CS 461 Project 1

Nick King

***Files***

*Bearing\_Off.py - - User Interface, functionality addresses project description and requirements.*

*Bearing\_Off\_Build.py – Data base building program. Can be used to find expected value of any given position.*

*Bearing\_Data.txt – Database storing position, expected value, and max value.*

***Notes***

*-I am certain doubles functionality is working in the database building process, but I am not sure it works correctly with all instances in the UI*

*-Some Max # of moves may not be 100% accurate*

**Major Data Structures**

Both the data base builder and the user interface make extensive use of a dictionary and several lists. Lists are used primarily to store sets of positions during the unguided search, and the dictionary is used to store string positions as keys, and tuples containing the expected moves and the max moves as the values.

**Major Functions**

The functions that do the heavy lifting are eval(position) and gen\_moves(position, roll).

Together they build the database, as eval calls gen\_moves and recursively branches to find a known case, and eventually returns the expected number of moves for that position.

The db\_builder() function generates the database using nested for loops to iterate through all possible positions containing 6 or fewer stones.

The UI uses dictionary accesses primarily to analyze positions and rolls, after the database has already been built up.

**Database building**

The database was built using the db\_builder() function – if for some reason a given position is not in the database, simply run the Bearing\_Off\_Build.py and enter the position you are looking for, and the database will be updated.

**Algorithmic Correctness**

I will not argue that my method was optimal, but I will say that the moves generated from a position and a roll are always correct, and the expected value is accurate when compared with probabilistic hand calculations. For a simple example, after generating the database the position “000001” has an expected value of 1.25 – hand calculation shows 1 \* (27/36) + 2 \* (9/36) = 1.25