1. title of project

OPTION 2: Game Playing Techniques

1. names and roles of each teammate

Jingqiang Wang:

Junhao Zeng: Created the static evaluation function and the movement generator.

1. what the program is supposed to do

This program is supposed to be able to play a game of Baroque Chess smartly by using some AI techniques.

1. technique used and brief description (half a page) of how that technique works. If you use multiple AI techniques then describe each one but with somewhat less detail for each one

Minmax Search:

Zobrist hashing:

alpha-beta pruning:

1. either a screen shot or a transcript of an interesting sample session

1. brief demo instructions

In order to play Baroque chess, you need to store BaroqueGameMaster.py and Thanos\_BC\_Player in a same folder, and then run BaroqueGameMaster.py to begin.

1. code excerpt showing some interesting part(s) of your Python code and some explanation of it

explanation:

Another enhanced techniques used in our project is using the more complicated static evaluation method to evaluate the board.  
First, for a specific status of the board, we count how many enemies all pincer can kill in one move, and adding five points for each kill. Vice versa, we count how many allies can be killed by enemy pincer in one move, and deducting five points for each kill.  
Besides, because the more allies a king has in his surrounding, the safer the king is, so we check the surroundings of our king, the surrounding means the eight cells which connect to the king cell. For each surrounding cell, if it is an ally, we add five points, if it is an enemy, we deduct five points. Vice versa, for each surrounding cell of enemy king, if it is an ally, we deduct five points, if it is an enemy, we add five points.  
What’s more, we count how many allies are frozen by enemy’s freezer or imitator, for each frozen ally, we deduct one point. Vice versa, we count how many enemy are frozen by our freezer or imitator, for each frozen enemy, we add one point.  
In the end, we add or de a specific points for each piece depending on what it is.

1. brief description of what each team member learned in this project

Junhao Zeng:

After this project, I learned how to use python to design complex code. Before this project, I mainly use java, and life becomes so easier after I use python to write code.

Besides, After finished this project, I know how to implement minimax search, alpha-beta pruning and Zobrist hash, and I also know how to write a good static eval function.

1. what you would like to add to your program if you had more time

If we had more time, we would implement feature-based learning approach to improve our static eval, the weighted feature vector could end up acting as decent static eval with enough training.

1. citations for any references you used in the project. This should include the names and URLs of any websites that you used and whose ideas or other resources were incorporated into your project. In each case, describe in a sentence what role that website played in your project and what you incorporated from it.

We used Wikipedia to learn about the basic movement of the Braoque chess.

<https://en.wikipedia.org/wiki/Baroque_chess>

We used Wikipedia to learn how to implement Alpha-beta pruning.

https://en.wikipedia.org/wiki/Alpha–beta\_pruning

we used Stackoverflow to learn how implement Zobrist Hashing

https://stackoverflow.com/questions/10067514/correctly-implementing-zobrist-hashing

1. For 5 points of extra credit, include a section with a heading "Partners' Reflections" with two subsections, one for each partner. Each subsection should give the partner's name, main role(s) in the project, a description of the challenges and benefits of the partnership from that partner's perspective.

Junhao Zeng:

Main role: Created the static evaluation function and the movement generator.

Challenges: Since each of us only wrote a part of codes for this project, one of the challenges were how to make codes more readable for parter. In order to solve this problem, we wrote some comments when some part of codes were tricky. In this way, this saves us a lot of time.

Benefits: The benefits are very obvious. By breaking up this project into two pieces, both of us can focus on the part for which we are responsible, so we can finish this project better. Besides, after we finished individual part, we can share idea with each other. Therefore, we can actually learn more than doing this project alone.