Alphabeta Player depth 1 0.0 0.0 10.0 0.0 0.0 0.0 10.0 0.0 0.0 0.5 1.0 1.0 Total: 22.5

Alphabeta Player depth 2 20.0 0.0 0.0 10.0 0.0 0.0 0.0 3.5 1.5 3.0 1.0 2.0 Total: 41.0

Alphabeta Player depth 3 10.0 20.0 0.0 10.0 10.0 10.0 20.0 4.0 4.0 2.0 0.0 5.0 Total: 95.0

Alphabeta Player depth 4 20.0 10.0 10.0 0.0 20.0 20.0 0.0 2.5 2.0 3.0 2.0 2.5 Total: 92.0

Alphabeta Player depth 5 20.0 20.0 10.0 0.0 0.0 0.0 10.0 7.0 3.0 7.0 3.0 5.0 Total: 85.0

Alphabeta Player depth 6 20.0 20.0 10.0 0.0 20.0 0.0 10.0 6.5 5.0 7.0 3.5 5.0 Total: 107.0

Alphabeta Player depth 7 10.0 20.0 0.0 20.0 10.0 10.0 0.0 7.0 5.0 8.5 9.0 9.5 Total: 109.0

Monte Carlo 100 20.0 16.5 16.0 17.5 13.0 13.5 13.0 0.0 11.0 11.0 10.0 7.0 Total: 148.5

Monte Carlo 200 20.0 18.5 16.0 18.0 17.0 15.0 15.0 9.0 0.0 9.5 5.5 7.5 Total: 151.0

Monte Carlo 300 19.5 17.0 18.0 17.0 13.0 13.0 11.5 9.0 10.5 0.0 6.5 11.0 Total: 146.0

Monte Carlo 400 19.0 19.0 20.0 18.0 17.0 16.5 11.0 10.0 14.5 13.5 0.0 11.0 Total: 169.5

Monte Carlo 500 19.0 18.0 15.0 17.5 15.0 15.0 10.5 13.0 12.5 9.0 9.0 0.0 Total: 153.5

So, we can see from our data that Monte Carlo outperforms Alphabeta for most cases, although, due to the randomness, it sometimes falters against the deeper depth AB players but sometimes comes out even or ahead. One thing we cannot explain is why the Monte Carlo player does not appear to get significantly better with an increased number of playouts. This indicates there is a bug in our code, but we could not find it before the deadline for this submission. There are also a few strange cases, for instance where Monte Carlo 100 beats Alpha Beta 7 100% of the time. This seems like it isn’t chance, and could be related to the other possible bug.