'checkSoundnessofExampleTransformer'. **[15 pts]**

5. For the other two functions $f(x) = ax + b$ and $f(x) = ax^2$, you must write the transformer *and* prove its soundness. NOTE: Do not write trivial transformers. We will have hidden test cases for testing precision. **[20x2 pts]**