CS 171 – Introduction to Artificial Intelligence

Programming Assignment: Wumpus World AI

Abdullah Younis

younisa@uci.edu

June 25, 2016

C++

*** If you are in a Windows Environment you will need to append ".exe" to the end of an executable. For example, instead of "./PracticeGame" one would type "./PracticeGame.exe"**

However, if you are in a Linux environment (Openlabs) you will not need the extra ".exe"

**** See Section IV.A for help understanding the console game screen.**

***** If you get a permission denied error when running the scorer, follow these steps:**

1. **Open your terminal.**
2. **Navigate to the Wumpus_World_cpp_shell/_Resources/ScorerScripts**
3. **Execute the command: chmod +x *.sh**

****** If you get an error message about CXXABI_1.3.8 and GLIBCXX_3.4.21, follow these steps:**

1. **On Windows: Make sure your gcc and g++ compilers are updated**
2. **On Openlab execute: module load gcc/5.2.0**

I. Code Base

```
Wumpus_World_cpp_shell
|-- Cpp_Shell.pdf                          // Shell Documentation
|-- Makefile                               // Makefile script for make
|-- _Resources                             // See Section IV
    \-- AverageAI.exe
    \-- PoorAI.exe
    \-- PracticeGame.exe
    \-- RandomAI.exe
    \-- SmartAI.exe
    \-- World_Generator.exe
    \-- WumpusWorldTemplateWorldFile.txt
    \-- RandomAI.cpp
    \-- RandomAI.hpp
    \-- Worlds
        \-- <TestWorlds>
    \-- ScorerScripts
        \-- Scorer.exe                     // Averages Scores
        \-- Scorer.sh                      // Runs programs
        \-- Makefile                       // Makefile script for score
|-- bin                                    // Binary Folder
    \-- [Final Program]
|-- doc                                    // Document Folder
    \-- [Project Report]
|-- src                                    // Source Folder
    \-- Agent.hpp
    \-- main.cpp
    \-- Tile.hpp
    \-- World.cpp
    \-- World.hpp
    \-- MyAI.cpp *
    \-- MyAI.hpp *
```

You are only allowed to have edits to MyAI.cpp and MyAI.hpp in your final submission.

II. Programming

In this programming assignment, you will edit the MyAI.cpp and MyAI.hpp files. In essence your agent can perceive its environment using:

```
virtual void perceiveResult ( std::set<Percept> percepts );
```

which will be called on every game tick with an updated set of percepts. You will be required to use:

```
virtual Action getAction();
```

to return an action taken by your agent at every game tick.

A “Percept” is an enum type defined in Agent.hpp to be one of the follow:
STENCH, BREEZE, GLITTER, BUMP, SCREAM

On every game tick your agent will be supplied a set of these.

An “Action” is an enum type defined in Agent.hpp to be one of the follow:
FORWARD, TURN_LEFT, TURN_RIGHT, GRAB, SHOOT, CLIMB

Feel free to write helper functions as long as your edits don’t leave MyAI.cpp or MyAI.hpp.

III. Compile/Score/Submit/Run

To use the make command, you need to be in a UNIX-Like environment.

A. Compiling

Follow these steps to compile to your program:

1. Open your terminal.
2. Navigate to the Wumpus_World_cpp_shell folder.
3. Run **make** (or **make all**) to compile the source code.

If successful, you will see a MyAI executable inside of the bin folder.

B. Scoring

Follow these steps to test your program on the scorer:

1. Open your terminal.
2. Navigate to the Wumpus_World_cpp_shell folder.
3. Run **make score**

If successful, the scoreboard will get printed to the console. Not having a binary file in /bin will only result in your score not getting printed to the console.

C. Submitting

Follow these steps to submit your program to EEE:

1. Place your Project Report inside of the /doc folder.
2. Open your terminal.
3. Navigate to the Wumpus_World_cpp_shell folder.
4. Run **make submission**
5. If successful, a submission.zip file will be created in the Wumpus_World_cpp_shell folder. Rename it according the naming rules stated below.
6. Submit the renamed archive to EEE.

Your EEE DropBox submission must be a single "zipped" file named "yourLastName_yourUCINumericID_yourTeamName." NO SPACES OR ANY OTHER SPECIAL UNIX CHARACTER in yourTeamName. Please restrict your TeamName to characters, digits, hyphen, and underscore, or else you may lose points.

D. Running

MyAI (as well as SmartAI, AverageAI, PoorAI, and RandomAI) can be run with 0, 1, or 2 command line arguments. When run with no command line arguments, the AI will create a random board based on user-defined dimensions. The user will also be asked to see the game step by step, or just skip to the ending. This might be useful for debugging your AI.

You can also run an AI with 1 or 2 command line arguments. The first argument is the world file from which to create the board, and the second argument is a file to record scores.

Follow these steps to run your program:

1. Open your terminal.
2. Navigate to the Wumpus_World_cpp_shell folder.
3. Run **bin/MyAI [World_Input_File] [Score_Output_File]**

The [] parts are optional.

IV. Resources

A. PracticeGame

The PracticeGame resource is a great resource to get a feel for the game mechanics. If you run this game it will allow you to play the Wumpus World game instead of an AI. The controls are listed for you during each move.

Understanding the printed game board:

- Each Dot Represents a cave room.
- The '**@**' is your agent.
- A '**P**' represents a pit.
- A '**W**' represents the wumpus.
- A '**D**' represents the dead wumpus.
- A '**G**' represents the gold.
- A '**B**' represents a breeze.
- A '**S**' represents a stench.

B. AIs

You can get a feel for how the standard AIs work with their compiled executables. They run just like MyAI runs, see section III.D.

C. Worlds and World_Generator

You can use the World_Generator to generate a large set of randomized Worlds for testing. There is also a World Template File supplied, if you would like to create your own specific worlds.

D. Scorer

The scorer program will measure your agent's performance on all the worlds in the Worlds directory. It will display your score alongside the scores of the standard AIs. They too will be tested on the same worlds. See section III.B for help running the scorer.

E. RandomAI

The source code of the RandomAI has been supplied for your study. It is a dummy file and will not be compiled.

V. Awknowledgment

A. **Minjae Wee** for the SmartAI, RandomAI, AverageAI, and PoorAI code, and for enhancements to the original C++ code.

B. **Tiancheng Xu** for the original C++ shell.

C. **Qiuxi (Charles) Zhu** for this Project Documentation format and instructions.