

Team: Blackbird

Inspiration: Blokus

Blokus is played with a 20x20 square board. Each player has 21 pieces, all of different shapes. (polyominoes). The object of the game is for each player to put all 21 of his/her pieces on the board.

Rules: Each player starts at one corner of the square, and can only play pieces that are diagonally adjacent to their own pieces (no sides can touch).

Our game: Blokus 3D

Our version is played on the outside faces of a cube instead of on a square, and can have up to 4 players. The game wraps around the edges of the cube onto other faces, following the same rules as the original game. Each player starts on two opposite corners of the cube, and builds from there. The game ends when no one can place any more pieces on the board.

Points are tallied based on the number of squares covered by a player's pieces, with bonuses for achieving certain goals (like playing all of your pieces). Each turn has a set time limit (specified at the beginning of each game), or a speed-chess option where each player has a maximum total time for the whole game.

The controls available to a player are:

- Rotate the board about the center of the cube (right click and drag the cube display to spin it)
- Pick a piece (click on it)
- Rotate and flip the selected piece (either by clicking buttons or using keyboard shortcuts)
- Play the selected piece (by clicking on the cube in the location you want to place the piece). After a user has selected a piece and their mouse is hovering over the cube, a "shadow" of the piece will appear on the board under the mouse. If the piece can't be played in that position, a warning indicator will show up on an edge of the display telling the user why that spot is an illegal move.

A roll-back history (including an undo feature for single player games) allows the user to view the board at any state it has been in. This feature comes linked with a replay function. Furthermore, the current state of the game can be saved at any time and reloaded later, including multiplayer or online games. These features allow the player to either analyze past games, or share games with friends.

The game is a desktop application with the following options for adding other players:

- AI players of varying levels of difficulty.
- Other players in a hotseat environment.
- Remotely connecting to another user's computer. An in-game chat feature will be enabled for network games. Users will connect directly to their friends' computers by

IP address, rather than looking up and connecting to games through a central server.

An array of friends' names and IP addresses will be stored after the user puts them in. These will all be combinable - for instance, you could be playing a game against two friends across the web, who are playing on the same computer with a "hotseat" arrangement, and a computer, which is running on your local machine.

Each game will have a central host. To join a game, you can either accept an invitation or request approval from the host. When you join a game, you can grab a spot at the table as soon as you join. If an AI player is part of a network game, the AI player will always run on the machine that is hosting the game, and therefore an AI player can only be added by the user who is hosting the game. At any point in the game, if a user disconnects (purposely or unintentionally), an option will be presented to the host asking whether a computer player should fill in that spot until the possible return of the player. In general, it will be very easy to swap the source for a particular player in between moves. If the host gets disconnected or leaves, the other players will be asked whether they want to host the rest of the game or save or quit. At the end of a game, all users will keep their spots at the table, but the first move will shift monotonically in one direction.

AI players come in various levels: beginner, intermediate, advanced, and ridiculous.

Although we will have to play against the AI to see how well we match these categories, we plan to have the ridiculous AI calculate at least 15 moves in advance once the game is 5 moves in, and have a fairly advanced board evaluation method for alpha-beta pruning with limited depth. Because the number of pieces to play on the board is about 25 (depending on exactly how we create the game) and there are LOTS of possible moves at the beginning of the game, 15 moves in advance is plenty once the game gets past the opening moves. Every category of AI will map more moves in advance as the game goes on, since the number of possible moves gets significantly smaller at the end of the game. It is likely that we will use some function of the number of "building points" an AI player has and the number of pieces remaining to determine how deep to explore the tree.

There are a few training programs for players wishing to get better at the game:

- Targeting beginners, there are tips for individual scenarios.
- Also Targeting beginners, when playing games against AI, there is a hint button.
- More for intermediate-advanced users, you can have the computer tell you how good the moves you make are based on the AI algorithm for identifying good moves.
- Although this isn't exactly an exciting option, it is also possible to make all the players be AI. That way, a new user could examine their playing strategies.