

To: Matthias Felleisen
From: Gino Jacob and Emily Miller-McGlone
Subject: Santorini

A game of Santorini consists of two players matched according to number of games played and their victory rate (wins / total games). The player with the lower win rate places and moves their workers first. Otherwise, they are randomly selected.

Players take turns moving/building until a game end condition is met. The first player to reach the third level of a building wins the game. If a player can't move either of their workers or have their worker build after a move, the other player wins.

On a high level, the game software consists of the board and the players. The board should know: information on players (their id's, wins/total games, assignment of P1 & P2), game state, game rules, and keep track of whose turn it is. Players should know: game state, which workers are theirs, and whose turn it is. At the start of each turn, players get the game state from the board. During their turn, players communicate to the board about their move/build and the board completes the turn if it is valid. If the player gives an invalid move/build the board throws an error, communicating what was invalid. The board will then prompt the player for another move/build.

Game logic is implemented in the Board class. The board is a 2D zero-indexed list of (Cell, level: number) where Cell is either Floor, or Worker and level is the level of the building.

Floor is an empty class. Worker is a class which contains a name: Int, coordinates: (x, y), and player: Player. Player is a class consisting of a name: unique id, number of wins: int, total played games: int. The player has the method prompt(list of list, worker 1 position, worker 2 position) to get the move/build from the player and except(exception, list of list) to get a new move/build if the previous move/build was invalid. The method prompt returns the player's worker number they want to move, an enum: direction, and enum: build position. The method except returns the player's worker number they want to move, an enum: direction, and enum: build position.

Exceptions are: blocking worker, floor unreachabeable, floor four, coordinates out of bounds, standing on coordinates. The direction and build position enumeration has members: N, S, W, E, NW, NE, SW, and SE.