HEURISTIC ANALYSIS

For an Adversarial Game Playing Agent for Isolation

Submitted by

Muhammad Alvin Pramadrian

WID180708

TABLE OF CONTENTS

[INTRODUCTIONS 2](#_Toc58375806)

[CUSTOM HEURISTICS 3](#_Toc58375807)

[1. HEURISTIC 1: AGRESSIVE HEURISTICS 3](#_Toc58375808)

[2. HEURISTIC 2: DEFENSIVE HEURISTICS 3](#_Toc58375809)

[3. HEURISTIC 3: MAXIMIZING WIN CHANCES 3](#_Toc58375810)

[4. HEURISTIC 4: MINIMIZING LOSING CHANCES 3](#_Toc58375811)

[5. HEURISTIC 5: CHANCES HEURISTICS 3](#_Toc58375812)

[6. HEURISTIC 6: WEIGHTED CHANCES HEURISTICS 4](#_Toc58375813)

[7. HEURISTIC 7: WEIGHTED CHANCES HEURISTICS 2 4](#_Toc58375814)

[8. HEURISTICS 8: CUSTOM\_SCORE 4](#_Toc58375815)

[EVALUATING HEURISTICS 5](#_Toc58375816)

[RESULTS 6](#_Toc58375817)

[APPENDICES 8](#_Toc58375818)

# INTRODUCTIONS

This project aims to create a heuristics function as *evaluation functions* to calculate win conditions so that the player agent will have competency in playing the *Isolation* game and being able to compete with existing algorithms such as *Minimax* and *alpha-beta* pruning

**Isolation** is a deterministic, two-player game of perfect information in which the players alternate turns moving a single piece from one cell to another on a board. Whenever either player occupies a cell, that cell becomes blocked for the remainder of the game. The first player with no remaining legal moves loses, and the opponent is declared the winner.

A **heuristic** is a technique designed for solving a problem more quickly when classic methods are too slow, or for finding an approximate solution when classic methods fail to find any exact solution. In exchange of flexibility, heuristic provides a generalized solution for a specific problem with a faster run time.

To test the functions, we’ll have the player agents equipped with these heuristics function to compete against other agents in a tournament of *Isolation*. This project uses a version of Isolation where each agent is restricted to L-shaped movements (like a knight in chess) on a rectangular grid (like a chess or checkerboard). The agents can move to any open cell on the board that is 2- rows and 1-column or 2-columns and 1-row away from their current position on the board. When either player occupies a cell, that particular cell will became blocked for the rest of the game. Movements are blocked at the edges of the board (the board does not wrap around), however, the player can "jump" blocked or occupied spaces (just like a knight in chess). The first player with no available legal moves left is declared the loser and those who managed to make them so is declared the winner

# CUSTOM HEURISTICS

## HEURISTIC 1: AGRESSIVE HEURISTICS

The heuristic is based on the logic that opponent’s moves should be minimized. It can be mathematically expressed as:

l𝑒𝑛(𝑚𝑦 𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒𝑠) − 𝛼 𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠), 𝑤ℎ𝑒𝑟𝑒 α ∈ (1, ∞)

The value of 𝛼 was empirically chosen as 1.5.

## HEURISTIC 2: DEFENSIVE HEURISTICS

The heuristic is based on the logic that player’s moves should be maximized. It can be mathematically expressed as:

α 𝑙𝑒𝑛(𝑚𝑦 𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒𝑠) − 𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠), 𝑤ℎ𝑒𝑟𝑒 𝛼 ∈ (1, ∞)

The value of 𝛼 was empirically chosen as 1.5.

## HEURISTIC 3: MAXIMIZING WIN CHANCES

The heuristic is based on the logic that player should have more moves in comparison to opponent. It can be mathematically expressed as:

𝑙𝑒𝑛(𝑚𝑦 𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒𝑠)

𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠)

## HEURISTIC 4: MINIMIZING LOSING CHANCES

The heuristic is based on the logic that opponent should have less moves in comparison to player. It can be mathematically expressed as:

𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠)

− 𝑙𝑒𝑛(𝑚𝑦 𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒𝑠)

## HEURISTIC 5: CHANCES HEURISTICS

Can be mathematically expressed as:

𝑙𝑒𝑛(𝑚𝑦 𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒𝑠)

𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠) − -

𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠)

𝑙𝑒𝑛(𝑚𝑦 𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒𝑠)

Maximizing above equation is equivalent to maximizing:

[𝑙𝑒𝑛(𝑚𝑦 𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒𝑠)]2 − [𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠)]2

The latter form has been implemented in the code.

## HEURISTIC 6: WEIGHTED CHANCES HEURISTICS

Can be mathematically expressed as:

𝑙𝑒𝑛(𝑚𝑦 𝑎v𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒𝑠)

𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠) − 𝛼

𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠)

𝑙𝑒𝑛(𝑚𝑦 𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒𝑠) , 𝑤ℎ𝑒𝑟𝑒 𝛼 ∈ (1, ∞)

Maximizing above equation is equivalent to maximizing:

[𝑙e𝑛(𝑚𝑦 𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒𝑠)]2 − 𝛽 [𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠)]2, 𝑤ℎ𝑒𝑟𝑒 𝛽 ∈ (1, ∞)

The latter form has been implemented in the code with 𝛽 chosen as 1.5 empirically.

## HEURISTIC 7: WEIGHTED CHANCES HEURISTICS 2

Can be mathematically expressed as:

𝑙𝑒𝑛(𝑚𝑦 𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒s)

α 𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠) −

𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠)

𝑙𝑒𝑛(𝑚𝑦 𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒𝑠) , 𝑤ℎ𝑒𝑟𝑒 𝛼 ∈ (1, ∞)

Maximizing above equation is equivalent to maximizing:

β[𝑙𝑒𝑛(𝑚𝑦 𝑎v𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒𝑠)]2 − [𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠)]2, 𝑤ℎ𝑒𝑟𝑒 𝛽 ∈ (1 , ∞)

The latter form has been implemented in the code with 𝛽 chosen as 1.5 empirically.

1. HEURISTICS 8: CUSTOM\_SCORE

By combining method 2 and 3

α 𝑙𝑒𝑛(𝑚𝑦 𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑚𝑜𝑣𝑒𝑠)

𝑙𝑒𝑛(𝑎𝑣𝑎𝑖𝑙𝑎𝑏𝑙𝑒 𝑜𝑝𝑝𝑜𝑛𝑒𝑛𝑡 𝑚𝑜𝑣𝑒𝑠)

# EVALUATING HEURISTICS

The tournament.py script is used to evaluate the performance of the heuristic. The efficiency of the heuristics is tested as a competition against 7 opponents. 3 opponents use the minimax algorithm with a depth of 3, 3 α − β pruning with depth of 5, and a random opponent that only chooses random positions

The tournament opponents are listed below:

* + Random: An agent that randomly chooses a move each turn.
  + MM\_Null: CustomPlayer agent using fixed-depth minimax search and the null\_score heuristic
  + MM\_Open: CustomPlayer agent using fixed-depth minimax search and the open\_move\_score heuristic
  + MM\_Improved: CustomPlayer agent using fixed-depth minimax search and the improved\_score heuristic
  + AB\_Null: CustomPlayer agent using fixed-depth alpha-beta search and the null\_score heuristic
  + AB\_Open: CustomPlayer agent using fixed-depth alpha-beta search and the open\_move\_score heuristic
  + AB\_Improved: CustomPlayer agent using fixed-depth alpha-beta search and the improved\_score heuristic
  + ID\_Improved: CustomPlayer agent using iterative alpha-beta search and the improved\_score heuristic
  + Student1: CustomPlayer agent using iterative alpha-beta search and the heuristic 1
  + Student2: CustomPlayer agent using iterative alpha-beta search and the heuristic 2
  + Student3: CustomPlayer agent using iterative alpha-beta search and the heuristic 3
  + Student4: CustomPlayer agent using iterative alpha-beta search and the heuristic 4
  + Student5: CustomPlayer agent using iterative alpha-beta search and the heuristic 5
  + Student6: CustomPlayer agent using iterative alpha-beta search and the heuristic 6
  + Student7: CustomPlayer agent using iterative alpha-beta search and the heuristic 7
  + Me :CustomPlayer agent using iterative alpha-beta search and the heuristic 8

To fasten the tournament’s runtime, the code was ran in *google colab* which fasten the runtime by using the GPU as a hardware accelerator and lowering the number of matches to 3.

# RESULTS

The performance of various agents is as follow:

|  |  |  |
| --- | --- | --- |
| **Agent** | **Performance** | **Rank** |
| **ID\_Improved** | 59.52% | 7 |
| **Student1** | 61.90% | 5 |
| **Student2** | 71.43% | 1 |
| **Student3** | 64.29% | 4 |
| **Student4** | 70.24% | 2 |
| **Student5** | 66.67% | 3 |
| **Student6** | 64.29% | 4 |
| **Student7** | 63.10% | 6 |
| **Me** | 70.24% | 2 |

All the custom heuristics perform better than ID\_Improved by a reasonable margin as can be seen in the above table.

* Student2 has the most wins, with 10/2 over random and a 12/0 against MM\_Null
  + Despite this, Student2 still lost the match against MM\_Open with a score of 4/8
* MM\_Open has the most successful result against the heuristics function
  + 7/5 against Student1 and Student3
  + 8/4 against Student2 and Student6
  + A 6/6 tie with Student7
* Despite having the least wins, ID\_Improved managed to win against MM\_Open with a score of 7/5 in comparison to Student2, who has the most wins, losing to MM\_Open
* The custom\_score has the most wins against Random with the perfect score of 12/0
  + Despite this, it loses the match to MM\_Improved with a score of 5/7
    - MM\_Improved also managed to get a tie of 6/6 with Student1, Student3, and Student7

Another test was conducted, this time with increasing the number of matches to 10

|  |  |  |
| --- | --- | --- |
| **Agent** | **Performance** | **Rank** |
| **ID\_Improved** | 63.57% | 6 |
| **Student1** | 63.21% | 7 |
| **Student2** | 69.29% | 2 |
| **Student3** | 60.00% | 8 |
| **Student4** | 71.07% | 1 |
| **Student5** | 63.21% | 7 |
| **Student6** | 68.93% | 3 |
| **Student7** | 64.64% | 5 |
| **Me** | 68.21% | 4 |

Notable findings from the second tournament:

* ID\_Improved managed to catch up with some of the heuristics function
* All the functions won against the opponent
* Student6 has the most gain in the longer the number of matches is with an improvement of 4,64%
* Student4 has been the most consistent the longer the match goes on

The raw evaluation result can be found in [A. Appendix: Evaluation Result.](#_bookmark12)

# APPENDICES

Tournament 1:

This script evaluates the performance of the custom heuristic function by

comparing the strength of an agent using iterative deepening (ID) search with

alpha-beta pruning against the strength rating of agents using other heuristic

functions. The `ID\_Improved` agent provides a baseline by measuring the

performance of a basic agent using Iterative Deepening and the "improved"

heuristic (from lecture) on your hardware. The `Student` agent then measures

the performance of Iterative Deepening and the custom heuristic against the

same opponents.

\*\*\*\*\*\*\*\*\*

Evaluating: ID\_Improved

\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: ID\_Improved vs Random Result: 10 to 2

Match 2: ID\_Improved vs MM\_Null Result: 7 to 5

Match 3: ID\_Improved vs MM\_Open Result: 7 to 5

Match 4: ID\_Improved vs MM\_Improved Result: 7 to 5

Match 5: ID\_Improved vs AB\_Null Result: 6 to 6

Match 6: ID\_Improved vs AB\_Open Result: 6 to 6

Match 7: ID\_Improved vs AB\_Improved Result: 7 to 5

Results:

----------

ID\_Improved 59.52%

\*\*\*\*\*\*\*\*\*

Evaluating: Student1

\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student1 vs Random Result: 10 to 2

Match 2: Student1 vs MM\_Null Result: 9 to 3

Match 3: Student1 vs MM\_Open Result: 5 to 7

Match 4: Student1 vs MM\_Improved Result: 6 to 6

Match 5: Student1 vs AB\_Null Result: 8 to 4

Match 6: Student1 vs AB\_Open Result: 7 to 5

Match 7: Student1 vs AB\_Improved Result: 7 to 5

Results:

----------

Student1 61.90%

\*\*\*\*\*\*\*\*\*

Evaluating: Student2

\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student2 vs Random Result: 10 to 2

Match 2: Student2 vs MM\_Null Result: 12 to 0

Match 3: Student2 vs MM\_Open Result: 4 to 8

Match 4: Student2 vs MM\_Improved Result: 8 to 4

Match 5: Student2 vs AB\_Null Result: 8 to 4

Match 6: Student2 vs AB\_Open Result: 10 to 2

Match 7: Student2 vs AB\_Improved Result: 8 to 4

Results:

----------

Student2 71.43%

\*\*\*\*\*\*\*\*\*

Evaluating: Student3

\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student3 vs Random Result: 11 to 1

Match 2: Student3 vs MM\_Null Result: 9 to 3

Match 3: Student3 vs MM\_Open Result: 5 to 7

Match 4: Student3 vs MM\_Improved Result: 6 to 6

Match 5: Student3 vs AB\_Null Result: 7 to 5

Match 6: Student3 vs AB\_Open Result: 10 to 2

Match 7: Student3 vs AB\_Improved Result: 6 to 6

Results:

----------

Student3 64.29%

\*\*\*\*\*\*\*\*\*

Evaluating: Student4

\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student4 vs Random Result: 11 to 1

Match 2: Student4 vs MM\_Null Result: 9 to 3

Match 3: Student4 vs MM\_Open Result: 8 to 4

Match 4: Student4 vs MM\_Improved Result: 10 to 2

Match 5: Student4 vs AB\_Null Result: 9 to 3

Match 6: Student4 vs AB\_Open Result: 6 to 6

Match 7: Student4 vs AB\_Improved Result: 6 to 6

Results:

----------

Student4 70.24%

\*\*\*\*\*\*\*\*\*

Evaluating: Student5

\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student5 vs Random Result: 9 to 3

Match 2: Student5 vs MM\_Null Result: 9 to 3

Match 3: Student5 vs MM\_Open Result: 8 to 4

Match 4: Student5 vs MM\_Improved Result: 8 to 4

Match 5: Student5 vs AB\_Null Result: 7 to 5

Match 6: Student5 vs AB\_Open Result: 8 to 4

Match 7: Student5 vs AB\_Improved Result: 7 to 5

Results:

----------

Student5 66.67%

\*\*\*\*\*\*\*\*\*

Evaluating: Student6

\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student6 vs Random Result: 11 to 1

Match 2: Student6 vs MM\_Null Result: 7 to 5

Match 3: Student6 vs MM\_Open Result: 4 to 8

Match 4: Student6 vs MM\_Improved Result: 8 to 4

Match 5: Student6 vs AB\_Null Result: 8 to 4

Match 6: Student6 vs AB\_Open Result: 9 to 3

Match 7: Student6 vs AB\_Improved Result: 7 to 5

Results:

----------

Student6 64.29%

\*\*\*\*\*\*\*\*\*

Evaluating: Student7

\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student7 vs Random Result: 10 to 2

Match 2: Student7 vs MM\_Null Result: 8 to 4

Match 3: Student7 vs MM\_Open Result: 6 to 6

Match 4: Student7 vs MM\_Improved Result: 6 to 6

Match 5: Student7 vs AB\_Null Result: 10 to 2

Match 6: Student7 vs AB\_Open Result: 7 to 5

Match 7: Student7 vs AB\_Improved Result: 6 to 6

Results:

----------

Student7 63.10%

\*\*\*\*\*\*\*\*\*

Evaluating: Me

\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Me vs Random Result: 12 to 0

Match 2: Me vs MM\_Null Result: 9 to 3

Match 3: Me vs MM\_Open Result: 7 to 5

Match 4: Me vs MM\_Improved Result: 5 to 7

Match 5: Me vs AB\_Null Result: 11 to 1

Match 6: Me vs AB\_Open Result: 8 to 4

Match 7: Me vs AB\_Improved Result: 7 to 5

Results:

----------

Me 70.24%

Tournament 2:

This script evaluates the performance of the custom heuristic function by

comparing the strength of an agent using iterative deepening (ID) search with

alpha-beta pruning against the strength rating of agents using other heuristic

functions. The `ID\_Improved` agent provides a baseline by measuring the

performance of a basic agent using Iterative Deepening and the "improved"

heuristic (from lecture) on your hardware. The `Student` agent then measures

the performance of Iterative Deepening and the custom heuristic against the

same opponents.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Evaluating: ID\_Improved

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: ID\_Improved vs Random Result: 35 to 5

Match 2: ID\_Improved vs MM\_Null Result: 29 to 11

Match 3: ID\_Improved vs MM\_Open Result: 22 to 18

Match 4: ID\_Improved vs MM\_Improved Result: 20 to 20

Match 5: ID\_Improved vs AB\_Null Result: 27 to 13

Match 6: ID\_Improved vs AB\_Open Result: 22 to 18

Match 7: ID\_Improved vs AB\_Improved Result: 23 to 17

Results:

----------

ID\_Improved 63.57%

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Evaluating: Student1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student1 vs Random Result: 35 to 5

Match 2: Student1 vs MM\_Null Result: 27 to 13

Match 3: Student1 vs MM\_Open Result: 22 to 18

Match 4: Student1 vs MM\_Improved Result: 21 to 19

Match 5: Student1 vs AB\_Null Result: 25 to 15

Match 6: Student1 vs AB\_Open Result: 24 to 16

Match 7: Student1 vs AB\_Improved Result: 23 to 17

Results:

----------

Student1 63.21%

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Evaluating: Student2

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student2 vs Random Result: 34 to 6

Match 2: Student2 vs MM\_Null Result: 32 to 8

Match 3: Student2 vs MM\_Open Result: 22 to 18

Match 4: Student2 vs MM\_Improved Result: 21 to 19

Match 5: Student2 vs AB\_Null Result: 30 to 10

Match 6: Student2 vs AB\_Open Result: 29 to 11

Match 7: Student2 vs AB\_Improved Result: 26 to 14

Results:

----------

Student2 69.29%

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Evaluating: Student3

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student3 vs Random Result: 31 to 9

Match 2: Student3 vs MM\_Null Result: 29 to 11

Match 3: Student3 vs MM\_Open Result: 19 to 21

Match 4: Student3 vs MM\_Improved Result: 16 to 24

Match 5: Student3 vs AB\_Null Result: 26 to 14

Match 6: Student3 vs AB\_Open Result: 25 to 15

Match 7: Student3 vs AB\_Improved Result: 22 to 18

Results:

----------

Student3 60.00%

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Evaluating: Student4

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student4 vs Random Result: 37 to 3

Match 2: Student4 vs MM\_Null Result: 31 to 9

Match 3: Student4 vs MM\_Open Result: 28 to 12

Match 4: Student4 vs MM\_Improved Result: 21 to 19

Match 5: Student4 vs AB\_Null Result: 28 to 12

Match 6: Student4 vs AB\_Open Result: 28 to 12

Match 7: Student4 vs AB\_Improved Result: 26 to 14

Results:

----------

Student4 71.07%

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Evaluating: Student5

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student5 vs Random Result: 34 to 6

Match 2: Student5 vs MM\_Null Result: 31 to 9

Match 3: Student5 vs MM\_Open Result: 23 to 17

Match 4: Student5 vs MM\_Improved Result: 16 to 24

Match 5: Student5 vs AB\_Null Result: 24 to 16

Match 6: Student5 vs AB\_Open Result: 25 to 15

Match 7: Student5 vs AB\_Improved Result: 24 to 16

Results:

----------

Student5 63.21%

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Evaluating: Student6

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student6 vs Random Result: 37 to 3

Match 2: Student6 vs MM\_Null Result: 30 to 10

Match 3: Student6 vs MM\_Open Result: 23 to 17

Match 4: Student6 vs MM\_Improved Result: 19 to 21

Match 5: Student6 vs AB\_Null Result: 26 to 14

Match 6: Student6 vs AB\_Open Result: 28 to 12

Match 7: Student6 vs AB\_Improved Result: 30 to 10

Results:

----------

Student6 68.93%

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Evaluating: Student7

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Student7 vs Random Result: 31 to 9

Match 2: Student7 vs MM\_Null Result: 27 to 13

Match 3: Student7 vs MM\_Open Result: 22 to 18

Match 4: Student7 vs MM\_Improved Result: 21 to 19

Match 5: Student7 vs AB\_Null Result: 27 to 13

Match 6: Student7 vs AB\_Open Result: 30 to 10

Match 7: Student7 vs AB\_Improved Result: 23 to 17

Results:

----------

Student7 64.64%

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Evaluating: Me

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Playing Matches:

----------

Match 1: Me vs Random Result: 33 to 7

Match 2: Me vs MM\_Null Result: 30 to 10

Match 3: Me vs MM\_Open Result: 25 to 15

Match 4: Me vs MM\_Improved Result: 21 to 19

Match 5: Me vs AB\_Null Result: 32 to 8

Match 6: Me vs AB\_Open Result: 25 to 15

Match 7: Me vs AB\_Improved Result: 25 to 15

Results:

----------

Me 68.21%