Team Report

Function Documentation

By Adrian and Boston

Table of Contents

[Draw 4](#_Toc515549446)

[Update 6](#_Toc515549447)

[Init Levels 7](#_Toc515549448)

[Create Grid 8](#_Toc515549449)

[Add Background 8](#_Toc515549450)

[Move 9](#_Toc515549451)

[Merge 9](#_Toc515549452)

[Is Odd 10](#_Toc515549453)

[Correct Box 10](#_Toc515549454)

[Attract Mode Init 11](#_Toc515549455)

[Attract Mode Update 12](#_Toc515549456)

[Collision 12](#_Toc515549457)

void draw();

void update();

void nextLevel();

void init();

void initTutorial();

void initStage1();

void initStage2();

void initStage3();

void initStage4();

void initStage5();

void initStage6();

void initStage7();

void initStage9();

void createGrid(const unsigned short \*map);

void addBackground(const unsigned short \*wallTiles, const unsigned short \*wallMap, const unsigned short \*floorTiles, const unsigned short \*floorMap);

void move(int changeX, int changeY);

bool merge(int boxToMerge);

bool isOdd(int n);

bool correctBox(Box box, Box dropbox);

int collision(int x, int y, bool mergeBoxes);

void attractModeInit();

void attractModeUdate(int step);

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# Draw

void draw();

Syntax:

void draw();

Parameters:

None.

Return Value:

Void.

Remarks:

Finds what needs to be drawn where and draws it.

Requirements:

#include <tonc.h>

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include "main.h"

#include "player.h"

// LEVELS

#include "levels/Stage1Color.h"

#include "levels/Stage1ColorWalls.h"

#include "levels/Stage2ColorCombo.h"

#include "levels/Stage2ColorComboWALL.h"

#include "levels/Stage3NumberCombo.h"

#include "levels/Stage3NumberComboWalls.h"

#include "levels/Stage4.h"

#include "levels/Stage4Walls.h"

#include "levels/Stage5new.h"

#include "levels/Stage5newWalls.h"

#include "levels/Stage6Both.h"

#include "levels/Stage6BothWalls.h"

#include "levels/Stage7Both.h"

#include "levels/Stage7BothWalls.h"

#include "levels/Stage9Both.h"

#include "levels/Stage9Walls.h"

#include "levels/StageTutPnP.h"

#include "levels/StageTutPnPWALLS.h"

// BOXES

#include "boxes/BoxClear.h"

#include "boxes/Box1.h"

#include "boxes/Box2.h"

#include "boxes/Box3.h"

#include "boxes/Box4.h"

#include "boxes/Box5.h"

#include "boxes/Box6.h"

#include "boxes/Box7.h"

#include "boxes/Box8.h"

#include "boxes/Box9.h"

// DROPBOXES

#include "dropbox/DBox1Clear.h"

#include "dropbox/DBox2Clear.h"

#include "dropbox/DBox3Clear.h"

#include "dropbox/DBox4Clear.h"

#include "dropbox/DBox5Clear.h"

#include "dropbox/DBox6Clear.h"

#include "dropbox/DBox7Clear.h"

#include "dropbox/DBox8Clear.h"

#include "dropbox/DBox9Clear.h"

#include "dropbox/DBoxClear.h"

// MISC

#include "other/EndZone.h"

#include "other/MovingBlocks.h"

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# Update

Syntax:

void update();

Parameters:

None.

Return Value:

Void.

Remarks:

Updates the world and sprites.

Requirements:

#include "main.h"

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# Init Levels

Syntax:

void initTutorial();

void initStage1();

void initStage2();

void initStage3();

void initStage4();

void initStage5();

void initStage6();

void initStage7();

void initStage9();

Parameters:

None.

Return Value:

Void.

Remarks:

Clears then populates the map. Holds Each levels sprite locations.

Requirements:

#include "main.h"

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# Create Grid

Syntax:

void createGrid(const unsigned short \*map);

Parameters:

const unsigned short

Return Value:

Void.

Remarks:

creates a 2d array to tell the player where the walls are so 0 = no tile and anything greater than 0 is a tile.

Requirements:

#include "main.h"

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# Add Background

Syntax:

void addBackground(const unsigned short \*wallTiles, const unsigned short \*wallMap, const unsigned short \*floorTiles, const unsigned short \*floorMap);

Parameters:

const unsigned short

Return Value:

Void.

Remarks:

Gets the map tiles. Puts them into memory or replaces previous tiles.

Requirements:

#include "main.h"

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# Move

Syntax:

void move(int changeX, int changeY);

Parameters:

int

Return Value:

Void.

Remarks:

Moves boxes. Checks for collisions to see if player can push or pull. Checks if player can move.

Requirements:

#include "main.h"

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# Merge

Syntax:

bool merge(int boxToMerge);

Parameters:

iny

Return Value:

true or false.

Remarks:

Checks if boxes can merge. Uses isOdd function.

(red = 1, orange = 2, yellow = 3, green = 4, blue = 5, purple = 6)

Only primary colors and numbers <= 9.

Requirements:

#include "main.h"

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# Is Odd

Syntax:

bool isOdd(int n);

Parameters:

int

Return Value:

true or false.

Remarks:

Checks if palette bank = an odd number.

Requirements:

#include "main.h"

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# Correct Box

Syntax:

bool correctBox(Box box, Box dropbox);

Parameters:

Box struct, and dropbox struct

Return Value:

true or false.

Remarks:

Checks if the player has placed the correct box in drop box, checks colors and values are the right ones.

Requirements:

#include "main.h"

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# Attract Mode Init

Syntax:

void attractModeInit();

Parameters:

None.

Return Value:

void.

Remarks:

Initialises attract mode Clears then populates the map. Holds Each levels sprite locations. Also create an array of coordinates for the scripted movement.

Requirements:

#include "main.h"

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# Attract Mode Update

Syntax:

void attractModeUdate(int step);

Parameters:

int

Return Value:

void.

Remarks:

It goes through the array and sets the new positions of the srites being moved by the script.

Requirements:

#include "main.h"

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

# Collision

Syntax:

int collision(int x, int y, bool mergeBoxes);

Parameters:

int and bool

Return Value:

int

Remarks:

Returns the index of the box that the player is colliding with.

Requirements:

#include "main.h"