

Dokumentace do projektu do matematického softwaru

PříHoDa

6. prosince 2024

Class: Board

T reprezentuj hracsku hry

Method: `_init_`

Konstruktor t Board

Method: `_populateBoard`

Inicializace hernsky

Method: `_str_`

Vytvoring hracsky na vpis do konzole

Method: `getListOfBoard`

Returns: List of Struct : List, kde kad k je list obsahuj figury na danu *Vracchovnici jako list*

Returns: List of Struct : List, kde kad k je list obsahuj figury na danu

Returns

List of Struct : List, kde kad k je list obsahuj figury na danu

Method: `_getitem_`

Args: index: Tuple dvou integer, (row, col)

Returns: Figuru na ur achovnici, pdne, pokud je prn

Pro monost pstupovat k poli board jako board[row,col] mo board.board[row][col]

Args: index: Tuple dvou integer, (row, col)

Returns: Figuru na ur achovnici, pdne, pokud je prn

Args

index: Tuple dvou integer, (row, col)

Returns

*Figuru na ur achovnici, pdnne, pokud je prn*Method: _setitem_*

Args: index ([int, int]): Tuple dvou integer, (row, col) value (any): Co se mstavít na danloNastavlo na achovnici jako board[row,col] namo board.board[row][col]

Args: index ([int, int]): Tuple dvou integer, (row, col) value (any): Co se mstavít na danlo

Args

index ([int, int]): Tuple dvou integer, (row, col) value (any): Co se mstavít na danlo

Class: ChallengeAccepted

Attribute: `_init_`

Attribute: `getBoard`

Attribute: `getField`

Attribute: `moveMole`

Attribute: `makeMove`

Attribute: `checkEnd`

Attribute: `printToTerminal`

Class: Checkers

T reprezentuj hru d

Method: `_init_`

Konstruktor t hry d

Method: `getBoard`

Args: color (Enum Colors, optional): barva hr na tahu. Vchozstaven na pravidelnReturns: List of List of [int, int]: hracskaFunkce pro z hracsky

Args: color (Enum Colors, optional): barva hr na tahu. Vchozstaven na pravidelnReturns: List of List of [int, int]: hracska

Args

color (Enum Colors, optional): barva hr na tahu. Vchozstaven na pravidelntn

Returns

List of List of [int, int]: hracska

Method: choosePiece

Args: index ([int,int]): pozice figurky, kterou chce hrac hrat color (Enum Colors, optional): barva hr na tahu. Vchozstaven na pravidelntnReturns: List of [int, int]: seznam dostupnych pozic, kam me hrr

Funkce pro vyber figurky, kterou chce hrac hrat

Args: index ([int,int]): pozice figurky, kterou chce hrac hrat color (Enum Colors, optional): barva hr na tahu. Vchozstaven na pravidelntnReturns: List of [int, int]: seznam dostupnych pozic, kam me hrr

Args

index ([int,int]): pozice figurky, kterou chce hrac hrat color (Enum Colors, optional): barva hr na tahu. Vchozstaven na pravidelntn

Returns

List of [int, int]: seznam dostupnych pozic, kam me hrr

Method: makeMove

Args: index ([int,int]): pozice, kam chce hrrolor (Enum Colors, optional): barva hr na tahu. Vchozstaven na pravidelntnightClick (bool, optional): True, pokud se jednravako myi. Vchozstaven False.

Returns: bool: True, pokud se tah podail, jinak FalseFunkce pro provedenhu figurkou

Args: index ([int,int]): pozice, kam chce hrrolor (Enum Colors, optional): barva hr na tahu. Vchozstaven na pravidelntnightClick (bool, optional): True, pokud se jednravako myi. Vchozstaven False.

Returns: bool: True, pokud se tah podail, jinak False

Args

index ([int,int]): pozice, kam chce hrrolor (Enum Colors, optional): barva hr na tahu. Vchozstaven na pravidelntnightClick (bool, optional): True, pokud se jednravako myi. Vchozstaven False.

Returns

bool: True, pokud se tah podail, jinak False

Method: reset

Funkce pro resetov hry

Method: checkEnd

Returns: string: Vract "barva won", pokud hra skona, jinak None
Funkce pro kontrolu konce hry

Returns: string: Vract "barva won", pokud hra skona, jinak None

Returns

string: Vract "barva won", pokud hra skona, jinak None

Method: _endOfTurn

Funkce pro ukonhu

Method: possibleMoves

Args: color (Enum Colors): barva hr, pro kter se majhy zat

Returns: List of [int, int]: seznam monch tah
Funkce pro z monch tah pro hr, primo rozng Of War

Args: color (Enum Colors): barva hr, pro kter se majhy zat

Returns: List of [int, int]: seznam monch tah

Args

color (Enum Colors): barva hr, pro kter se majhy zat

Returns

List of [int, int]: seznam monch tah

Method: killPiece

Args: piecePosition ([int, int]): pozice figurky, kter bt odstran
Funkce pro odstran figurky z hracsky

Args: piecePosition ([int, int]): pozice figurky, kter bt odstran

Args

piecePosition ([int, int]): pozice figurky, kter bt odstran

Method: _printToTerminal

Funkce pro vpis stavu hry na termin

Class: CheckersBoard

T reprezentuj achovnici pro hru d. D z t Board.

Method: `__init__`

Konstruktor t CheckersBoard. Vytvochovnici a um ni vechny figurky do poe polohy.

Method: `__getitem__`

Args: index [int, int]: Tuple dvou integer, (row, col), oba 0-7 Vracguru na ur achovnici, nebo None, pokud je polo prnArgs: index [int, int]: Tuple dvou integer, (row, col), oba 0-7

Args

index [int, int]: Tuple dvou integer, (row, col), oba 0-7

Method: `__setitem__`

Args: index ([int, int]): Tuple dvou integer, (row, col), oba 0-7 value (Piece): Instance t Piece, nebo None, pokud mt polo prnreturns: bool: True, pokud se podailo nastavit polo, jinak FalseNastavlo na achovnici jako board[row,col] namo board.board[row][col]

Args: index ([int, int]): Tuple dvou integer, (row, col), oba 0-7 value (Piece): Instance t Piece, nebo None, pokud mt polo prnreturns: bool: True, pokud se podailo nastavit polo, jinak False

Args

*index ([int, int]): Tuple dvou integer, (row, col), oba 0-7 value (Piece): Instance t Piece, nebo None, pokud mt polo prn*Returns bool: True, pokud se podailo nastavit polo, jinak False*

Method: `__str__`

Returns: string: string reprezentace achovnice Vracring reprezentaci achovnice. Kadlo je reprezentov jako string, kter je tvoen z informacarvymbolu figury, nebo jako string "_", pokud je polo prnola jsou oddna mezerou a jednotlivky jsou oddny znakem nov u ().

Returns: string: string reprezentace achovnice

Returns

string: string reprezentace achovnice

Method: _populateBoard

Nastavchovnici do pocho stavu. B je dole a n nahoe.

Method: pieceList

Args: color (Enum Colors): Barva figurek, ktermene najColors.WHITE nebo Colors.BLACK)

Returns: List of Pieces: List figurek danrvyVracst vech figurek danrvy na achovnici.

Args: color (Enum Colors): Barva figurek, ktermene najColors.WHITE nebo Colors.BLACK)

Returns: List of Pieces: List figurek danrvy

Args

color (Enum Colors): Barva figurek, ktermene najColors.WHITE nebo Colors.BLACK)

Returns

List of Pieces: List figurek danrvy

Method: getListOfBoard

Returns: List of Fields: List figurek na achovniciVracst vech figurek na achovnici.

Returns: List of Fields: List figurek na achovnici

Returns

List of Fields: List figurek na achovnici

Class: CheckersMines

Hra d s minami.

Method: _init_

Inicializace hry.

Method: _placeMines

Umny na ndnzice.

Method: getBoard

Vr hracku.

Method: makeMove

Zpracuje tah.

Class: CheckersMinesWithFogOfWar

T CheckersWithFogOfWar slouží k prezentaci hry D s mlhou vy.

Method: _init_

Konstruktor třídy CheckersWithFogOfWar

Method: getBoard

Args: color (Enum Colors): Barva hr na tahu

Returns: Array of Field: zakrytchovnice Vrácí krytou a chovnici

Args: color (Enum Colors): Barva hr na tahu

Returns: Array of Field: zakrytchovnice

Args

color (Enum Colors): Barva hr na tahu

Returns

Array of Field: zakrytchovnice

Class: Pawn

T reprezentuje pěška

Method: _str_

Returns: str: P + barva figurky Funkce na vypsání figurky

Returns: str: P + barva figurky

Returns

str: P + barva figurky

Method: possibleMoves

Args: board (two-dimensional array of ints): hracská

Returns: [int,int]: pole monch tah Funkce najde všechny možné figurky

Args: board (two-dimensional array of ints): hracská

Returns: [int,int]: pole monch tah

Args

board (two-dimensional array of ints): hracska

Returns

[int,int]: pole monch tah

Method: possibleJumps

Args: board (two-dimensional array of ints): hracska position ([int,int]), optional): `description.DefaultstoNone`.

*Returns: [int,int]: pole monch skok**Funkce najde vechny jednoduchoky z pozice*

Args: board (two-dimensional array of ints): hracska position ([int,int]), optional): `description.DefaultstoNone`.

Returns: [int,int]: pole monch skok

Args

board (two-dimensional array of ints): hracska position ([int,int]), optional): `description.DefaultstoNone`.

Returns

[int,int]: pole monch skok

Method: trackJumps

Args: endPosition ([int, int]): koncovzice

*Returns: [int, int]: pozice peskoegurky**Funkce na zjit peskoch figurek*

Args: endPosition ([int, int]): koncovzice

Returns: [int, int]: pozice peskoegurky

Args

endPosition ([int, int]): koncovzice

Returns

[int, int]: pozice peskoegurky

Class: Piece**Method: `_init_`**

*Args: color (Enum Colors): poadovanrva figurky position ([int,int]): vchoz-zice figurky**Konstruktor t Piece*

Args: color (Enum Colors): poadovanrva figurky position ([int,int]): vchoz-zice figurky

Args

color (Enum Colors): poadovanrva figurky position ([int,int]): vchozzice figurky

Method: possibleMoves

Args: board (Board): hracska, na kter figurka nachMetoda pro zjit monch tah figurky

Args: board (Board): hracska, na kter figurka nach

Args

board (Board): hracska, na kter figurka nach

Method: possibleJumps

Args: board (Board): hracska, na kter figurka nach position ([int, int], optional): pozice, ze kter mt. Vchozdnota je None, co znamene se sk ze sounzice figurkyMetoda pro zjit monch skok figurky

Args: board (Board): hracska, na kter figurka nach position ([int, int], optional): pozice, ze kter mt. Vchozdnota je None, co znamene se sk ze sounzice figurky

Args

board (Board): hracska, na kter figurka nach position ([int, int], optional): pozice, ze kter mt. Vchozdnota je None, co znamene se sk ze sounzice figurky

Method: trackJumps

Args: board (Board): hracska, na kter figurka nach endPosition ([int, int]): pozice, kam se mtMetoda pro zjit monch skok figurky

Args: board (Board): hracska, na kter figurka nach endPosition ([int, int]): pozice, kam se mt

Args

board (Board): hracska, na kter figurka nach endPosition ([int, int]): pozice, kam se mt

Method: row

Returns: int: k, na ktere figurka nachVlastnost pro z u, na ktere figurka nach

Returns: int: k, na ktere figurka nach

Returns

int: k, na ktorej figurka nach

Method: col

Returns: int: sloupec, na ktorej figurka nach Vlastnost pro z sloupce, na ktorej figurka nach

Returns: int: sloupec, na ktorej figurka nach

Returns

int: sloupec, na ktorej figurka nach

Class: Queen**Method: __init__**

Args: pawn (Pawn): Pk, kter se momt na dKonstruktor t Queen

Args: pawn (Pawn): Pk, kter se momt na d

Args

pawn (Pawn): Pk, kter se momt na d

Method: __str__

Returns: string: string reprezentace dVracring reprezentaci d

Returns: string: string reprezentace d

Returns

string: string reprezentace d

Method: possibleMoves

Args: board (CheckersBoard): achovnice, na kter d nach

Returns: List of [int, int] : seznam monch tah dVracznam monch tah d

Args: board (CheckersBoard): achovnice, na kter d nach

Returns: List of [int, int] : seznam monch tah d

Args

board (CheckersBoard): achovnice, na kter d nach

Returns

List of [int, int] : seznam monch tah d

Method: possibleJumps

Args: board (CheckersBoard): achovnice, na kter d nach position ([int, int], optional): pozice, ze kter mt. Vchozdnota je None, co znamene se sk ze sounzice d

Returns: List of [int, int]: seznam monch prvn skok dVracnam monch prvn skok d

Args: board (CheckersBoard): achovnice, na kter d nach position ([int, int], optional): pozice, ze kter mt. Vchozdnota je None, co znamene se sk ze sounzice d

Returns: List of [int, int]: seznam monch prvn skok d

Args

board (CheckersBoard): achovnice, na kter d nach position ([int, int], optional): pozice, ze kter mt. Vchozdnota je None, co znamene se sk ze sounzice d

Returns

List of [int, int]: seznam monch prvn skok d

Method: trackJumps

Args: endPosition ([int, int]): koncovzice, na kterou se mma dostat

Returns: List of [int, int]: seznam pozic figurek, kterma pesko ne se dostane na koncovou poziciVracnam pozic figurek, kterma pesko ne se dostane na koncovou pozici

Args: endPosition ([int, int]): koncovzice, na kterou se mma dostat

Returns: List of [int, int]: seznam pozic figurek, kterma pesko ne se dostane na koncovou pozici

Args

endPosition ([int, int]): koncovzice, na kterou se mma dostat

Returns

List of [int, int]: seznam pozic figurek, kterma pesko ne se dostane na koncovou pozici

Class: CheckersWithFogOfWar

T CheckersWithFogOfWar sloueprezentaci hry D s mlhou vy.

Method: _init_

Konstruktor t CheckersWithFogOfWar

Method: getBoard

Args: color (Enum Colors): Barva hr na tahu

Returns: Array of Field: zakrytchovniceVrackrytou achovnici

Args: color (Enum Colors): Barva hr na tahu

Returns: Array of Field: zakrytchovnice

Args

color (Enum Colors): Barva hr na tahu

Returns

Array of Field: zakrytchovnice

Class: Chess

T reprezentuj hru achy

Method: _init_

Konstruktor t ach

Method: getBoard

Args: color (Enum Colors): Barva hr, pro kter se mchovnice vykreslit

*Returns: list of list of Field: achovniceVr achovnici v aktuvalu jako dvouroz-
mle Field*

Args: color (Enum Colors): Barva hr, pro kter se mchovnice vykreslit

Returns: list of list of Field: achovnice

Args

color (Enum Colors): Barva hr, pro kter se mchovnice vykreslit

Returns

list of list of Field: achovnice

Method: choosePiece

*Args: positionToPlay ([int, int]): pozice figurky, kterou chce hrac hrat color
(Enum Colors): Barva hrace, který chce hrat*

*Returns: (list of [int, int]): dostupne pozice, kam muze hrac hratFunkce pro
vyber figurky, kterou chce hrac hrat*

*Args: positionToPlay ([int, int]): pozice figurky, kterou chce hrac hrat color
(Enum Colors): Barva hrace, který chce hrat*

Returns: (list of [int, int]): dostupne pozice, kam muze hrac hrat

Args

positionToPlay ([int, int]): pozice figurky, kterou chce hrac hrat color (Enum Colors): Barva hrace, který chce hrat

Returns

(list of [int, int]): dostupne pozice, kam muze hrac hrat

Method: makeMove

Args: playedMove ([int, int]): pozice, kam chce hrac hrat color (Enum Colors): Barva hrace, který chce hrat rightClick (bool): True, pokud hrac klikl pravym tlacitkem mysi, jinak False

Returns: bool: tah se zdail nebo ne string: "Promote"pokud je potreba provest vylepseni pesakaProvedenhu hrace

Args: playedMove ([int, int]): pozice, kam chce hrac hrat color (Enum Colors): Barva hrace, který chce hrat rightClick (bool): True, pokud hrac klikl pravym tlacitkem mysi, jinak False

Returns: bool: tah se zdail nebo ne string: "Promote"pokud je potreba provest vylepseni pesaka

Args

playedMove ([int, int]): pozice, kam chce hrac hrat color (Enum Colors): Barva hrace, který chce hrat rightClick (bool): True, pokud hrac klikl pravym tlacitkem mysi, jinak False

Returns

bool: tah se zdail nebo ne string: "Promote"pokud je potreba provest vylepseni pesaka

Method: promote

Args: newFigure (string): figurka, na kterou se ma pesak zmenit ("Q", "R", "B", "N")

Returns: bool: podle toho zda se tah podaril nebo ne string: pokud hra skončilaPromote pesaka

Args: newFigure (string): figurka, na kterou se ma pesak zmenit ("Q", "R", "B", "N")

Returns: bool: podle toho zda se tah podaril nebo ne string: pokud hra skončila

Args

newFigure (string): figurka, na kterou se ma pesak zmenit ("Q", "R", "B", "N")

Returns

bool: podle toho zda se tah podaril nebo ne string: pokud hra skončila

Method: _endOfMove

Returns: True: novy stav sachovnice po tahu string: string pokud hra skončilaKonec tahu

Returns: True: novy stav sachovnice po tahu string: string pokud hra skončila

Returns

True: novy stav sachovnice po tahu string: string pokud hra skončila

Method: possibleMoves

Args: color (Enum Colors): barva hrace, pro ktereho se maji tahy vypsát

Returns: list of [int, int]: seznam moznych tahuVrati mozne tahy pro hrace, primo rozng Of War

Args: color (Enum Colors): barva hrace, pro ktereho se maji tahy vypsát

Returns: list of [int, int]: seznam moznych tahu

Args

color (Enum Colors): barva hrace, pro ktereho se maji tahy vypsát

Returns

list of [int, int]: seznam moznych tahu

Method: checkEnd

Returns: string: "Draw by fifty-move rule"pokud bylo 50 tahu bez pohybu pesaku nebo braneni string: "Draw by threefold repetition"pokud se stejná pozice opakovala 3x string: "color won"pokud byl vyhozen kral None: pokud hra neskonzilaKontrola konce hry

Returns: string: "Draw by fifty-move rule"pokud bylo 50 tahu bez pohybu pesaku nebo braneni string: "Draw by threefold repetition"pokud se stejná pozice opakovala 3x string: "color won"pokud byl vyhozen kral None: pokud hra neskonzila

Returns

string: "Draw by fifty-move rule"pokud bylo 50 tahu bez pohybu pesaku nebo braneni string: "Draw by threefold repetition"pokud se stejná pozice opakovala 3x string: "color won"pokud byl vyhozen kral None: pokud hra neskonzila

Method: killPiece

Args: piecePosition ([int, int]): pozice figurky, kterou chceme vyhodit Vyhodit figurky z hracsky

Args: piecePosition ([int, int]): pozice figurky, kterou chceme vyhodit

Args

piecePosition ([int, int]): pozice figurky, kterou chceme vyhodit

Method: _printToTerminal

Vytiskne hracsku do konzole

Class: ChessBoard

T reprezentuj achovnici s figurkami. D od t Board.

Method: _init_

Konstruktor t ChessBoard. Vytvochovnici 8x8 a um ni vechny figury ve standardnoad

Method: _getitem_

Args: index [int, int]: Tuple dvou integer, (row, col), oba 0-7

Returns: Piece: Instance t Piece, nebo None, pokud je polo prn

Vracguru na ur achovnici, nebo None, pokud je polo prnArgs: index [int, int]: Tuple dvou integer, (row, col), oba 0-7

Returns: Piece: Instance t Piece, nebo None, pokud je polo prn

Args

index [int, int]: Tuple dvou integer, (row, col), oba 0-7

Returns

*Piece: Instance t Piece, nebo None, pokud je polo prn*Method: _setitem_*

Args: index ([int, int]): Tuple dvou integer, (row, col), oba 0-7 value (Piece): Instance t Piece, nebo None, pokud mt polo prneturns: bool: True, pokud se podailo nastavit polo, jinak FalseNastavlo na achovnici jako board[row,col] namo board.board[row][col]

Args: index ([int, int]): Tuple dvou integer, (row, col), oba 0-7 value (Piece): Instance t Piece, nebo None, pokud mt polo prneturns: bool: True, pokud se podailo nastavit polo, jinak False

Args

*index ([int, int]): Tuple dvou integer, (row, col), oba 0-7 value (Piece): Instance t Piece, nebo None, pokud mt polo prn*Returns bool: True, pokud se podailo nastavit polo, jinak False*

Method: _str_

Vracring reprezentaci achovnice. Kadlo je reprezentov jako string, kter je tvoen z informacarymbolu figury, nebo jako string ”_”, pokud je polo prnola jsou oddna mezerou a jednotlivky jsou oddny znakem nov u ().

Method: _populateBoard

Nastavchovnici do norm stavu. Vichni pci jsou v druh sedmu, vechny ostatngury jsou v prvn osmu. Barva figurek je v souladu s konvence b je dole a n nahoe.

Method: pieceList

Args: color (Enum Colors): Barva figurek, kterceme najColors.WHITE nebo Colors.BLACK)

Returns: List of Pieces: List figurek danrvyVracst vech figurek danrvy na achovnici.

Args: color (Enum Colors): Barva figurek, kterceme najColors.WHITE nebo Colors.BLACK)

Returns: List of Pieces: List figurek danrvy

Args

color (Enum Colors): Barva figurek, kterceme najColors.WHITE nebo Colors.BLACK)

Returns

List of Pieces: List figurek danrvy

Method: copy

Returns: ChessBoard: Kopie achovniceVr kopii achovnice. Kadgura z origin-chovnice je nahrazena jejpiReturns: ChessBoard: Kopie achovnice

Returns

ChessBoard: Kopie achovnice

Method: compare

Args: board: achovnice, kter mrovnat s aktuchovniceturns: Boolean:True, pokud jsou achovnice stejninak FalsePorovnvchovnice. Tato metoda porovnvchovnice a vr True, pokud jsou stejninak vr False.

Args: board: achovnice, kter mrovnat s aktuchovniceturns: Boolean:True, pokud jsou achovnice stejninak False

Args

*board: achovnice, kter mrovnat s aktuchovnic*Returns Boolean:True, pokud jsou achovnice stejninak False*

Method: getListOfBoard

Returns: List of Struct Field: achovnice jako list of FieldVr achovnici jako list of Field.

Returns: List of Struct Field: achovnice jako list of Field

Returns

List of Struct Field: achovnice jako list of Field

Class: ChessMines

Hra achy s minami.

Method: _init_

Inicializace hry.

Method: _placeMines

Umny na ndnzice.

Method: getBoard

Vr hracsku.

Method: makeMove

Zpracuje tah.

Class: ChessMinesWithFogOfWar

Hra achy s minami a mlhou vy.

Method: `_init_`

Konstruktor t ChessMinesWithFogOfWar

Method: `getBoard`

Vr hracsku.

Class: Bishop

T reprezentuj figurku stelce v achu. D od t Piece.

Method: `_init_`

Konstruktor t Bishop. Volnstruktor t Piece.

Method: `copy`

Vytvopii instance t Bishop.

Method: `possibleMoves`

Vr seznam monch tah pro stelce.

Class: King

T reprezentuj figurku kr v achu. D od t Piece.

Method: `_init_`

Konstruktor t King. Volnstruktor t Piece.

Method: `copy`

Vytvopii instance t King.

Method: `move`

Args: board - achovnice end - cvzice tahuZkontroluje, zda je monov tah krm a provede ho.

Args: board - achovnice end - cvzice tahu

Args

board - achovnice end - cvzice tahu

Method: possibleMoves

Vr seznam monch tah pro kr.

Class: Knight

T reprezentuj figurku jezdce v achu. D od t Piece.

Method: _init_

Args: color (Colors): barva figurky position ([int, int]): pozice figurky Konstruktor t Knight. Volnstruktor t Piece

Args: color (Colors): barva figurky position ([int, int]): pozice figurky

Args

color (Colors): barva figurky position ([int, int]): pozice figurky

Method: copy

Vytvopii instance t Knight.

Method: possibleMoves

Args: board (dict): achovnice Vr seznam monch tah pro jezdce.

Args: board (dict): achovnice

Args

board (dict): achovnice

Class: Pawn

T reprezentuj pe v achu. D od t Piece.

Method: _init_

Konstruktor t Pawn. Volnstruktor t Piece.

Method: copy

Vytvopii instance t Pawn.

Method: move

Args: board (dict): achovnice end ([int, int]): cvzice tahu
Zkontroluje, zda je monov tah pem a provede ho.

Args: board (dict): achovnice end ([int, int]): cvzice tahu

Args

board (dict): achovnice end ([int, int]): cvzice tahu

Method: possibleMoves

Args: board (dict): achovnice
Vr seznam monch tah pro pe.

Args: board (dict): achovnice

Args

board (dict): achovnice

Class: Piece

T reprezentuj jednu figurku na achovnici.

Method: init

Konstruktor t Piece. Nastavru a pozici figurky.

Method: move

Args: board (Board): Sachovnice, na kterou se tah prov end ([row, col]):
Policko, kam se tah prov
Metoda pro pohnutgurky po achovnici

Args: board (Board): Sachovnice, na kterou se tah prov end ([row, col]):
Policko, kam se tah prov

Args

board (Board): Sachovnice, na kterou se tah prov end ([row, col]):
Policko, kam se tah prov

Method: row

Returns: int: dek figurky
Vracdek, na ktere figurka nach

Returns: int: dek figurky

Returns

int: dek figurky

Method: col

Returns: int: Sloupec figurky

Returns: int: Sloupec figurky

Returns

int: Sloupec figurky

Method: possibleMoves

Args: board (Board): Sachovnice, na kterou se tah prov

Returns: (list of [row, col]): Seznam vech monch tah ve form
vech monch tah danou figurkou

Args: board (Board): Sachovnice, na kterou se tah prov

Returns: (list of [row, col]): Seznam vech monch tah ve form

Args

board (Board): Sachovnice, na kterou se tah prov

Returns

(list of [row, col]): Seznam vech monch tah ve form

Method: copy

Vraci kopii objektu. Pouziva se, kdybychom chteli mit kopii objektu, bez toho, aby se menil puvodni objekt. Returns: (Piece): Kopie objektu

Returns

(Piece): Kopie objektu

Class: Queen

T reprezentuj figurku krvny v achu. D od t Piece.

Method: _init_

Konstruktor t Queen. Volnstruktor t Piece.

Method: copy

Vytvopii instance t Queen.

Method: possibleMoves

Vr seznam monch tah pro krnu.

Class: Rook

T reprezentuj figurku v v achu. D od t Piece.

Method: _init_

Konstruktor t Rook. Volnstruktor t Piece.

Method: copy

Vytvopii instance t Rook.

Method: possibleMoves

Args: board (dict): achovnice Vr seznam monch tah pro v
Args: board (dict): achovnice

Args

board (dict): achovnice

Class: ChessTrackGame

Hra ChessTrackGame.

Method: _init_

Inicializace hry.

Method: getBoard

Args: color (Enum Colors): Barva hr.
Returns: ChessTrackGameBoard: Hraccka. Vrnacsky.
Args: color (Enum Colors): Barva hr.
Returns: ChessTrackGameBoard: Hraccka.

Args

color (Enum Colors): Barva hr.

Returns

ChessTrackGameBoard: Hraccka.

Method: makeMove

Args: position ([int,int]): Pozice, na kterou se mmen umit. color (Enum Colors): Barva kamene. rightClick (bool): True, pokud se jednravako myi.

Returns: bool: True, pokud je tah platn, jinak False.Provedenhu.

Args: position ([int,int]): Pozice, na kterou se mmen umit. color (Enum Colors): Barva kamene. rightClick (bool): True, pokud se jednravako myi.

Returns: bool: True, pokud je tah platn, jinak False.

Args

position ([int,int]): Pozice, na kterou se mmen umit. color (Enum Colors): Barva kamene. rightClick (bool): True, pokud se jednravako myi.

Returns

bool: True, pokud je tah platn, jinak False.

Method: checkEnd

Returns: Enum EndGame: Vsledek hry.Kontrola konce hry.

Returns: Enum EndGame: Vsledek hry.

Returns

Enum EndGame: Vsledek hry.

Method: _printToTerminal

Vyps hracsky do termin.

Class: ChessTrackGameBoard

Hracska hry ChessTrackGame.

Method: _init_

Inicializace hracsky.

Method: _populateBoard

Inicializace hracsky.

Method: _spinBoard

Otoacsky.

Method: `_placeStone`

Args: position ([int,int]): Pozice, na kterou se mmen umit. color (Enum Colors): Barva kamene.Um kamene na desku.

Args: position ([int,int]): Pozice, na kterou se mmen umit. color (Enum Colors): Barva kamene.

Args

position ([int,int]): Pozice, na kterou se mmen umit. color (Enum Colors): Barva kamene.

Method: `makeMove`

Args: position ([int,int]): Pozice, na kterou se mmen umit. color (Enum Colors): Barva kamene.Provedenhu.

Args: position ([int,int]): Pozice, na kterou se mmen umit. color (Enum Colors): Barva kamene.

Args

position ([int,int]): Pozice, na kterou se mmen umit. color (Enum Colors): Barva kamene.

Method: `getListOfBoard`

Returns: list: Seznam seznam reprezentuj hracsku.Vr seznam seznam reprezentuj hracsku.

Returns: list: Seznam seznam reprezentuj hracsku.

Returns

list: Seznam seznam reprezentuj hracsku.

Method: `checkEnd`

Returns: str: Vsledek hry.Kontrola konce hry.

Returns: str: Vsledek hry.

Returns

str: Vsledek hry.

Method: `_checkWinner`

Returns: str: Vsledek hry.Kontrola vherce.

Returns: str: Vsledek hry.

Returns

str: Vsedek hry.

Class: ChessWithFogOfWar

T ChessWithFogOfWar sloueprezentaci hry achy s mlhou vy.

Method: _init_

Konstruktor t ChessWithFogOfWar

Method: getBoard

Args: color (Enum Colors): Barva hr na tahu

Returns: Array of Field: zakrytchovniceVrackrytou achovnici

Args: color (Enum Colors): Barva hr na tahu

Returns: Array of Field: zakrytchovnice

Args

color (Enum Colors): Barva hr na tahu

Returns

Array of Field: zakrytchovnice

Class: ConnectFour

Hraccka pro hru ConnectFour

Method: _init_

Inicializace hry ConnectFour

Method: getBoard

Args: color (Enum, optional): Barva hr. Defaults to None.

Returns: List of Struct: HracckaVracacku

Args: color (Enum, optional): Barva hr. Defaults to None.

Returns: List of Struct: Hraccka

Args

color (Enum, optional): Barva hr. Defaults to None.

Returns

List of Struct: Hracska

Method: makeMove

Args: position ([int,int]): Pozice tahu color (Enum, optional): Barva hr. Defaults to None. rightClick (bool, optional): True, pokud se jednravako myi. Defaults to False.

Returns: bool: True, pokud se tah podail, jinak FalseProvedenhu

Args: position ([int,int]): Pozice tahu color (Enum, optional): Barva hr. Defaults to None. rightClick (bool, optional): True, pokud se jednravako myi. Defaults to False.

Returns: bool: True, pokud se tah podail, jinak False

Args

position ([int,int]): Pozice tahu color (Enum, optional): Barva hr. Defaults to None. rightClick (bool, optional): True, pokud se jednravako myi. Defaults to False.

Returns

bool: True, pokud se tah podail, jinak False

Method: checkEnd

Returns: Enum: Barva vzeZjit konce hry

Returns: Enum: Barva vze

Returns

Enum: Barva vze

Method: _printToTerminal

Vpis hracsky do konzole

Class: ConnectFourBoard

Hracska pro hru ConnectFour

Method: _init_

Inicializace hry ConnectFour

Method: `_str_`

Vytvoring hracsky na vpis do konzole

Method: `_populateBoard`

Inicializace hernsky

Method: `findMove`

Args: position ([int,int]): Pozice tahu up (bool, optional): Smjiov um. Defaults to False, tedy dol.

Returns: [int,int]: Pozice, kam se mit kn Provedenhu

Args: position ([int,int]): Pozice tahu up (bool, optional): Smjiov um. Defaults to False, tedy dol.

Returns: [int,int]: Pozice, kam se mit kn

Args

position ([int,int]): Pozice tahu up (bool, optional): Smjiov um. Defaults to False, tedy dol.

Returns

[int,int]: Pozice, kam se mit kn

Method: `makeMove`

Args: position ([int,int]): Pozice tahu color (Enum): Barva hr

Returns: bool: True, pokud se tah podail, jinak FalseProvedenhu

Args: position ([int,int]): Pozice tahu color (Enum): Barva hr

Returns: bool: True, pokud se tah podail, jinak False

Args

position ([int,int]): Pozice tahu color (Enum): Barva hr

Returns

bool: True, pokud se tah podail, jinak False

Method: `isFull`

Returns: bool: True, pokud je deska plninak FalseZjistda je hernska plneturns:
bool: True, pokud je deska plninak False

Returns

bool: True, pokud je deska plninak False

Method: checkEnd

Args: position ([int,int]): Pozice tahu color (Enum): Barva hr

Returns: bool: True, pokud je konec hry, jinak FalseZjistda je konec hry

Args: position ([int,int]): Pozice tahu color (Enum): Barva hr

Returns: bool: True, pokud je konec hry, jinak False

Args

position ([int,int]): Pozice tahu color (Enum): Barva hr

Returns

bool: True, pokud je konec hry, jinak False

Method: checkWin

Args: color (Enum): Barva hr

Returns: bool: True, pokud hryhr jinak FalseZjistda hryhrArgs: color (Enum): Barva hr

Returns: bool: True, pokud hryhr jinak False

Args

color (Enum): Barva hr

Returns

bool: True, pokud hryhr jinak False

Method: checkHorizontal

Args: color (Enum): Barva hr

Returns: bool: True, pokud hryhrorizontinak FalseZjistda hryhrorizontrgs: color (Enum): Barva hr

Returns: bool: True, pokud hryhrorizontinak False

Args

color (Enum): Barva hr

Returns

bool: True, pokud hryhrorizontinak False

Method: checkVertical

Args: color (Enum): Barva hr

Returns: bool: True, pokud hryhrertikinak FalseZjistda hryhrertikrgs: color (Enum): Barva hr

Returns: bool: True, pokud hryhrertikinak False

Args

color (Enum): Barva hr

Returns

bool: True, pokud hryhrertikinak False

Method: checkDiagonal

Args: color (Enum): Barva hr

Returns: bool: True, pokud hryhriagoninak FalseZjistda hryhriagonrgs: color (Enum): Barva hr

Returns: bool: True, pokud hryhriagoninak False

Args

color (Enum): Barva hr

Returns

bool: True, pokud hryhriagoninak False

Method: getListOfBoard

Returns: List of Struct : List, kde kad k je list obsahuj figury na danuVracchovnici jako list

Returns: List of Struct : List, kde kad k je list obsahuj figury na danu

Returns

List of Struct : List, kde kad k je list obsahuj figury na danu

Class: Field

T reprezentuj polo na achovnici

Class: Figures

Enum pro typy figurek

Attribute: PAWN
Attribute: ROOK
Attribute: BISHOP
Attribute: KNIGHT
Attribute: QUEEN
Attribute: KING
Attribute: X
Attribute: O
Attribute: FLAG
Attribute: EXPLOSION
Attribute: MINE
Attribute: ONE
Attribute: TWO
Attribute: THREE
Attribute: FOUR
Attribute: FIVE
Attribute: SIX
Attribute: SEVEN
Attribute: EIGHT
Attribute: SHADOW
Attribute: MOLE
Attribute: LOGO

Class: Colors

Enum pro barvy

Attribute: WHITE

Attribute: BLACK

Attribute: RED

Attribute: GREEN

Method: `_str_`

Returns: String: reprezentovanáVrací reprezentaci barvy

Returns: String: reprezentovaná

Returns

String: reprezentovaná

Method: `changeColor`

Returns: Enum Colors: invertovanáVr invertovanu

Returns: Enum Colors: invertovaná

Returns

Enum Colors: invertovaná

Method: `changeColorFour`

Returns: Enum Colors: dalšíVr další

Returns: Enum Colors: další

Returns

Enum Colors: další

Class: `GameTemplate`

T GameTemplate slouží reprezentaci abstraktní hry.

Method: `_init_`

Konstruktor t GameTemplate

Class: `GameTests`

Testy na hry

Method: testInitialBoard

Args: name (string): jm hry game_class(game) : thryTestuje, zdasevytvaoacle

Args: name (string): jm hry game_class(game) : thry

Args

name (string): jm hry game_class(game) : thry

Method: testChoosePiece

Args: name (string): jm hry game_class(game) : thryposition([int,int]) : pozicefigurky, sekterousemhnoutcolor(EnumColors) : barvanatahuTestuje, zdasefigurkadbrat

Args: name (string): jm hry game_class(game) : thryposition([int,int]) : pozicefigurky, sekterousemhnoutcolor(EnumColors) : barvanatahu

Args

name (string): jm hry game_class(game) : thryposition([int,int]) : pozicefigurky, sekterousemhnoutcolor(EnumColors) : barvanatahu

Method: testChooseWrongPiece

Args: name (string): jm hry game_class(game) : thryposition([int,int]) : pozicefigurky, sekterousemhnoutcolor(EnumColors) : barvanatahuTestuje, zdasefigurkanedbrat

Args: name (string): jm hry game_class(game) : thryposition([int,int]) : pozicefigurky, sekterousemhnoutcolor(EnumColors) : barvanatahu

Args

name (string): jm hry game_class(game) : thryposition([int,int]) : pozicefigurky, sekterousemhnoutcolor(EnumColors) : barvanatahu

Method: testChooseUnablePiece

Args: name (string): jm hry game_class(game) : thryposition([int,int]) : pozicefigurky, sekterousemhnoutcolor(EnumColors) : barvanatahuTestuje, zdasefigurkanedbrat

Args: name (string): jm hry game_class(game) : thryposition([int,int]) : pozicefigurky, sekterousemhnoutcolor(EnumColors) : barvanatahu

Args

name (string): jm hry game_class(game) : thryposition([int,int]) : pozicefigurky, sekterousemhnoutcolor(EnumColors) : barvanatahu

Method: testMakeMove

Args: name (string): jm hry game_class(game) : thrychoose_position([int, int]) : pozice_figurky, sekterousemhnoutcolor(EnumColors) : barvanatahumove_position([int, int]) : pozice, kamsemgurkapohnoutTestuje, zdasepohybprovede

Args: name (string): jm hry game_class(game) : thrychoose_position([int, int]) : pozice_figurky, sekterousemhnoutcolor(EnumColors) : barvanatahumove_position([int, int]) : pozice, kamsemgurkapohnout

Args

name (string): jm hry game_class(game) : thrychoose_position([int, int]) : pozice_figurky, sekterousemhnoutcolor(EnumColors) : barvanatahumove_position([int, int]) : pozice, kamsemgurkapohnout

Method: testMakeWrongMove

Args: name (string): jm hry game_class(game) : thrychoose_position([int, int]) : pozice_figurky, sekterousemhnoutcolor(EnumColors) : barvanatahumove_position([int, int]) : pozice, kamsemgurkapohnoutTestuje, zdasepohybnelzeprovo

Args: name (string): jm hry game_class(game) : thrychoose_position([int, int]) : pozice_figurky, sekterousemhnoutcolor(EnumColors) : barvanatahumove_position([int, int]) : pozice, kamsemgurkapohnout

Args

name (string): jm hry game_class(game) : thrychoose_position([int, int]) : pozice_figurky, sekterousemhnoutcolor(EnumColors) : barvanatahumove_position([int, int]) : pozice, kamsemgurkapohnout

Method: testMakeUnableMove

Args: name (string): jm hry game_class(game) : thrychoose_position([int, int]) : pozice_figurky, sekterousemhnoutcolor(EnumColors) : barvanatahumove_position([int, int]) : pozice, kamsemgurkapohnoutTestuje, zdasepohybnelzeprovo

Args: name (string): jm hry game_class(game) : thrychoose_position([int, int]) : pozice_figurky, sekterousemhnoutcolor(EnumColors) : barvanatahumove_position([int, int]) : pozice, kamsemgurkapohnout

Args

name (string): jm hry game_class(game) : thrychoose_position([int, int]) : pozice_figurky, sekterousemhnoutcolor(EnumColors) : barvanatahumove_position([int, int]) : pozice, kamsemgurkapohnout

Method: testCheckEnd

Args: name (string): jm hry game_class(game) : thryTestuje, zdahraskona
Args: name (string): jm hry game_class(game) : thry

Args

name (string): jm hry game_class(game) : thry

Method: testWithChoosingPiece

Args: name (string): jm hry game_class(game) : thryTestuje, zda jedefinovvybngurky

Args: name (string): jm hry game_class(game) : thry

Args

name (string): jm hry game_class(game) : thry

Method: testFog

Args: name (string): jm hry game_class(game) : thryTestuje, zda jedefinovlhovefekt

Args: name (string): jm hry game_class(game) : thry

Args

name (string): jm hry game_class(game) : thry

Method: testNumberOfPlayers

Args: name (string): jm hry game_class(game) : thryTestuje, zda jedefinovohr

Args: name (string): jm hry game_class(game) : thry

Args

name (string): jm hry game_class(game) : thry

Method: testSimulateFullGame

Args: name (string): jm hry game_class(game) : thryTestuje, zdasehrazahrajedokonce

Args: name (string): jm hry game_class(game) : thry

Args

name (string): jm hry game_class(game) : thry

Class: HumanDoNotWorry

T pro hru v, nezlob se.

Method: _init_

Konstruktor t hry v, nezlob se.

Method: getBoard

Args: color (Enum Colors, optional): Barva hr na tahu. Defaults to None.

Returns: List of list of field: HracskaMetoda vr hracsku.

Args: color (Enum Colors, optional): Barva hr na tahu. Defaults to None.

Returns: List of list of field: Hracska

Args

color (Enum Colors, optional): Barva hr na tahu. Defaults to None.

Returns

List of list of field: Hracska

Method: choosePiece

Args: position (int): Pozice figurky color (Enum Colors, optional): Barva hr. Defaults to None.

Returns: bool: True, pokud se podailo zvolit figurku, jinak FalseMetoda zvolurku, kterou se bude hr

Args: position (int): Pozice figurky color (Enum Colors, optional): Barva hr. Defaults to None.

Returns: bool: True, pokud se podailo zvolit figurku, jinak False

Args

position (int): Pozice figurky color (Enum Colors, optional): Barva hr. Defaults to None.

Returns

bool: True, pokud se podailo zvolit figurku, jinak False

Method: makeMove

Args: position ([int,int]): Pozice figurky color (Enum Colors, optional): Barva hr. Defaults to None. rightClick (bool, optional): True, pokud se jednravako myi. Defaults to False.

Returns: bool: True, pokud se podailo prov tah, jinak FalseMetoda provede tah figurkou.

Args: position ([int,int]): Pozice figurky color (Enum Colors, optional): Barva hr. Defaults to None. rightClick (bool, optional): True, pokud se jednravako myi. Defaults to False.

Returns: bool: True, pokud se podailo prov tah, jinak False

Args

position ([int,int]): Pozice figurky color (Enum Colors, optional): Barva hr. Defaults to None. rightClick (bool, optional): True, pokud se jedná o pravý klik. Defaults to False.

Returns

bool: True, pokud se podařilo provést tah, jinak False

Method: _makeStandardMove

Args: row (int): řádek col (int): sloupec Metoda provede standardní tah figurky.
Args: row (int): řádek col (int): sloupec

Args

row (int): řádek col (int): sloupec

Method: _placeToFinal

Args: board (Board): Hrací deska piece (Piece): Figurka Metoda umístí figurku na c.
Args: board (Board): Hrací deska piece (Piece): Figurka

Args

board (Board): Hrací deska piece (Piece): Figurka

Method: checkEnd

Returns: string: Vrátí barvu hráče, který vyhrál, jinak None Metoda zkontroluje, zda hra skončila.

Returns: string: Vrátí barvu hráče, který vyhrál, jinak None

Returns

string: Vrátí barvu hráče, který vyhrál, jinak None

Method: rollDice

Returns: int: Hodnota kostky Metoda hodí kostku.

Returns: int: Hodnota kostky

Returns

int: Hodnota kostky

Method: _printToTerminal

Metoda vytiskne hracsku na obrazovku.

Class: HumanDoNotWorryBoard

T HumanDoNotWorryBoard sloueprezentaci hracsky hry v, nezlob se.

Method: _init_

Konstruktor t HumanDoNotWorryBoard.

Method: _str_

Vr textovou reprezentaci instance t HumanDoNotWorryBoard.

Method: isDeployed

Args: color (Colors): Barva figurkyMetoda, kteristda je alespona figurka zadanrvy nasazena.

Args: color (Colors): Barva figurky

Args

color (Colors): Barva figurky

Method: _populateBoard

Metoda, kterplnacsku figurkami.

Method: getListOfBoard

Returns: List of List of Field: Seznam seznam reprezentuj hracskuMetoda, kter seznam seznam reprezentuj hracsku.

Returns: List of List of Field: Seznam seznam reprezentuj hracsku

Returns

List of List of Field: Seznam seznam reprezentuj hracsku

Class: BlackPiece

T BlackPiece sloueprezentaci jednrngurky.

Method: `__init__`

Args: position ([int,int]): Pozice figurkyKonstruktor t BlackPiece.

Args: position ([int,int]): Pozice figurky

Args

position ([int,int]): Pozice figurky

Method: `__str__`

Returns: str: Textovprezentace instance t BlackPieceVracetovou reprezentaci instance t BlackPiece.

Returns: str: Textovprezentace instance t BlackPiece

Returns

str: Textovprezentace instance t BlackPiece

Class: GreenPiece

T GreenPiece sloueprezentaci jednengurky.

Method: `__init__`

Args: position ([int,int]): Pozice figurkyKonstruktor t GreenPiece.

Args: position ([int,int]): Pozice figurky

Args

position ([int,int]): Pozice figurky

Method: `__str__`

Returns: str: Textovprezentace instance t GreenPieceVracetovou reprezentaci instance t GreenPiece.

Returns: str: Textovprezentace instance t GreenPiece

Returns

str: Textovprezentace instance t GreenPiece

Class: Piece

T Piece sloueprezentaci jednengurky.

Method: `__init__`

Args: color (Colors): Barva figurky position ([int,int]): Pozice figurkyKonstruktor t Piece.

Args: color (Colors): Barva figurky position ([int,int]): Pozice figurky

Args

color (Colors): Barva figurky position ([int,int]): Pozice figurky

Method: `returnHome`

Metoda vr figurku dom.

Method: `possibleMoves`

Args: number (int): Po o hodch na kostce

Returns: List of [int,int]: Seznam monch tah figurkyMetoda vr seznam monch tah figurky.

Args: number (int): Po o hodch na kostce

Returns: List of [int,int]: Seznam monch tah figurky

Args

number (int): Po o hodch na kostce

Returns

List of [int,int]: Seznam monch tah figurky

Method: `makeStandartMove`

Args: board (Board): Hracska position ([int,int]): Pozice figurkyMetoda provede standardnh figurkou.

Args: board (Board): Hracska position ([int,int]): Pozice figurky

Args

board (Board): Hracska position ([int,int]): Pozice figurky

Class: `RedPiece`

T RedPiece slouueprezentaci jednrvenurky.

Method: `__init__`

Args: position ([int,int]): Pozice figurkyKonstruktor t RedPiece.

Args: position ([int,int]): Pozice figurky

Args

position ([int,int]): Pozice figurky

Method: `_str_`

Returns: str: Textovprezentace instance t RedPiece Vracetovou reprezentaci instance t RedPiece.

Returns: str: Textovprezentace instance t RedPiece

Returns

str: Textovprezentace instance t RedPiece

Class: WhitePiece

T WhitePiece slouueprezentaci jednlgurky.

Method: `_init_`

Args: position ([int,int]): Pozice figurkyKonstruktor t WhitePiece.

Args: position ([int,int]): Pozice figurky

Args

position ([int,int]): Pozice figurky

Method: `_str_`

Returns: str: Textovprezentace instance t WhitePiece Vracetovou reprezentaci instance t WhitePiece.

Returns: str: Textovprezentace instance t WhitePiece

Returns

str: Textovprezentace instance t WhitePiece

Class: Game

T reprezentuj hru v seznamu her pro frontend

Class: ListOfGames

T reprezentuj seznam her pro frontend

Method: getListOfGames

Returns: List[Game]: seznam herVracznam her pro frontend

Returns: List[Game]: seznam her

Returns

List[Game]: seznam her

Class: MathGame

T reprezentuj hru MathGame

Method: __init__

Konstruktor t matematicky. Vytvovou hru a nastave hodnoty.

Method: getBoard

Args: color (Enum Colors, optional): Barva hr. Defaults to None.

Returns: list: dvourozmle objekt FieldVr achovnici ve formourozm pole objekt Field

Args: color (Enum Colors, optional): Barva hr. Defaults to None.

Returns: list: dvourozmle objekt Field

Args

color (Enum Colors, optional): Barva hr. Defaults to None.

Returns

list: dvourozmle objekt Field

Method: choosePiece

Args: position ([int,int]): pozice figury color (Enum Colors, optional): Barva figury. Defaults to None.

Returns: List of List of int: List monch tah figury, prn list pokud tah nenn Vr monhy pro danou pozici

Args: position ([int,int]): pozice figury color (Enum Colors, optional): Barva figury. Defaults to None.

Returns: List of List of int: List monch tah figury, prn list pokud tah nenn

Args

position ([int,int]): pozice figury color (Enum Colors, optional): Barva figury. Defaults to None.

Returns

List of List of int: List monch tah figury, prn list pokud tah nenn

Method: makeMove

Args: move ([int,int]): novzice figury color (Enum Colors, optional): Barva figury. Defaults to None. rightClick (bool, optional): True, pokud hrlíkl pravm tlakem myi, jina False. Defaults to False.

Returns: bool: True, pokud se tah podail, jinak False List of List of int: List monch tah figury, pokud se dkraat v pohybuPesune figuru na jinou pozici

Args: move ([int,int]): novzice figury color (Enum Colors, optional): Barva figury. Defaults to None. rightClick (bool, optional): True, pokud hrlíkl pravm tlakem myi, jina False. Defaults to False.

Returns: bool: True, pokud se tah podail, jinak False List of List of int: List monch tah figury, pokud se dkraat v pohybu

Args

move ([int,int]): novzice figury color (Enum Colors, optional): Barva figury. Defaults to None. rightClick (bool, optional): True, pokud hrlíkl pravm tlakem myi, jina False. Defaults to False.

Returns

bool: True, pokud se tah podail, jinak False List of List of int: List monch tah figury, pokud se dkraat v pohybu

Method: checkEnd

Returns: String: Vracsledek hry, pokud hra skona, jinak NoneZkontroluje, zda hra skona

Returns: String: Vracsledek hry, pokud hra skona, jinak None

Returns

String: Vracsledek hry, pokud hra skona, jinak None

Method: _printToTerminal

Vytiskne aktuav hry na termin

Class: MathGameBoard

T reprezentuj achovnici pro matematickou hru. achovnice je 8x8 a obsahuje figury typu Colors a "TASK". Figury typu Colors jsou na achovnici reprezentov barvou, figury typu "TASK"jsou koly, kter poteba splnit.

Method: `_init_`

Konstruktor t hernsky pro matematickou hru. Vytvovou achovnici a nastave hodnoty.

Method: `_getitem_`

Args: index [int, int]: Tuple dvou integer, (row, col), oba 0-7 Vracguru na ur achovnici, nebo None, pokud je polo prn
Args: index [int, int]: Tuple dvou integer, (row, col), oba 0-7

Args

index [int, int]: Tuple dvou integer, (row, col), oba 0-7

Method: `_setitem_`

Args: index ([int, int]): Tuple dvou integer, (row, col), oba 0-7 value (Piece): Instance t Piece, nebo None, pokud mt polo prn
returns: bool: True, pokud se podailo nastavit polo, jinak False
Nastavlo na achovnici jako board[row,col] namo board.board[row][col]

Args: index ([int, int]): Tuple dvou integer, (row, col), oba 0-7 value (Piece): Instance t Piece, nebo None, pokud mt polo prn
returns: bool: True, pokud se podailo nastavit polo, jinak False

Args

index ([int, int]): Tuple dvou integer, (row, col), oba 0-7 value (Piece): Instance t Piece, nebo None, pokud mt polo prn
**Returns bool: True, pokud se podailo nastavit polo, jinak False*

Method: `_str_`

Returns: String: String reprezentace achovnice Vracring reprezentaci achovnice. Kadlo je reprezentov jako string, kter je tvoen z informacarvymbolu figury, nebo jako string "_", pokud je polo prnola jsou oddna mezerou a jednotlivky jsou oddny znakem nov u ().

Returns: String: String reprezentace achovnice

Returns

String: String reprezentace achovnice

Method: `_populateBoard`

Nastavchovnici do pocho stavu. B je vpravo dole a n vlevo nahoe.

Method: TasksLeft

Returns: int: Po zbvajch kolVrac zbvajch kol na achovnici.

Returns: int: Po zbvajch kol

Returns

int: Po zbvajch kol

Method: getPosition

Args: color (Enum Colors): Barva figurek, kterceme najColors.WHITE nebo Colors.BLACK)

Returns: [int, int]: Pozice figury na achovniciVraczici figury danrvy.

Args: color (Enum Colors): Barva figurek, kterceme najColors.WHITE nebo Colors.BLACK)

Returns: [int, int]: Pozice figury na achovnici

Args

color (Enum Colors): Barva figurek, kterceme najColors.WHITE nebo Colors.BLACK)

Returns

[int, int]: Pozice figury na achovnici

Method: getPossibleMoves

Args: position ([int, int]): Pozice figury na achovnici

Returns: List of [int, int]: List monch tah figuryVracnhy figury na danzici.

Args: position ([int, int]): Pozice figury na achovnici

Returns: List of [int, int]: List monch tah figury

Args

position ([int, int]): Pozice figury na achovnici

Returns

List of [int, int]: List monch tah figury

Method: movePiece

Args: move ([int, int]): Novzice figury color (Enum Colors): Barva figury, kterou chceme pesunoutPesune figuru na jinsto na achovnici.

Args: move ([int, int]): Novzice figury color (Enum Colors): Barva figury, kterou chceme pesunout

Args

move ([int, int]): Novzice figury color (Enum Colors): Barva figury, kterou chceme pesunout

Method: getListOfBoard

Returns: List of Fields: List figurek na achovniciVracst vech figurek na achovnici.

Returns: List of Fields: List figurek na achovnici

Returns

List of Fields: List figurek na achovnici

Class: Mines

T reprezentuj hru Miny

Method: _init_

Inicializace hry Miny

Method: makeMove

Args: position ([int, int]): pozice, kam se mohnout color (Enum Colors): barva na tahu rightClick (bool): True, pokud hrlikl pravm tlakem myi, jinak False

Returns: bool: spost tahuProvede tah hr

Args: position ([int, int]): pozice, kam se mohnout color (Enum Colors): barva na tahu rightClick (bool): True, pokud hrlikl pravm tlakem myi, jinak False

Returns: bool: spost tahu

Args

position ([int, int]): pozice, kam se mohnout color (Enum Colors): barva na tahu rightClick (bool): True, pokud hrlikl pravm tlakem myi, jinak False

Returns

bool: spost tahu

Method: makeUncoverMove

Args: position ([int, int]): pozice, kam se mohnout color (Enum Colors): barva na tahu

Returns: bool: spost tahuProvede tah

Args: position ([int, int]): pozice, kam se mohnout color (Enum Colors): barva na tahu

Returns: bool: spost tahu

Args

position ([int, int]): pozice, kam se mohnout color (Enum Colors): barva na tahu

Returns

bool: spost tahu

Method: checkEnd

Returns: None: pokud hra neskona string: vsledek hryZkontroluje, zda hra skona

Returns: None: pokud hra neskona string: vsledek hry

Returns

None: pokud hra neskona string: vsledek hry

Method: placeFlag

Args: position ([int, int]): pozice, kam se majka umit color (Enum Colors): barva na tahu

Returns: bool: spost um vlajkyUmajku na danou pozici

Args: position ([int, int]): pozice, kam se majka umit color (Enum Colors): barva na tahu

Returns: bool: spost um vlajky

Args

position ([int, int]): pozice, kam se majka umit color (Enum Colors): barva na tahu

Returns

bool: spost um vlajky

Method: `getBoard`

Args: color (Enum Colors, optional): Barva, kter na tahu. Vchoz None.

Returns: list: hernskaVr hernsku

Args: color (Enum Colors, optional): Barva, kter na tahu. Vchoz None.

Returns: list: hernska

Args

color (Enum Colors, optional): Barva, kter na tahu. Vchoz None.

Returns

list: hernska

Method: `_printToTerminal`

Vyp hernsku do konzole

Class: MinesBoard

T reprezentuj hracsku hry Miny

Method: `__init__`

Args: numberOfMines (int): po min na hracsceInicializace hracsky Miny

Args: numberOfMines (int): po min na hracsce

Args

numberOfMines (int): po min na hracsce

Method: `_populateBoard`

Naplncsku minami

Method: `_placeMines`

Umny na hracsku

Method: `countMinesAroundSymbol`

Args: row (int): k col (int): sloupec

Returns: int: po min kolem symboluSpony kolem symbolu

Args: row (int): k col (int): sloupec

Returns: int: po min kolem symbolu

Args

row (int): k col (int): sloupec

Returns

int: po min kolem symbolu

Method: minesRemaining

Returns: int: po zbvajch neoznach minSpovaj neoznany na hracsce

Returns: int: po zbvajch neoznach min

Returns

int: po zbvajch neoznach min

Method: flagsPlanted

Returns: int: po vlajekSpo vlajek na hracsce

Returns: int: po vlajek

Returns

int: po vlajek

Method: makeMove

Args: row (int): k col (int): sloupec

Returns: bool: True, pokud tah byl sp, jinak FalseProvede tah na hracsce

Args: row (int): k col (int): sloupec

Returns: bool: True, pokud tah byl sp, jinak False

Args

row (int): k col (int): sloupec

Returns

bool: True, pokud tah byl sp, jinak False

Method: _showBoard

Args: row (int): k col (int): sloupecOdkryje dostupnmboly na hracsce po odkrytla

Args: row (int): k col (int): sloupec

Args

row (int): k col (int): sloupec

Method: placeFlag

Args: row (int): k col (int): sloupec

Returns: bool: False, pokud se vlajka neumt: -1, pokud se sprajka odstran, pokud se vlajka umr, pokud se vlajka odstranbo pid patnstoUmajku na hracsku

Args: row (int): k col (int): sloupec

Returns: bool: False, pokud se vlajka neumt: -1, pokud se sprajka odstran, pokud se vlajka umr, pokud se vlajka odstranbo pid patnsto

Args

row (int): k col (int): sloupec

Returns

bool: False, pokud se vlajka neumt: -1, pokud se sprajka odstran, pokud se vlajka umr, pokud se vlajka odstranbo pid patnsto

Method: _str_

Returns: string: textovprezentace hracskyVygeneruje textovou reprezentaci hracsky

Returns: string: textovprezentace hracsky

Returns

string: textovprezentace hracsky

Class: Pexeso

T reprezentuj hru Pexeso

Method: _init_

Inicializace hry Pexeso

Method: getBoard

Args: color (Enum Colors): barva hr

Returns: list: hracochaVr hracochu

Args: color (Enum Colors): barva hr

Returns: list: hracocha

Args

color (Enum Colors): barva hr

Returns

list: hracocha

Method: makeMove

Args: position ([int, int]): pozice, kterou chce hrto color (Enum Colors): barva na tahu rightClick (bool): True, pokud hrlikl pravm tlakem myi, jinak False

Returns: bool: spost tahuProvede tah

Args: position ([int, int]): pozice, kterou chce hrto color (Enum Colors): barva na tahu rightClick (bool): True, pokud hrlikl pravm tlakem myi, jinak False

Returns: bool: spost tahu

Args

position ([int, int]): pozice, kterou chce hrto color (Enum Colors): barva na tahu rightClick (bool): True, pokud hrlikl pravm tlakem myi, jinak False

Returns

bool: spost tahu

Method: checkEnd

Returns: None: pokud hra neskona string: vsledek hryZkontroluje, zda hra skona

Returns: None: pokud hra neskona string: vsledek hry

Returns

None: pokud hra neskona string: vsledek hry

Method: _printToTerminal

Funkce pro vyps hracochy do termin

Class: PexesoBoard

Reprezentace hracsky hry Pexeso

Method: `_init_`

Inicializace hracsky

Method: `_populateBoard`

Vygeneruje ndnrty na hracsku

Method: `shuffleBoard`

Zamrti na hracsce

Method: `hideCards`

Schovej vechny karti

Method: `_str_`

Vr textovou reprezentaci hracsky

Method: `makeMove`

Provede tah

Method: `isRevealed`

Args: position ([int, int]): pozice karti

Returns: bool: zda je karti odhalena Vr, zda je karti na danzici odhalena

Args: position ([int, int]): pozice karti

Returns: bool: zda je karti odhalena

Args

position ([int, int]): pozice karti

Returns

bool: zda je karti odhalena

Method: `isCompleted`

Args: position ([int, int]): pozice karti

Returns: bool: zda je karti uhodnuta Vr, zda je karti na danzici uhodnuta

Args: position ([int, int]): pozice karti

Returns: bool: zda je karti uhodnuta

Args

position ([int, int]): pozice karti

Returns

bool: zda je karti uhodnuta

Method: getListOfBoard

Vr hracochu

Class: PexesoCard

Reprezentace jednrti hry Pexeso

Method: _init_

Args: symbol (Field): symbol karti odhalena (bool): zda je karti odhalena
Initializace karti

Args: symbol (Field): symbol karti odhalena (bool): zda je karti odhalena

Args

symbol (Field): symbol karti odhalena (bool): zda je karti odhalena

Method: getSymbol

Returns: Field: symbol karti
Vr symbol karti

Returns: Field: symbol karti

Returns

Field: symbol karti

Method: equals

Args: other (PexesoCard): druhrti

Returns: bool: zda jsou symboly stejné

Porovnání dvou karti

Args: other (PexesoCard): druhrti

Returns: bool: zda jsou symboly stejné

Args

other (PexesoCard): druhrti

Returns

*bool: zda jsou symboly stejné*Method: turn Otoarti*

Method: hide

Schovej karti

Method: reveal

Odhal karti

Method: match

Oznarti jako vyeeenou

Class: TicTacToe

T reprezentuj hru Pikvorky

Method: _init_

Inicializace hry

Method: getBoard

Args: color (Enum Colors, optional): barva hr na tahu. Vchozstaven na pravidelntnReturns: List: hracskaVr hracsku

Args: color (Enum Colors, optional): barva hr na tahu. Vchozstaven na pravidelntnReturns: List: hracska

Args

color (Enum Colors, optional): barva hr na tahu. Vchozstaven na pravidelntn

Returns

List: hracska

Method: makeMove

Args: index ([int,int]): pozice kterou chce hrbsadit player (Enum Colors, optional): Hro mt na tahu. Vchoz nastaveno na stnightClick (bool, optional): True, pokud se jednravako myi. Defaults to False.

Returns: Boolean: true pokud se tah podail, jinak falseMetoda na zahr tahu

Args: index ([int,int]): pozice kterou chce hrbsadit player (Enum Colors, optional): Hro mt na tahu. Vchoz nastaveno na stnightClick (bool, optional): True, pokud se jednravako myi. Defaults to False.

Returns: Boolean: true pokud se tah podail, jinak false

Args

index ([int,int]): pozice kterou chce hrbsadit player (Enum Colors, optional): Hro mt na tahu. Vchoz nastaveno na stnightClick (bool, optional): True, pokud se jednravako myi. Defaults to False.

Returns

Boolean: true pokud se tah podail, jinak false

Method: checkEnd

Returns: String: "Draw"pokud je rem String: "barva won"pokud no vyhrone: pokud hra neskonaKontroluje jestli hra skona

Returns: String: "Draw"pokud je rem String: "barva won"pokud no vyhrone: pokud hra neskona

Returns

String: "Draw"pokud je rem String: "barva won"pokud no vyhrone: pokud hra neskona

Method: reset

Resetuje hru

Method: _printToTerminal

Vytiskne hracsku do konzole

Method: _checkWinner

Returns: Enum Colors: barva hr, kter vyhrone: pokud nikdo nevyhr

Kontroluje jestli no vyhrReturns: Enum Colors: barva hr, kter vyhrone: pokud nikdo nevyhr

Returns

Enum Colors: barva hr, kter vyhrone: pokud nikdo nevyhr

Method: _checkDraw

*Returns: Boolean: true pokud je rem, jinak false*Kontroluje jestli je rem

Returns: Boolean: true pokud je rem, jinak false

Returns

Boolean: true pokud je rem, jinak false

Class: TicTacToeBoard

T reprezentuj hracsku hru Pikvorky

Method: _init_

Konstruktor t TicTacToeBoard

Method: _populateBoard

Inicializace hernsky

Method: _str_

Vytiskne hracsku do konzole

Method: getListOfBoard

*Returns: List of Struct Field: *a*description*Vracznampolknahracsce

*Returns: List of Struct Field: *a*description*

Returns

*List of Struct Field: *a*description*

Class: ClickableLabel

T ClickableLabel slouzytvoenikateln labelu, kter me vyslat sign kliknut danlo

Attribute: clicked**Method: _init_**

Konstruktor t ClickableLabel

Method: mousePressEvent

Metoda, kter zavoli kliknut danlo

Class: GameView

T GameView slouobrazeny na grafickozhranmocQt5

Method: `_init_`

Args: game (game): Hra, kterou chceme zobrazitKonstruktor

Args: game (game): Hra, kterou chceme zobrazit

Args

game (game): Hra, kterou chceme zobrazit

Method: `set_piece_image`

Args: row (int): k col (int): sloupec filePath (string): cesta k obruFunkce pro nastavenru figurky na danzici

Args: row (int): k col (int): sloupec filePath (string): cesta k obru

Args

row (int): k col (int): sloupec filePath (string): cesta k obru

Method: `create_board`

Funkce pro vytvoenrnsky

Method: `show_question`

Funkce pro zobrazeny

Method: `handle_answer`

Args: correct (bool): Byla odpovpr

Funkce pro zpracov odpov na otu

Args: correct (bool): Byla odpovpr

Args

correct (bool): Byla odpovpr

Method: `handle_square_click`

Args: row (int): k col (int): sloupec button (string): tlako, kterlo stisknuto-

Funkce pro obsluhu kliknut polo

Args: row (int): k col (int): sloupec button (string): tlako, kterlo stisknuto

Args

row (int): k col (int): sloupec button (string): tlako, kterlo stisknuto

Method: choose_{piece}

Args: row (int): k col (int): sloupec Funkce pro vbigurky

Args: row (int): k col (int): sloupec

Args

row (int): k col (int): sloupec

Method: make_{move}

Args: row (int): k col (int): sloupec Funkce pro provedenhu

Args: row (int): k col (int): sloupec

Args

row (int): k col (int): sloupec

Method: highlight_{square}

Args: row (int): k col (int): sloupec Funkce pro zvravn pola

Args: row (int): k col (int): sloupec

Args

row (int): k col (int): sloupec

Method: update_{board}

Args: isFirst (bool, optional): Je to prvntualizace?. Defaults to False. Funkce pro aktualizaci hernsky

Args: isFirst (bool, optional): Je to prvntualizace?. Defaults to False.

Args

isFirst (bool, optional): Je to prvntualizace?. Defaults to False.

Method: remove_{board}

Funkce pro odstran hernsky

Method: game_{ended}

Args: message (string): Vsledek hry Funkce pro zobrazení logu okna s výsledkem hry

Args: message (string): Vsledek hry

Args

message (string): Vsledek hry

Method: promote_{pawn}

Funkce pro vbičurky, na kterou se msm

Class: GetResource

T GetResource slouží cestu k obru, která reprezentuje daný zdroj.

Method: getResource

Args: resource (str): Zdroj, pro který chceme získat cestu k obru.

Returns: str: cesta k obru, která reprezentuje zadaný zdroj. Metoda na základě zdroje vrátí cestu k obru, která reprezentuje daný zdroj.

Args: resource (str): Zdroj, pro který chceme získat cestu k obru.

Returns: str: cesta k obru, která reprezentuje zadaný zdroj.

Args

resource (str): Zdroj, pro který chceme získat cestu k obru.

Returns

str: cesta k obru, která reprezentuje zadaný zdroj.

Class: MainView

T MainView slouží k zobrazení menu aplikace.

Method: __init__

Konstruktor t

Method: start_{game}

Args: game (Game): objekt Game, který obsahuje náhry a objekt hry Spustit dle jm hry

Args: game (Game): objekt Game, který obsahuje náhry a objekt hry

Args

game (Game): objekt Game, kter obsahuje nv hry a objekt hry

Class: GenerateQuestion

T na generov otk Pro generov otk pouij metodu generateQuestion Pro kontrolu odpov pouij metodu checkAnswer s parametrem answer Pro z sprpov pouij funkci doupovcuvOperator Vsledky se zaokrouhluj celsla!!!

Method: `_init_`

Konstruktor t GenerateQuestion

Method: generateQuestion

Args: n (int): slo oty, defaultnhodneturns: string, string (questionText, questionLatex): otaMetoda na generov otk

Args: n (int): slo oty, defaultnhodneturns: string, string (questionText, questionLatex): ota

Args

*n (int): slo oty, defaultnhodn*Returns string, string (questionText, questionLatex): ota*

Class: AnalyticGeometryQuestionGenerator

Generr otk na analytickou geometrii

Method: `_init_`

Konstruktor t AnalyticGeometryQuestionGenerator

Method: generateQuestion

Args: n (int): slo oty, defaultnhodneturns: AnalyticGeometryQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpovGenerov ndny na analytickou geometrii

Args: n (int): slo oty, defaultnhodneturns: AnalyticGeometryQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov

Args

*n (int): slo oty, defaultnhodn*Returns AnalyticGeometryQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov*

Class: DerivativeQuestionGenerator

Generer otk na derivace

Method: `__init__`

Konstruktor t DerivativeQuestionGenerator

Method: `generatePolynomial`

Args: degree (int): stupeynomu

Returns: sympy symbol: polynom Generov ndn polynomu

Args: degree (int): stupeynomu

Returns: sympy symbol: polynom

Args

degree (int): stupeynomu

Returns

sympy symbol: polynom

Method: `generateQuestion`

Args: n (int): slo oty, defaultnhodnreturns: DerivativeQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpovGenerov ndny na derivace

Args: n (int): slo oty, defaultnhodnreturns: DerivativeQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov

Args

*n (int): slo oty, defaultnhodn*Returns DerivativeQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov*

Class: FractionQuestionGenerator

Generer otk na zlomky

Method: `__init__`

Konstruktor t FractionQuestionGenerator

Method: generateFraction

Returns: int, int: (numerator, denominator), kde numerator je atel v rozmez-100 a denominator jmenovatel v rozmez10 Vygenerov ndnojice el

Returns: int, int: (numerator, denominator), kde numerator je atel v rozmez-100 a denominator jmenovatel v rozmez10

Returns

int, int: (numerator, denominator), kde numerator je atel v rozmez-100 a denominator jmenovatel v rozmez10

Method: simplifyFraction

Args: numerator (int): atel denominator (int): jmenovatel Returns: int, int: (numerator, denominator), kde numerator je atel a denominator jmenovatel zjednoduen zlomku Zjednoduenomku

Args: numerator (int): atel denominator (int): jmenovatel Returns: int, int: (numerator, denominator), kde numerator je atel a denominator jmenovatel zjednoduen zlomku

Args

numerator (int): atel denominator (int): jmenovatel

Returns

int, int: (numerator, denominator), kde numerator je atel a denominator jmenovatel zjednoduen zlomku

Method: lowestCommonMultiple

Args: a (int): prvnslo b (int): druhslo

Returns: int: Nejmenole nbek Vpo nejmen spoleho nbku

Args: a (int): prvnslo b (int): druhslo

Returns: int: Nejmenole nbek

Args

a (int): prvnslo b (int): druhslo

Returns

int: Nejmenole nbek

Method: greatestCommonDivisor

*Args: a (int): prvnso b (int): druhslo Returns: int: Nejvole dtelVpo neju
spoleho dtele*

Args: a (int): prvnso b (int): druhslo Returns: int: Nejvole dtel

Args

a (int): prvnso b (int): druhslo

Returns

int: Nejvole dtel

Method: fractionToString

Args: numerator (int): atel denominator (int): jmenovatel

Returns: String: reprezentace zlomku v latexuPevedenomku na string

Args: numerator (int): atel denominator (int): jmenovatel

Returns: String: reprezentace zlomku v latexu

Args

numerator (int): atel denominator (int): jmenovatel

Returns

String: reprezentace zlomku v latexu

Method: fractionToAnswer

Args: numerator (int): atel denominator (int): jmenovatel

Returns: String: reprezentace zlomku ve form "a/b"

Pevedenomku na string

Args: numerator (int): atel denominator (int): jmenovatel

Returns: String: reprezentace zlomku ve form "a/b"

Args

numerator (int): atel denominator (int): jmenovatel

Returns

String: reprezentace zlomku ve form "a/b"

Method: generateQuestion

Args: n (int): slo oty, defaultnhodn returns: FractionQuestionGenerator: Samo sebe s vygenerovanou otou a odpov Vygenerov ndny na zlomky

Args: n (int): slo oty, defaultnhodn returns: FractionQuestionGenerator: Samo sebe s vygenerovanou otou a odpov

Args

*n (int): slo oty, defaultnhodn*Returns FractionQuestionGenerator: Samo sebe s vygenerovanou otou a odpov*

Class: InfinitiveSeriesQuestionGenerator

Generr otk na konvergenci nekonech ad

Method: _init_

Konstruktor t nekonech ad

Method: generateQuestion

Args: n (int): slo oty, defaultnhodn returns: InfinitiveSeriesQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov Generov ndny na konvergenci nekonech ad

Args: n (int): slo oty, defaultnhodn returns: InfinitiveSeriesQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov

Args

*n (int): slo oty, defaultnhodn*Returns InfinitiveSeriesQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov*

Class: IntegralQuestionGenerator

Generr otk na urdnoty integr

Method: _init_

Konstruktor t IntegralQuestionGenerator

Method: generateQuestion

Args: n (int): slo oty, defaultnhodneturns: IntegralQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpovGenerov ndny na urdnoty integr

Args: n (int): slo oty, defaultnhodneturns: IntegralQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov

Args

*n (int): slo oty, defaultnhodn*Returns IntegralQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov*

Class: KardinalNumberQuestionGenerator

Generr otk na kardinsla

Method: _init_

Konstruktor t otk na kardinsla

Method: generateQuestion

Args: n (int): slo oty, defaultnhodneturns: KardinalNumberQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpovGenerov ndny na kardinsla

Args: n (int): slo oty, defaultnhodneturns: KardinalNumberQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov

Args

*n (int): slo oty, defaultnhodn*Returns KardinalNumberQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov*

Class: LinearEquationSystemQuestionGenerator

Generr otk na lineustavy rovnice

Method: _init_

Konstruktor t soustav line rovnice

Method: generateLinearEquationSystem

Args: num_equations(int) : pochtchrovnice num_variables(int) : pochtchpromch

Returns: sympy equations, sympy symbols: Vygenerovanvnice a jejich prom

Vygenerov lineustavy rovnice

Args: num_equations(int) : pochtchrovnice num_variables(int) : pochtchpromch

Returns: sympy equations, sympy symbols: Vygenerovanvnice a jejich prom

Args

num_equations(int) : pochthronnicnum_variables(int) : pochthpromch

Returns

*sympy equations, sympy symbols: Vygenerovanvnice a jejich prom*Method: convert_to_ilatex*

Args: equations (sympy equations): vstupnvnice

Returns: string: latexov zs rovník ve form string

Args

equations (sympy equations): vstupnvnice

Returns

string: latexov zs rovník ve form string

Method: generateQuestion

Args: n (int): slo oty, defaultnhodnreturns: LinearEquationSystemQuestion-Generator: Funkce vrabe sama s vygenerovanou otouMetoda na generov oty

Args: n (int): slo oty, defaultnhodnreturns: LinearEquationSystemQuestion-Generator: Funkce vrabe sama s vygenerovanou otou

Args

*n (int): slo oty, defaultnhodn*Returns LinearEquationSystemQuestionGenera-
tor: Funkce vrabe sama s vygenerovanou otou*

Class: MatrixQuestionGenerator

Generr otk na matice

Method: _init_

Konstruktor t matice

Method: generateRegularMatrix

Args: n (int): Velikost matice, defaultnhodnzi 2 a 4

Returns: numpy array int: Ndn_gulticeGenerov regultice

Args: n (int): Velikost matice, defaultnhodnzi 2 a 4

Returns: numpy array int: Ndn_gultice

Args

n (int): Velikost matice, defaultn hodnzi 2 a 4

Returns

numpy array int: Ndngultice

Method: calculateDeterminant

Args: matrix (numpy array int): matice, ve tvaru numpy array int

Returns: float: Hodnota determinantu matice Vpo determinantu matice

Args: matrix (numpy array int): matice, ve tvaru numpy array int

Returns: float: Hodnota determinantu matice

Args

matrix (numpy array int): matice, ve tvaru numpy array int

Returns

float: Hodnota determinantu matice

Method: calculateInverseMatrix

Args: matrix (numpy array int): matice

Returns: numpy array int: inverzntice Vpo inverzntice

Args: matrix (numpy array int): matice

Returns: numpy array int: inverzntice

Args

matrix (numpy array int): matice

Returns

numpy array int: inverzntice

Method: calculateRank

Args: matrix (numpy array int): matice

Returns: int: atice Vpo matice

Args: matrix (numpy array int): matice

Returns: int: atice

Args

matrix (numpy array int): matice

Returns

int: atice

Method: calculateEigenvalues

Args: matrix (numpy array int): matice

Returns: float: sou vlastnsla matice Vpo vlastn el matice

Args: matrix (numpy array int): matice

Returns: float: sou vlastnsla matice

Args

matrix (numpy array int): matice

Returns

float: sou vlastnsla matice

Method: getLatexMatrix

Args: matrix (numpy.ndarray): matice

Returns: string: matice ve form LaTeX Metoda pro vygenerov matice ve form LaTeX

Args: matrix (numpy.ndarray): matice

Returns: string: matice ve form LaTeX

Args

matrix (numpy.ndarray): matice

Returns

string: matice ve form LaTeX

Method: generateQuestion

Returns: MatrixQuestionGenerator: Vracbe sama s vygenerovanmi otami Metoda na generov otk na matice

Returns: MatrixQuestionGenerator: Vracbe sama s vygenerovanmi otami

Returns

MatrixQuestionGenerator: Vracbe sama s vygenerovanmi otami

Class: OrdinalNumberQuestionGenerator

Generr otk na ordinsla

Method: `_init_`

Konstruktor t otk na ordinsla

Method: `generateQuestion`

Args: n (int): slo oty, defaultnhodneturns: OrdinalNumberQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpovGenerov ndny na usponsla

Args: n (int): slo oty, defaultnhodneturns: OrdinalNumberQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov

Args

*n (int): slo oty, defaultnhodn*Returns OrdinalNumberQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov*

Class: `RegularLanguageQuestionGenerator`

Generr otk na regulzky

Method: `_init_`

Konstruktor t regul jazyk

Method: `generateQuestion`

Args: n (int): slo oty, defaultnhodneturns: RegularLanguageQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpovGenerov ndny na regulzky

Args: n (int): slo oty, defaultnhodneturns: RegularLanguageQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov

Args

*n (int): slo oty, defaultnhodn*Returns RegularLanguageQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov*

Class: `SetQuestionGenerator`

Generr otk na mnoiny

Method: `_init_`

Konstruktor t otk na mnoiny

Method: generateQuestion

Args: n (int): slo otý, defaultnhodneturns: SetQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpovGenerov ndny na mnoiny

Args: n (int): slo otý, defaultnhodneturns: SetQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov

Args

*n (int): slo otý, defaultnhodn*Returns SetQuestionGenerator: Vracmo sebe s vygenerovanou otou a odpov*

Class: Question

T pedpisu otý

Method: _init_

Konstruktor t Question

Method: _str_

Returns: string: Vr otu a odpov

Metoda na vpis otý s odpov

Returns: string: Vr otu a odpov

Returns

string: Vr otu a odpov

Method: checkAnswer

Args: string: answer - zadanpovReturns: bool: true pokud je odpovpr

Metoda na kontrolu odpov

Args: string: answer - zadanpovReturns: bool: true pokud je odpovpr

Args

string: answer - zadanpov

Returns

*bool: true pokud je odpovpr*Method: doupovcuvOperator*

Returns: string: odpov

Metoda na z odpov

Returns: string: odpov

Returns

string: odpov

Method: generateQuestion

Returns: string, string: VracuFunkce na generov oty

Returns: string, string: Vracu

Returns

string, string: Vracu

Class: QuestionTests

Testy na generry otk

Attribute: allClasses

Method: testNumberOfQuestions

Args: name (string): jm generru generator_class(Question) : tgenerruTestuje, zdasegenerujenenulovpootk

Args: name (string): jm generru generator_class(Question) : tgenerru

Args

name (string): jm generru generator_class(Question) : tgenerru

Method: testGenerateQuestion

Args: name (string): jm generru generator_class(Question) : tgenerruTestuje, zdasevygenerujeota

Args: name (string): jm generru generator_class(Question) : tgenerru

Args

name (string): jm generru generator_class(Question) : tgenerru

Method: testDoupovcuvOperator

Args: name (string): jm generru generator_class(Question) : tgenerruTestuje, zdasevygenerujeodpovArgs

name(string) : jmgenerrugenerator_class(Question) : tgenerru

Args

name (string): jm generru generator_class(Question) : tgenerru

Method: testCheckAnswer

*Args: name (string): jm generru generator_class(Question) : tgenerruTestuje, zda je odpovprrgs :
name(string) : jmgenerrugenerator_class(Question) : tgenerru*

Args

name (string): jm generru generator_class(Question) : tgenerru

Method: testWrongAnswer

*Args: name (string): jm generru generator_class(Question) : tgenerruTestuje, zda je odpovpatnrgs :
name(string) : jmgenerrugenerator_class(Question) : tgenerru*

Args

name (string): jm generru generator_class(Question) : tgenerru

Method: testTimeSet

Method: testHoderova

Class: MathQuestion

T MathQuestion slouobrazentematicky.

Method: _init_

*Args: question (Question): Ota, kter mbrazit color (Colors): Barva hr, kter
mpovt fullscreen (bool): Zda se mbrazit na celou obrazovku callback (function):
Funkce, kter mvolat po zodpovnyKonstruktor t*

*Args: question (Question): Ota, kter mbrazit color (Colors): Barva hr, kter
mpovt fullscreen (bool): Zda se mbrazit na celou obrazovku callback (function):
Funkce, kter mvolat po zodpovny*

Args

*question (Question): Ota, kter mbrazit color (Colors): Barva hr, kter mpovt
fullscreen (bool): Zda se mbrazit na celou obrazovku callback (function): Funkce,
kter mvolat po zodpovny*

Method: update_timer

Metoda na aktualizaci ova

Method: time_out

Metoda na oznne vyprel

Method: `checkaanswer`

Metoda na kontrolu odpov

Method: `killyourself`

Metoda pro ukonna galantnstou

Method: `renderlatextokatex`

*Args: latexEq (string): rovnice, kter majkreslit v LaTeXuMetoda na vytvo-
envnic ve form LaTeX*

Args: latexEq (string): rovnice, kter majkreslit v LaTeXu

Args

latexEq (string): rovnice, kter majkreslit v LaTeXu