CSC 4240/5240 Artificial Intelligence
Homework 3
Due: **Wed October 15th, 11:59pm**

[50 total points] General Instructions: Submit your files as a single zip file under Assessments → Homework 3 for this course. Note that you may submit multiple times, but we will only grade the most recent entry submitted before the deadline.

For this homework you will implement a reflex agent with state to play the Wumpus World game. Specifically,

1. Download the Wumpus World simulator from iLearn.
2. Read the README for instructions on how to use the simulator.
3. For this homework, the world size is 4x4, there are no pits, the Wumpus is always in location (4,4), and the gold can be anywhere, even in (4,4).
4. Implement an agent that executes the following reflex rules (**and only these rules**).
    a. If the Glitter percept is True, then execute the GRAB action.
    b. If the agent is in the (1,1) location and has the gold, then CLIMB.
    c. If the agent has an arrow, and the agent is in the top row (Y=4), and the agent's orientation=RIGHT, then SHOOT.
    d. If the agent has an arrow, and the agent is in the rightmost column (X=4), and the agent's orientation=UP, then SHOOT.
    e. If none of the above conditions are met, then the agent should randomly choose one of the actions: GOFORWARD, TURNLEFT, TURNRIGHT.
5. Your agent should maintain state information about the agent's location and orientation and whether it has an arrow and the gold. Your agent does not have access to the game state within the simulator, so the agent will have to update its own state after each turn. But you may copy/include/import simulator code into your agent.
6. Your agent should be implemented entirely in the Agent.h and Agent.cc files. You should look at the other files to understand how the code works. Besides the C++ implementation, you may implement your agent in Agent.py file. **You MAY need to update the PYTHON-INC and PYTHON-LIB variables in the make file with the correct python configuration path on your computer. The compiler will give an error "unable to find file xxxx" if you don't fix this issue. You should not dwell too much on this problem if it persists and use the C++ code instead**. You may also include an optional readme.txt file with any extra instructions for running your agent. Your agent should not require any user input. Your agent will be tested by copying only your Agent.h and Agent.cc files, or Agent.py file, into a fresh copy of the simulator code, and compiling and running it on several test worlds (there is a default testworld file included with the simulator). **You will need to implement more helping functions to get your code to work.**

Your grade will be based on satisfying the above requirements, performance on the test worlds (your code should work as expected and cover corner cases, if exist), and good programming style like modularity, documentations, naming conventions, indentation, etc. Checkout the following links to programming style guides: https://google.github.io/styleguide/cppguide.html (C/C++) and https://www.python.org/dev/peps/pep-0008/ (Python).