

# Assignment 3: Logical Inference with Wumpus World

**Deadline: Monday, December 3, 11:59:59PM**

**As on Assignments 1 and 2, you have the option of working in groups of up to three people.** You are free to either stay with the same team or pick a new one.

The purpose of this assignment is to give you a chance to implement a basic, first-order logic reasoning system. In this project, you will implement a system that does unification and resolution on clauses. We will be assuming that the rules are already encoded in clause form. You will be testing your system on various instances of the Wumpus world.

For this project, you need to write a “problem generator” whereby you will create several Wumpus worlds of various sizes. The Wumpus world cave will be generated as a square consisting of several cells. You will use the description of Wumpus world in Section 7.2 of your text book (and as we also discussed in class).

After generating the world, you will select one of the empty cells at random and place the gold there. You will do the same for the Wumpus. Also place pits with 20% probability in the cells (as discussed in Section 7.2).

Once the gold is found, you must return to cell [1,1] (also the starting cell). You will receive the 1000 points when you enter cell [1,1] with the gold. The game ends at this point. If you get killed (by the Wumpus or the pit), the game also ends with points lost as stated in section 7.2. These are the only 3 ways the game can end. Remember it costs 1 point for every move. It doesn't cost anything to turn or bump into a wall.

You need to generate first-order rules to tell our agent how to behave. As described in class, our agent can only see the current cell, can smell adjacent cells, and can feel wind from adjacent cells. It can also hear a scream from anywhere in the cave, so if an arrow finds its mark, our agent will know. These rules may be provided in clause form, so you do not need to create a rule converter. After building the Wumpus worlds, you will use your rule set with your reasoning system and have our agent explore each cave.

You need to report on the following for each run:

How did the game end (found gold, killed by Wumpus or fell in a pit). Also

How many cells the agent entered

Performance measure at end of game

Generate Wumpus worlds of sizes:

4X4

5X5

8X8

10X10

## Report Checklist

Your report should briefly describe your implemented solution. Your description should focus on the most "interesting" aspects of your solution, i.e., any non-obvious implementation choices (including programming language chosen) and parameter settings, and what you have found to be especially important for getting good performance. Feel free to include pseudocode (**DO NOT PUT CODE IN YOUR REPORT**) or figures if they are needed to clarify your approach. Your report should be self-contained and it should (ideally) make it possible for us to understand your solution without having to run your source code.

Give your First Order Logic rules (in logic and natural language) and any choices you made in determining those rules (that weren't given in Section 7.2). Report the metrics given above.

It is required that you provide an analysis of your results.

### Statement of individual contribution:

- All group reports need to include a brief summary of which group member was responsible for which parts of the solution and submitted material. We reserve the right to contact group members individually to verify this information

## Submission Instructions

By the submission deadline, **one designated person from the group** will need to upload the following to D2L/Brightspace:

1. A **report** in **PDF format**. Be sure to put the **names** of all the group members at the top of the report. The name of the report file should be **lastname\_firstname\_a3.pdf** (based on the name of the designated person).
2. Your **source code** compressed to a **single ZIP file**. The code should be well commented, and it should be easy to see the correspondence between what's in the code and what's in the report. You don't need to include executables or various supporting files (e.g., utility libraries) whose content is irrelevant to the assignment. If we find it necessary to run your code in order to evaluate your solution, we will get in touch with you. **INCLUDE YOUR OUTPUT FILE IN THIS ZIP FILE.**

The name of the code archive should be **lastname\_firstname\_a3.zip**.

Multiple attempts will be allowed but only your last submission before the deadline will be graded. **We reserve the right to take off points for not following directions.**

**Late policy:** You must submit by Midnight of Dec 3 the full package (report and source code). No exceptions.