CONTENTS

- Assignment Overview
- Setup
- Rules
- Constraints
- GTP Commands
- The Usual Warnings, Hints and Details - Read them All
- Pre-submission Test and Submission
- Marking

Cmput 455 Assignment 4

Due Apr 21, **11:55pm**. Submit via eClass submission link on the main eClass course page.
Late submission (with 20% deduction) Apr 23, 11:55pm. Submit via separate eClass link for late submissions.

In this assignment, you develop your own player for the game of Gomoku. You can implement any player you want using any techniques, as long as it's in Python3 and follows the rules and constraints below. We will test the performance of your player in two tournaments.

- In the **marking tournament**, for regular assignment marks, we will match your player against a fixed set of our Gomoku players.
- The **championship tournament**, for bonus points and eternal fame, will be a tournament among student programs. The winning team will be crowned the 2021 Cmput 455 Gomoku champions.

Setup

1. First, as in previous assignments, make sure you have your Python 3, NumPy and GoGui 1.51 set up. You can review the procedures under Lecture 3 activities.
2. Download assignment4.tgz and expand it. The directory `assignment4` contains:
   - Code for two of the four opponents for the marking tournament: a random_player directory containing the random Gomoku player `Gomoku2.py`, and a flat_mc_player directory containing the simulation-based player `Gomoku3.py`.
   - A script play.py that demonstrates for playing test games between two programs.
   - A directory gomoku4 which you will use to implement your player. Right now this is just a copy of flat_mc_player, but you can (and should!) replace the playing engine with your own.
   - We will not publish the other two players before the tournament, but here is a little bit of information about them. One will be an optimized MCTS-based player, and the other one will be a mystery opponent. We will tune the strength of these two players such that they will be a little bit more challenging than the simulation player, but not impossible to defeat within the scope of an assignment.
3. Modify/add code in directory assignment4 only to implement your solution.

RULES

- All games will be played on a 7x7 board.
- The time limit will be 60 seconds per move. If a program does not play within the time limit, it will be killed by the script and instantly loses the game. We recommend leaving a little bit of extra time and not fully use the 60 seconds, just to be safe.
- The memory limit will be 1 Gigabyte per program.
- If a program generates an illegal move, it instantly loses the game.
- If a program crashes or exceeds the memory limit, it instantly loses the game.

- If a game is interrupted for other reasons, it may be replayed at the decision of the instructor.
- Your player is allowed but not required to resign. It will not be called to generate a move after the game is over.

TEAM NAMES

We would like to provide updates as the tournament progresses.
So in your readme file, please provide a teamname as well, which we will use for public updates. The team name should not contain the team member names to provide a level of anonymity.

As stated below, **when you submit your assignment, please rename the dir `gomoku4` to your team name**, using underscores in place of spaces and removing any special characters if needed.

CONSTRAINTS

- You can use any code provided as part of this course, but no other outside sources of code.
- Your program is not allowed to use more than one thread.
- Your program is not allowed to use programming languages other than Python3.
- Your program is allowed to read/write files within your assignment4 directory only.
- The total size of assignment submissions, including all files, is limited to 1 Megabyte uncompressed.
- Further reasonable constraints to prevent abuse may be imposed as we become aware of them.

GTP COMMANDS

- `genmove color`

  Your `genmove` command should generate a move using your player, and comply with the rules and constraints above.
- All other existing GTP commands should be left as-is.

The Usual Warnings, Hints and Details - Read them All

- Please make sure that these details are correct in your assignment 4 submission.
- Your file must be a valid tgz file named `assignment4.tgz` which can be uncompressed with `tar -xf assignment4.tgz`
- Keep the name of your main file `Gomoku4.py` and **keep it in a directory named after your team.**
- Do not introduce extra levels of directories.
- You may add extra python files within the same directory.
- By default, we assume that teams will stay the same as in assignment 3. If you change your team, email the TAs the new information before the assignment 4 deadline.

Pre-submission Test and Submission

Play one test game against `Gomoku2.py` as follows:

- Change player1 in the script to your progam's path, team_name/Gomoku4.py
- Use numGame=1 as the parameter for the playGames() function.
- Run play.py
- After the game finishes, there will be a file named game_result.txt, that contains the summary of the test games.

Follow the same general steps as in assignment 1 to create your `presubmission.log` file and your submission, but (of course) using your `assignment4` directory, `assignment4.tgz` as file name, and the sample test game described above as test. Add both your `game_result.txt` and `readme.txt` to your `assignment4` directory. **The readme file should also state your public team name for the tournament.**

Marking

There will be 5+2 marks for this assignment.

- 1 mark for your file `presubmission.log` which shows the log of your correct test execution, and your `readme.txt` file (as in Assignment 1).
- 4x1 mark for your wins and draws against our 4 test opponents.
- Up to 2 bonus marks for doing well in the championship tournament.

Created: Mar 18, 2019 Last modified: Jan 8, 2021

Martin Müller and Ting-Han Wei