GamesCrafters

1,2,...10

Dan Garcia(Dev)

Documentation for Developers

Fall 2007: Michael So

Date	Version	Notes
2007.10.14	1.0.0	First version created
2007.12.4	1.0.1	First revision

Table of Contents

Game Overview

1,2,…,10 is a two-player game with that uses a simple board with 10 slots. Each player, using their own piece(s), move one or two spaces on their turn. The player's goal is to be the first player to reach the tenth slot.

The game is non-partisan, non-loopy, and a tie is not possible.

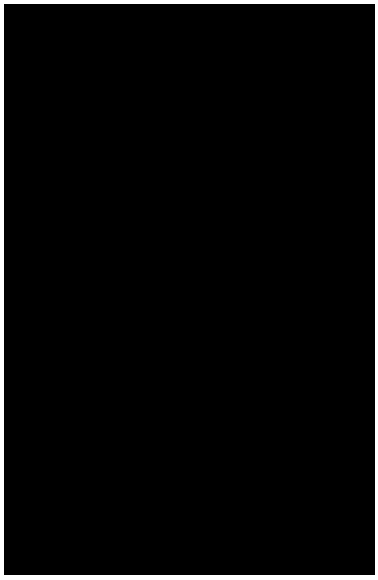

Figure 1 The 1,2,…,10 game board

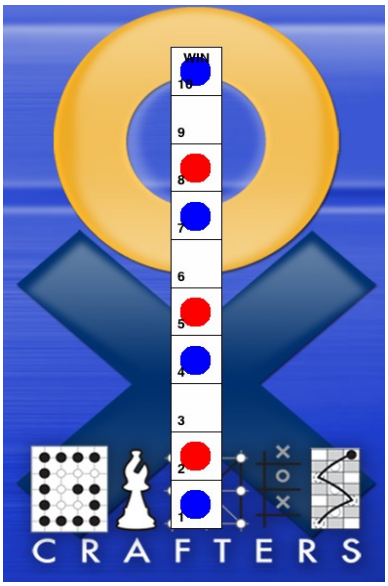

Figure 2 Winning board for blue piece

Design Overview

There are no defines, structs, or global variables.

Taking in a parameter called position generates moves. Positions take the value of an integer from 0 to 10. If position is less than 9, it means the player can move 1 or 2 places. Otherwise, it means the player is at position 9 and can only move 1 place to position 10.

Win is confirmed if player is at position 10. Lose is confirmed if the other player reached position 10 and is therefore in a win state. Besides the win and lose state is the undecided state. There is no tie state.

Data Structures

The only data structure is the board, which consists of 11 positions numbered from 0 to 10. Position 0 corresponds to being not on the board, position 1 corresponds to being in the first slot of the board, and so on until position 10 which corresponds to last slot of the board.

0	1	2	3	4	5	6	7	8	9	10

C Game Checklist

- ✔ All help strings are done.
- ✔ Debug turned off
- ✔ There are no variants.
- ✔ No memory leaks
- ✔ PrintPosition done
- ✔ getoption, setoption, and option stuff is done
- ✔ Placed in Makefile
- ✔ Clean/commented code
- ✔ Implementation of MoveToString() is done
- ✔ Calls the common functions from the Game function libraries
  - GenerateMoves calls library function CreateMovelistNode().
  - PrintPosition calls library function GetPrediction().
- ✔ symmetries is not used.
- ✔ GPS is not used.

Complexity Analysis

```
POSITION gNumberOfPositions  = 11;      /*Every board from 0 to 10*/
POSITION gInitialPosition    = 0;
POSITION kBadPosition        = -1;
```

There are 11 possible positions in this game. Positions 1 to 10 are on the board whereas position 0 is not on the board and is considered the initial position.

Optimal Strategy

The optimal strategy is to make sure there is an odd number of moves left. The first player is guaranteed to win if the player plays perfectly.