

From Language to Logic

Stanford CS221 Spring 2021-2022

Owner CA: Xiaoyuan Ni



Version: 05/23/2021

Ed Release Post

General Instructions

This (and every) assignment has a written part and a programming part.

The full assignment with our supporting code and scripts can be downloaded as [logic.zip](#).

- a.  This icon means you should write responses in [logic.pdf](#).
- b.  This icon means you should write code in [submission.py](#).

All written answers must be **typeset (preferably in LaTeX)**. We strongly recommend using Overleaf. A link to a LaTeX template with prompts can be found on Ed, or further down this page.

Also note that your answers should be **in order** and **clearly and correctly labeled** to receive credit. Be sure to submit your final answers as a PDF and tag all pages correctly when submitting to Gradescope.

You should modify the code in [submission.py](#) between

```
# BEGIN_YOUR_CODE
```

and

```
# END_YOUR_CODE
```

but you can add other helper functions outside this block if you want. Do not make changes to files other than [submission.py](#).

Your code will be evaluated on two types of test cases, **basic** and **hidden**, which you can see in [grader.py](#). Basic tests, which are fully provided to you, do not stress your code with large inputs or tricky corner cases. Hidden tests are more complex and do stress your code. The inputs of hidden tests are provided in [grader.py](#), but the correct outputs are not. To run the tests, you will need to have [graderUtil.py](#) in the same directory as your code and [grader.py](#). Then, you can run all the tests by typing

```
python grader.py
```

This will tell you only whether you passed the basic tests. On the hidden tests, the script will alert you if your code takes too long or crashes, but does not say whether you got the correct output. You can also run a single test (e.g., [3a-0-basic](#)) by typing

```
python grader.py 3a-0-basic
```

We strongly encourage you to read and understand the test cases, create your own test cases, and not just blindly run [grader.py](#).

In this assignment, you will get some hands-on experience with logic. You'll see how logic can be used to represent the meaning of natural language sentences, and how it can be used to solve puzzles and prove theorems. Most of this assignment will be translating English into logical formulas, but in Problem 4, we will delve into the mechanics of logical inference. Finally, for the ethics question you will revisit the ethical frameworks and concepts you learned about this quarter through once again writing negative impact statements.

NOTE: There is a final **mandatory** written question after the extra credit problem 5. Do not forget to answer this question! Also for this assignment only, there are no hidden test cases for the programming questions -- if you pass all the visible test cases, then you will get full credit!

We've created a LaTeX template [here](#) for you to use that contains the prompts for each question.

To get started, launch a Python shell and try typing the following commands to add logical expressions into the knowledge base.

```
from logic import *
Rain = Atom('Rain')           # Shortcut
Wet = Atom('Wet')             # Shortcut
kb = createResolutionKB()      # Create the knowledge base
kb.ask(Wet)                   # Prints "I don't know."
kb.ask(Not(Wet))              # Prints "I don't know."
```

```
kb.tell(Implies(Rain, Wet)) # Prints "I learned something."
kb.ask(Wet)                 # Prints "I don't know."
kb.tell(Rain)               # Prints "I learned something."
kb.tell(Wet)                # Prints "I already knew that."
kb.ask(Wet)                 # Prints "Yes."
kb.ask(Not(Wet))            # Prints "No."
kb.tell(Not(Wet))           # Prints "I don't buy that."
```

To print out the contents of the knowledge base, you can call `kb.dump()`. For the example above, you get:

```
==== Knowledge base [3 derivations] ===
* Or(Not(Rain),Wet)
* Rain
- Wet
```




In the output, '*' means the fact was explicitly added by the user, and '-' means that it was inferred. Here is a table that describes how logical formulas are represented in code. Use it as a reference guide:

Name	Mathematical notation	Code
Constant symbol	stanford	<code>Constant('stanford')</code> (must be lowercase)
Variable symbol	x	<code>Variable('\$x')</code> (must be lowercase)
Atomic formula (atom)	Rain	<code>Atom('Rain')</code> (predicate must start with uppercase)
	LocatedIn(stanford, x)	<code>Atom('LocatedIn', 'stanford', '\$x')</code> (arguments are symbols)
Negation	\neg Rain	<code>Not(Atom('Rain'))</code>
Conjunction	$\text{Rain} \wedge \text{Snow}$	<code>And(Atom('Rain'), Atom('Snow'))</code>
Disjunction	$\text{Rain} \vee \text{Snow}$	<code>Or(Atom('Rain'), Atom('Snow'))</code>
Implication	$\text{Rain} \rightarrow \text{Wet}$	<code>Implies(Atom('Rain'), Atom('Wet'))</code>
Equivalence	$\text{Rain} \leftrightarrow \text{Wet}$ (syntactic sugar for $\text{Rain} \rightarrow \text{Wet} \wedge \text{Wet} \rightarrow \text{Rain}$)	<code>Equiv(Atom('Rain'), Atom('Wet'))</code>
Existential quantification	$\exists x. \text{LocatedIn}(\text{stanford}, x)$	<code>Exists('\$x', Atom('LocatedIn', 'stanford', '\$x'))</code>
Universal quantification	$\forall x. \text{MadeOfAtoms}(x)$	<code>Forall('\$x', Atom('MadeOfAtoms', '\$x'))</code>

The operations `And` and `Or` only take two arguments. If we want to take a conjunction or disjunction of more than two, use `AndList` and `OrList`. For example: `AndList([Atom('A'), Atom('B'), Atom('C')])` is equivalent to `And(And(Atom('A'), Atom('B')), Atom('C'))`.

Problem 1: Propositional logic

Write a propositional logic formula for each of the following English sentences in the given function in `submission.py`. For example, if the sentence is *"If it is raining, it is wet,"* then you would write `Implies(Atom('Rain'), Atom('Wet'))`, which would be $\text{Rain} \rightarrow \text{Wet}$ in symbols (see `examples.py`). Note: Don't forget to return the constructed formula!

- a.  [2 points] "If it's summer and we're in California, then it doesn't rain."
- b.  [2 points] "It's wet if and only if it is raining or the sprinklers are on."
- c.  [2 points] "Either it's day or night (but not both)."

You can run the following command to test each formula:

```
python grader.py 1a
```

If your formula is wrong, then the grader will provide a counterexample, which is a model that your formula and the correct formula don't agree on. For example, if you accidentally wrote `And(Atom('Rain'), Atom('Wet'))` for "If it is raining, it is wet," then the grader would output the following:

```
Your formula (And(Rain,Wet)) says the following model is FALSE, but it should be TRUE:
* Rain = False
* Wet = True
* (other atoms if any) = False
```


In this model, it's not raining and it is wet, which satisfies the correct formula $\text{Rain} \rightarrow \text{Wet}$ (TRUE), but does not satisfy the incorrect formula $\text{Rain} \wedge \text{Wet}$ (FALSE). Use these counterexamples to guide you in the rest of the assignment.

Problem 2: First-order logic


Write a first-order logic formula for each of the following English sentences in the given function in `submission.py`. For example, if the sentence is "There is a light that shines," then you would write `Exists('$x', And(Atom('Light', '$x'), Atom('Shines', '$x')))`, which would be $\exists x. \text{Light}(x) \wedge \text{Shines}(x)$ in symbols (see `examples.py`).

Tips:

- You can directly write '\$x' instead of `Variable('$x')` to represent a variable symbol.
- Python tuples can span multiple lines, which help with readability when you are writing logic expressions (some of them in this homework can get quite large).

- a.  [2 points] "Every person has a mother."

Note: You do NOT have to enforce that the mother is a "person".

- b.  [2 points] "At least one person has no children."

Note: You do NOT have to enforce that the child is a "person".

- c.  [2 points] Create a formula which defines `Father(x,y)` in terms of `Male(x)` and `Parent(x,y)`.

- d.  [2 points] Create a formula which defines `Granddaughter(x,y)` in terms of `Female(x)` and `Child(x,y)`.

Note: It is ok for a person to be their own child.

Problem 3: Liar puzzle

Someone crashed the server, and accusations are flying. For this problem, we will encode the evidence in first-order logic formulas to find out who crashed the server. You've narrowed it down to four suspects: John, Susan, Mark, and Nicole. You have the following information:

- Mark says: "It wasn't me!"
- John says: "It was Nicole!"
- Nicole says: "No, it was Susan!"
- Susan says: "Nicole's a liar."
- You know that exactly one person is telling the truth.

6. You also know exactly one person crashed the server.



a. [8 points] Fill out `liar()` to return a list of 6 formulas, one for each of the above facts.

The grader is set up such that you could run individual parts `3a-0` to `3a-5` to debug each formula only if you implement them **in the order specified**.

You can test your code using the following commands:

```
python grader.py 3a-0
python grader.py 3a-1
python grader.py 3a-2
python grader.py 3a-3
python grader.py 3a-4
python grader.py 3a-5
python grader.py 3a-all # Tests the conjunction of all the formulas
```

To solve the puzzle and find the answer, `tell` the formulas to the knowledge base and `ask` the query `CrashedServer('$x')`, by running:

```
python grader.py 3a-run
```

Problem 4: Odd and even integers

In this problem, we will see how to use logic to automatically prove mathematical theorems. We will focus on encoding the theorem and leave the proving part to the logical inference algorithm. Here is the theorem:

If the following constraints hold:

1. Each number x has exactly one successor, which is not equal to x .
2. Each number is either odd or even, but not both.
3. The successor of an even number is odd.
4. The successor of an odd number is even.
5. For every number x , the successor of x is larger than x .
6. Larger is a transitive property: if x is larger than y and y is larger than z , then x is larger than z .

Then we have the following consequence:

- For each number, there is an even number larger than it.

Note: in this problem, "larger than" is just an arbitrary relation, and you should not assume it has any prior meaning. In other words, don't assume things like "a number can't be larger than itself" unless explicitly stated.



a. [8 points] Fill out `ints()` to construct 6 formulas for each of the constraints. The consequence has been filled out for you (`query` in the code).

The grader is set up such that you could run individual parts `4a-0` to `4a-5` to debug each formula only if you implement them **in the order specified**. You can test your code using the following commands:


```
python grader.py 4a-0
python grader.py 4a-1
python grader.py 4a-2
python grader.py 4a-3
python grader.py 4a-4
python grader.py 4a-5
python grader.py 4a-all # Tests the conjunction of all the formulas
```

To finally prove the theorem, `tell` the formulas to the knowledge base and `ask` the query by running model checking (on a finite model):


```
python grader.py 4a-run
```

Problem 5: Semantic parsing (extra credit)


Semantic parsing is the task of converting natural language utterances into first-order logic formulas. We have created a small set of grammar rules in the code for you in `createBaseEnglishGrammar()`. In this problem, you will add additional grammar rules to handle a wider variety of sentences. Specifically, create a `GrammarRule` for each of the following sentence structures.

- a.  [2.5 points] Example: *Every person likes some cat.* General template:

$\$Clause \leftarrow \text{every } \$Noun \$Verb \text{ some } \$Noun$

- b.  [1.5 points] Example: *There is some cat that every person likes.* General template:

$\$Clause \leftarrow \text{there is some } \$Noun \text{ that every } \$Noun \$Verb$

- c.  [2 points] Example: *If a person likes a cat then the former feeds the latter.* General template:

$\$Clause \leftarrow \text{if a } \$Noun \$Verb \text{ a } \$Noun \text{ then the former } \$Verb \text{ the latter}$

After implementing these functions, you should be able to try some simple queries using `nli.py`! For example:

```
$ python nli.py
> Every person likes some cat.

>>>> I learned something.
-----
> Every cat is a mammal.

>>>> I learned something.
-----
> Every person likes some mammal?

>>>> Yes.
```

Problem 6: Revisiting Ethical Issue Spotting

In [Homework 1: Foundations](#), you practiced spotting ethical issues in real-world scenarios. In this question, you will return to a similar task equipped with the ethical frameworks and concepts you have learned and applied during the quarter. Just like on homework 1, you will write a negative impacts statement for the scenarios below that justifies your decision using one of the ethical frameworks / concepts you learned about this quarter. Specifically, you will:

1. List one of the sixteen numbered ethical concerns taken from the [NeurIPS Ethical Guidelines](#) page.
2. List an ethical concept or framework from the course, which can be found listed below.
3. Finally, justify why the chosen ethical concern from the NeurIPS guidelines applies to this scenario *using the ethical concept or framework from the course*.


Ethical Frameworks / Concepts From the Course:

1. Theories of Distributive Justice (utilitarianism, prioritarianism, difference principle, avoid compounding injustice) If using this framework in your answer, pick a specific theory of distributive justice.
2. Communicating Limitations with a Transparency Statement
3. Theories of Responsibility To Future Generations (Veil of Ignorance, Uncertain Future, Precautionary Risks, Symmetry of Future Generations, Vulnerability of Future Generations) If using this framework in your answer, pick a specific theory of responsibility to future generations.
4. Wicked Problems
5. Ethics Dumping
6. Dual-Use Technologies

Example Scenario: Recent resolutions by the United Nations and policy statements by roboticists propose a 5 year moratorium on “development, deployment, transfer, and use of anti-personnel lethal autonomous weapon systems,” or weapons systems that can target and kill people without direct human oversight [4]. Human oversight ensures that a human takes moral and legal responsibility for all actions and for ensuring that actions comply with the laws of war. Some lethal autonomous drones couple facial recognition, used to autonomously identify their targets, with action-planning used to kill them.


A research paper proposes a new AI-based action-planning system for automatically detecting and aiding tourists stranded in Death Valley in the summer. The system couples two systems on an unmanned drone: (a) computer vision to do vehicle recognition and detect the stranded cars in the desert and (b) a motorized arm to provide the stranded tourists with water bottles while rescuers are on their way, since humans can easily die of heatstroke and exhaustion when temperatures exceed 50°C, as often happens in Death Valley. The drone is able to detect disabled vehicles with almost 99.5% accuracy, and so the paper proposes that the drone system can likely be fully automated with no human verification of the stranded car before delivering the water, also speeding up the time of water delivery and potentially saving the stranded tourists before heatstroke sets in. [1].

Example answer: The proposed technology Directly facilitates injury to living beings, as it can be used as part of a weapons system without significant modification. A drone capable of recognizing cars and delivering water, and more importantly coupling these capacities as part of a multi-stage autonomous planning system, is a dual-use technology, or a technology that can be used for peaceful purposes or as part of a weapon of war. It would be very simple to replace the water delivery system with a weapon of war, and since the drone has a very high accuracy rate, it can likely be operated successfully without human supervision, in violation of the UN resolution.

- a.  [3 points] **Scenario 1:** After their study design was not approved by IRB in their home country, a research group tests their new low-cost AI-based screening device for cervical cancer in another country with a lower standard for clinical trials. In the home country, the standard of care for the control versus test group for a new screening technology is to still administer the standard of care test to all patients in the trial, which in this case would be a Pap smear. However, in the host country this research team administered only the low-cost AI-based screen to 140,000 women in the test group and administered **no** screen to 140,000 women in the control group. No pap smears were administered to any women in the study. Of the 140,000 women who received no screen in the control arm of the study, 254 of them ended up developing cervical cancer and dying as a result. [2].

What we expect: A 2-5 sentence paragraph for each of the scenarios where you **A.** identify one ethical concern from the NeurIPS Ethical Guidelines and **B.** one ethical concept from the course and justify why you think they apply, and **C.** link the ethical concept to the NeurIPS ethical concern. Copy out and underline both the ethical checklist item and the ethical concept to which you are referring as part of your answer (i.e.: Severely damage the environment; wicked problems).

This project is an example of ethics dumping which led to a research program that raises human rights concerns (item 3. from NeurIPS ethical concerns). Because the research group performed their research in a country with a lower bar for clinical trials approval, they were able to conduct their screening assessment without providing the standard-of-care Pap smear to any of the women in the study. This ethics dumping meant that women who were in the control arm of the study had their access to healthcare negatively impacted by being denied the potentially life-saving cervical cancer screen they were told they would be receiving. Indeed, hundreds of women died from the lack of proper cervical cancer screening.

- b.  [3 points] **Scenario 2:** A recent economic forestry paper proposed an MDP model for optimizing commercial logging in a National Forest that uses a 30-year time horizon to suggest when trees should be cut for optimal logging yields. It recommends cutting each tree once it is at 30 years or older, suggesting the policy of harvesting every tree in the forest every 30 years and leading to a clear-cut forest by the end of the simulation. [3].

What we expect: A 2-5 sentence paragraph for each of the scenarios where you **A.** identify one ethical concern from the NeurIPS Ethical Guidelines and **B.** one ethical concept from the course and justify why you think they apply, and **C.** link the ethical concept to the NeurIPS ethical concern. Copy out and underline both the ethical checklist item and the ethical concept to which you are referring as part of your answer (i.e.: Severely damage the environment; wicked problems).

This model, if applied, would Severely damage the environment because it fails to incorporate a theory of responsibility to future generations. Specifically, under symmetry of future generations, the researchers should have considered the effects of their model's predictions from the perspective of the year of the end of the simulation, where the forest is destroyed through clear-cutting. From the perspective of a generation born after the forest has been clear cut, this is unjust, as they will be unable to benefit from the forest as the present generation was able to. There is a large body of work that shows clear-cutting negatively impacts the local ecosystem where the cutting occurs. Therefore by choosing a short 40 year time horizon and failing to incorporate a symmetry of future generations perspective, this work will likely lead to severe damage to the environment. See [this](#) paper for more info.

[1] Inspired by <https://www.cc.gatech.edu/ai/robot-lab/online-publications/AWS.pdf>.

[2] Modified from [cervical cancer screening in India](#).

[3] Modified from [optimal forest growth modeling](#).