N-dimensional Tic Tac Toe, and Adventure in Modules

Alex Grasley Jeff Young Michael McGirr

1 Introduction

2 Overview of Project

Our project initially began as a pokemon simulator but early on we realized that we could make better use of the SML module system by approaching the problem of simulating player-based games in a more abstract way. By doing this we could use the basic notion of what a game simulation required and isolate a pattern to follow for any number of games that fit this adversarial model. A game would then be a specific implementation - in our case tic tac toe - that used this pattern.

3 Program description

Our project is separated between the code that describes the game simulation and the code that used this to make a specific tic tac toe implementation. The game simulation is located in the `game.sml` file. The code for tic tac toe implementation is spread out between `tictactoe.sml` and `matrix.sml`.

4 Design Decisions

4.1 Creating an Abstract Game Engine

4.2 Higher Ordered Signatures, and the "Include" incantation

4.3 Separation of IO, or How I learned to not fight SML in search of Purity

4.4 You can do it in 2-dimensions, but can you do it in n-dimensions!

4.5 The Functor is love, the Functor is life