AI class Project

Generated by Doxygen 1.8.7

Thu Sep 4 2014 15:13:51

Contents

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

Chapter 2

File Documentation

2.1 /home/sweetness/Documents/AI-git/implementation/AI/classifier.h File Reference

Functions

- int classifier_init ()
- int classifier_make_move (int *xyarray)
- int classifier_prune (int prune)
- int classifier_free ()
- int classifier_depth (int depth)

2.1.1 Function Documentation

2.1.1.1 int classifier_depth (int *depth*)

Set the depth of search if used

Parameters

depth	the depth to search

Returns

0 for success

2.1.1.2 int classifier_free ( )

Free up memory allocated for the classifier

Returns

0 for success

2.1.1.3 int classifier_init ( )

Author

Kaleb, Jacob, Angelica

This file is part of Artificially Intelligent Polar Tic Tac Toe game.

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with AI Polar Tic Tac Toe.

If not, see http://www.gnu.org/licenses/. Initialize the classifier

Returns

returns 0 on success and 1 on fail

2.1.1.4 int classifier_make_move (int * *xyarray*)

Make a move with the classifier

Parameters

xyarray	the x and y integer for placement

Returns

0 for success

2.1.1.5 int classifier_prune (int *prune*)

Set option for potential prunning

Parameters

prune	a boolean flag to set prunning

Returns

0 for success

2.2 /home/sweetness/Documents/AI-git/implementation/AI/heur.h File Reference

Functions

- int heur_init ()
- int heur_make_move (int *xyarray)
- int heur_prune (int prune)
- int heur_free ()
- int heur_depth (int depth)

2.2.1 Function Documentation

2.2.1.1 int heur_depth (int *depth*)

Set the depth of search if used

Parameters

depth	the depth to search

Returns

0 for success

2.2.1.2 int heur_free ( )

Free up memory allocated for standard heuristic

Returns

0 for success

2.2.1.3 int heur_init ( )

Copyright 2014

Author

Kaleb, Jacob, Angelica

This file is part of Artificially Intelligent Polar Tic Tac Toe game.

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with AI Polar Tic Tac Toe.

If not, see http://www.gnu.org/licenses/. Initialize the standard heuristic

Returns

returns 0 on success and 1 on fail

2.2.1.4 int heur_make_move (int * *xyarray*)

Make a move with the standard heuristic

Parameters

xyarray	the x and y integer for placement

Returns

0 for success

2.2.1.5 int heur_prune (int *prune*)

Set option for potential prunning

Parameters

prune	a boolean flag to set prunning

Returns

0 for success

2.3 /home/sweetness/Documents/AI-git/implementation/AI/TDNN.h File Reference

Functions

- int TDNN_init ()
- int TDNN_make_move (int *xyarray)
- int TDNN_prune (int prune)
- int TDNN_free ()
- int TDNN_depth (int depth)

2.3.1 Function Documentation

2.3.1.1 int TDNN_depth (int *depth*)

Set the depth of search if used

Parameters

depth	the depth to search

Returns

0 for success

2.3.1.2 int TDNN_free ( )

Free up memory allocated for neural network

Returns

0 for success

2.3.1.3 int TDNN_init ( )

Copyright 2014

Author

Kaleb, Jacob, Angelica

This file is part of Artificially Intelligent Polar Tic Tac Toe game.

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with AI Polar Tic Tac Toe.

If not, see http://www.gnu.org/licenses/. Initialize the neural network

Returns

returns 0 on success and 1 on fail

2.3.1.4 int TDNN_make_move (int * *xyarray*)

Make a move with the neural network

Parameters

xyarray	the x and y integer for placement

Returns

0 for success

2.3.1.5 int TDNN_prune (int *prune*)

Set option for potential prunning

Parameters

prune	a boolean flag to set prunning

Returns

0 for success

2.4 /home/sweetness/Documents/AI-git/implementation/board_state.c File Reference

2.5 /home/sweetness/Documents/AI-git/implementation/board_state.h File Reference

Functions

- void get_state (int **array, int *height, int *width)

2.5.1 Function Documentation

2.5.1.1 void get_state (int ** *array*, int * *height*, int * *width*)

Gets the current positions of pieces on the board

Parameters

array	a two demensional array with board locations
height	sets the height of array [height][]
width	sets the width of array [][width]

2.6 /home/sweetness/Documents/AI-git/implementation/winchecker.c File Reference

```
#include <stdio.h>
#include <stdbool.h>
#include "winchecker.h"
```

Functions

- bool check_win (char cPlayerTurn, char cPlayerBoardLoc[ ][17])

2.6.1 Function Documentation

2.6.1.1 bool check_win (char *cPlayerTurn,* char *cPlayerBoardLoc[ ][17]*)

Function to check for a win

Parameters

cPlayerTurn	current player to check
cPlayerBoardLoc	

Returns

a boolean value of win or not

2.7 /home/sweetness/Documents/AI-git/implementation/winchecker.h File Reference

Functions

- bool check_win (char cPlayerTurn, char cPlayerBoardLoc[ ][17])

2.7.1 Function Documentation

2.7.1.1 bool check_win (char *cPlayerTurn,* char *cPlayerBoardLoc[ ][17]*)

Function to check for a win

Parameters

cPlayerTurn	current player to check
cPlayerBoardLoc	

Returns

a boolean value of win or not