Froglet manual

Necessary Actions

The program only needs the SICSTUS Prolog server opened and running by consulting the server.pl file and then running "server.". The usual local web server is needed for running the WebGL portion that connects to the server. It can be reached at "localhost:port", no special parameters are needed, a default LSX file is already chosen.

Main Rules

The game consists of a 12x12 board with a total of 144 frogs, each frog is worth a different amount of points and there also is a fixed number of each color. The game starts by removing any green frog from the board, this is always done by the first player by game rules.

After picking a green frog the game enters its loop, consisting of jumps like Checkers. A valid jump must be horizontally or vertically, over a frog and to an empty cell without jumping over more than one frog. Like Checkers, the game also features multiple jumping, however, it is not obligatory like in Checker, the player can choose whether to perform the jump or skip his turn at any phase of multiple jumping. The game ends when no more jumps can be performed, and victor is declared.

User Instructions

DAT GUI

The "New Game" folder contains all options available to customize the next game started, if a game is already midway, changes do not take effect until next game reset.

Game mode: the type of players (Human or AI)

AI # Difficulty: selects the difficulty of AI players defined in "Game mode".

Turn limit (s): time available to make move if Human player, time to automatic movement if AI player.

Undo past own turn?: choose whether undo controls the flow of the game or if undo is locked up to current player's turn.

The "Movie" folder contains controls for movie playback / stopping.

Play Movie: plays movie up to current move or whole game if done after a game over. The button updates automatically to inform the user whether this action is available.

Stop Movie: stops movie playback if movie is playing. Like "Play Movie" button visually updates to inform user of availability.

The "Froglet" folder contains scene and game options that can be changed almost at any time.

Current scene: the current LSX file being displayed, can be changed at any time.

Current viewpoint: the viewpoint being displayed. There's a "free" camera mode with mouse control, a "rotating" camera that follows the current player's side, and finally a "fixed" camera mode that overviews the board. Viewpoint can be changed at any time that a viewpoint change animation isn't being played.

Frog animation speed: the hop and jump speed of the board frogs.

Viewpoint anim speed: the speed at which viewpoint changing occurs. Doesn't affect the "rotating" camera animation.

Rotating camera anim?: whether to animate the "rotating" camera. If turned off, camera changes instantly.

Frog animation?: whether frogs should animate.

Use cube frogs?: whether to display frogs as cubes to lower performance impact of the custom frogs.

Paused?: whether the game is paused. Can't be toggled on viewpoint switching.

Do AI Move: confirm AI move so it requests a move from the server. Button is updated automatically to inform the user of availability.

AI auto play?: whether AI moves should require no confirmation from user.

The "Lights" folder allows toggling scene lights.

Play/Undo Menu

All the buttons on the picking menu flash green if action is allowed and red if not allowed.

Play: starts a new game based on "New Game" folder options.

Undo: undo previous move, affected by "Undo past own turn?" checkbox.

Jump Again Menu

Yes: confirm multiple jump intention.

No: pass on multiple jump opportunity.

Keyboard Keys

c/C: confirms AI move.

m/M: toggles movie.

p/P: toggles pause.

s/S: starts new game.

u/U: undoes last move.

v/V: cycles viewpoints. Can't be used if viewpoint is already switching.

Mouse Gameplay

The first move is done by removing a green frog simply by selecting its cell. Afterwards, any frog can be selected, and a jump attempted by selecting an empty cell, appropriate feedback is given. If player switches his mind, another frog can be selected before selecting an empty cell. On a multiple jump opportunity, the "Jump Again" menu appears, and the user can confirm or pass on the opportunity. However, besides these two buttons, the user can also select the same frog to cancel the multiple jump or select an empty cell to confirm it, thus obliviating the need for the menu.