PA11: Rush Hour 2

Generated by Doxygen 1.8.11

Contents

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Chapter 3

Class Documentation

3.1 Car Class Reference

`#include <RushHour.h>`

Public Member Functions

- Car ()

  Car Class Constructor.
- void setCar (const int size, const char facing, const int row_p, const int col_p)

  setCar function sets the values of the car object
- int getLength () const

  Returns the length of the vehicle.
- int getRow () const

  Returns row index value of the back of vehicle.
- int getCol () const

  Returns column index value of the back of vehicle.
- char getOrient () const

  Returns the orientation of the vehicle.
- void setRow (const int newRow)

  Sets the row value for car object.
- void setCol (const int newCol)

  Sets the col value for car object.

Public Attributes

- int row
- int col
- int length
- char orientation

3.1.1 Constructor & Destructor Documentation

3.1.1.1 Car::Car ( )

Car Class Constructor.

Postcondition

Car object is instantiated

3.1.2 Member Function Documentation

3.1.2.1 int Car::getCol ( ) const

Returns column index value of the back of vehicle.

Returns

Column index value of the back of vehicle

3.1.2.2 int Car::getLength ( ) const

Returns the length of the vehicle.

Returns

length of vehicle

3.1.2.3 char Car::getOrient ( ) const

Returns the orientation of the vehicle.

Returns

Orientation of the vehicle

3.1.2.4 int Car::getRow ( ) const

Returns row index value of the back of vehicle.

Returns

Row index value of the back of vehicle

3.1.2.5 void Car::setCar (const int *size*, const char *facing*, const int *row_p*, const int *col_p*)

setCar function sets the values of the car object

Parameters

size	Size of vehicle
facing	Direction of vehicle
row_p	Row location of vehicle
col_p	Column location of vehicle

Postcondition

Car object values are initialized

3.1.2.6 void Car::setCol (const int *newCol*)

Sets the col value for car object.

Parameters

newCol	int value to set car object's col value

Postcondition

col value for car object is set to newCol parameter

3.1.2.7 void Car::setRow (const int *newRow*)

Sets the row value for car object.

Parameters

newRow	int value to set car object's row value

Postcondition

row value for car object is set to newRow parameter

3.1.3 Member Data Documentation

3.1.3.1 int Car::col

3.1.3.2 int Car::length

3.1.3.3 char Car::orientation

3.1.3.4 int Car::row

The documentation for this class was generated from the following files:

- RushHour.h
- RushHour.cpp

3.2 Rushhour Class Reference

```
#include <RushHour.h>
```

Public Member Functions

- Rushhour ()

  Rushhour() default constructor.
- void solveIt ()

  Solves the board to find the least amount of moves.
- bool didWeWin () const

  Checks if board is in a winning position.
- void fillGameBoard ()

  Places stored data from user onto the game board.
- void printBoard () const

  Prints board to screen.
- void clearBoard ()

  Clears the queue of boards.
- void fixCarList ()

  Sets the Carlist to be the same as board that is popped off the queue.
- bool moveForward (const int index)

  Moves the vehicle forward.
- bool moveBackward (const int numCar)

  Move the vehicle backward.
- int getBest () const

  Returns best number of moves.
- bool getSolved () const

  Returns whether board has been solved.
- void setNumberOfCars (const int cars)

  Sets the number of cars.
- int getNumberOfCars () const

  Returns the value of numberOfCars.
- void addQueue ()

  Adds board to the queue.
- bool queueEmpty ()

  Checks if queue is empty.
- void readData ()

  Reads data entered by user.

3.2.1 Constructor & Destructor Documentation

3.2.1.1 Rushhour::Rushhour ( )

Rushhour() default constructor.

Precondition

{ description of the precondition }

Postcondition

Rushhour object is created

Note

initializes the key

3.2.2 Member Function Documentation

3.2.2.1 void Rushhour::addQueue ( )

Adds board to the queue.

Postcondition

Board has been added to the queue

Note

The board is converted from a 2D array to a string for storage on the queue

3.2.2.2 void Rushhour::clearBoard ( )

Clears the queue of boards.

Postcondition

Queue is empty

3.2.2.3 bool Rushhour::didWeWin ( ) const

Checks if board is in a winning position.

Note

Checks if board is in a winning board by searching last column for car 0

Returns

True if win condition. False otherwise.

3.2.2.4 void Rushhour::fillGameBoard ( )

Places stored data from user onto the game board.

Precondition

Board is empty

Postcondition

Board is filled

3.2.2.5 void Rushhour::fixCarList ( )

Sets the Carlist to be the same as board that is popped off the queue.

3.2.2.6 int Rushhour::getBest ( ) const

Returns best number of moves.

Returns

Best number of moves

3.2.2.7 int Rushhour::getNumberOfCars ( ) const

Returns the value of numberOfCars.

Returns

numberOfCars

3.2.2.8 bool Rushhour::getSolved ( ) const

Returns whether board has been solved.

Returns

True if board is solved. False otherwise.

3.2.2.9 bool Rushhour::moveBackward (const int *numCar*)

Move the vehicle backward.

Parameters

numCar	vehicle identification number

Postcondition

vehicle is moved one space backward

Returns

True if successfully move. False otherwise.

3.2.2.10 bool Rushhour::moveForward (const int *index*)

Moves the vehicle forward.

Parameters

index	Index value of the back position of vehicle

Postcondition

vehicle is moved one space forward

Returns

True if successfully move. False otherwise.

3.2.2.11 void Rushhour::printBoard ( ) const

Prints board to screen.

Postcondition

Board is printed to screen

3.2.2.12 bool Rushhour::queueEmpty ( )

Checks if queue is empty.

Returns

True if queue is empty, false otherwise.

3.2.2.13 void Rushhour::readData ( )

Reads data entered by user.

Postcondition

Data from terminal is stored

3.2.2.14 void Rushhour::setNumberOfCars (const int *cars*)

Sets the number of cars.

Parameters

cars	int representing the amount of vehicles to be loaded onto the game board

Postcondition

numberOfCars variable is initialized

3.2.2.15 void Rushhour::solveIt ( )

Solves the board to find the least amount of moves.

Note

Uses a breadth-first search to find the least amount of moves necessary to solve the board

The documentation for this class was generated from the following files:

- RushHour.h
- RushHour.cpp

Chapter 4

File Documentation

4.1 RushHour.cpp File Reference

```
#include <iostream>
#include <map>
#include <queue>
#include <string>
#include "RushHour.h"
```

Macros

- #define MAXROWS 6
- #define MAXCOLS 6
- #define MAXCARS 18

Functions

- int main ()

4.1.1 Macro Definition Documentation

4.1.1.1 #define MAXCARS 18

4.1.1.2 #define MAXCOLS 6

4.1.1.3 #define MAXROWS 6

4.1.2 Function Documentation

4.1.2.1 int main ( )

4.2 RushHour.h File Reference

```
#include <map>
#include <queue>
#include <iostream>
#include <string>
```

Classes

- class Car
- class Rushhour

Macros

- #define MAXCARS 18
- #define MAXROWS 6
- #define MAXCOLS 6

4.2.1 Detailed Description

Author

Terra Williams
Natalie Arnold
Leah Kramer

Date

12/05/2017

4.2.2 Macro Definition Documentation

4.2.2.1 #define MAXCARS 18

4.2.2.2 #define MAXCOLS 6

4.2.2.3 #define MAXROWS 6

Index