

DoomCopy

Generated by Doxygen 1.8.18

1 Namespace Index	1
1.1 Namespace List	1
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	5
3.1 Class List	5
4 Namespace Documentation	7
4.1 DoomCopy Namespace Reference	7
4.1.1 Detailed Description	8
4.1.2 Function Documentation	8
4.1.2.1 numpad()	8
5 Class Documentation	9
5.1 DoomCopy::Array2D< T > Class Template Reference	9
5.1.1 Constructor & Destructor Documentation	9
5.1.1.1 Array2D()	9
5.1.1.2 ~Array2D()	10
5.2 DoomCopy::Blocks Class Reference	10
5.2.1 Detailed Description	10
5.2.2 Member Function Documentation	10
5.2.2.1 getTextureName()	10
5.2.2.2 isTypeSolid()	11
5.2.2.3 load()	11
5.3 DoomCopy::BlockType Class Reference	11
5.3.1 Detailed Description	12
5.4 DoomCopy::cmpr Class Reference	12
5.4.1 Detailed Description	12
5.4.2 Constructor & Destructor Documentation	12
5.4.2.1 cmpr()	12
5.4.3 Member Function Documentation	13
5.4.3.1 nearby()	13
5.5 DoomCopy::Creature Class Reference	13
5.5.1 Detailed Description	14
5.5.2 Constructor & Destructor Documentation	14
5.5.2.1 Creature()	14
5.5.3 Member Function Documentation	15
5.5.3.1 damage()	15
5.5.3.2 dmgPlayer()	15
5.5.3.3 getPos()	15
5.5.3.4 heal()	16
5.5.3.5 isPlayerVisble()	16

5.5.3.6 move()	16
5.5.3.7 setPos()	17
5.5.3.8 turn()	17
5.5.3.9 update()	17
5.6 DoomCopy::Exit Class Reference	18
5.6.1 Detailed Description	18
5.7 DoomCopy::Game Class Reference	18
5.7.1 Detailed Description	19
5.7.2 Constructor & Destructor Documentation	19
5.7.2.1 Game()	19
5.7.3 Member Function Documentation	20
5.7.3.1 command()	20
5.7.3.2 deleteDeadOrNonExistent()	20
5.7.3.3 getResHeight()	20
5.7.3.4 getResWidth()	21
5.7.3.5 getScreenHeight()	21
5.7.3.6 getScreenWidth()	21
5.7.3.7 loadSettings()	21
5.7.3.8 renderEnemies()	21
5.7.3.9 renderProjectiles()	21
5.7.3.10 setResHeight()	21
5.7.3.11 setResWidth()	22
5.7.3.12 setScreenHeight()	22
5.7.3.13 setScreenWidth()	22
5.7.3.14 startCLIgame()	23
5.7.3.15 startGraphicalGame()	23
5.7.3.16 winCondition()	23
5.8 DoomCopy::GStartGame Class Reference	24
5.8.1 Detailed Description	24
5.9 DoomCopy::HighScore Class Reference	24
5.9.1 Detailed Description	24
5.10 DoomCopy::List< T > Class Template Reference	25
5.10.1 Detailed Description	25
5.10.2 Member Function Documentation	25
5.10.2.1 addItem()	25
5.10.2.2 addrAt()	26
5.10.2.3 at()	26
5.10.2.4 deleteAt() [1/2]	26
5.10.2.5 deleteAt() [2/2]	27
5.10.2.6 doesItContain()	27
5.10.2.7 pSearchBy()	27
5.10.2.8 searchBy()	28

5.10.2.9 setDestructFunction()	28
5.10.2.10 setHead()	28
5.10.2.11 sort()	29
5.11 DoomCopy::ListItem< T > Class Template Reference	29
5.11.1 Detailed Description	29
5.12 DoomCopy::MainMenu Class Reference	30
5.12.1 Detailed Description	30
5.13 DoomCopy::Map Class Reference	30
5.13.1 Detailed Description	31
5.13.2 Constructor & Destructor Documentation	31
5.13.2.1 Map()	31
5.13.3 Member Function Documentation	31
5.13.3.1 load()	31
5.13.3.2 loadEnemies()	32
5.14 DoomCopy::MapHighScore Class Reference	32
5.14.1 Detailed Description	32
5.15 DoomCopy::MeleeEnemy Class Reference	33
5.15.1 Constructor & Destructor Documentation	33
5.15.1.1 MeleeEnemy()	33
5.16 DoomCopy::Menu Class Reference	34
5.16.1 Detailed Description	34
5.16.2 Constructor & Destructor Documentation	34
5.16.2.1 Menu()	34
5.16.3 Member Function Documentation	35
5.16.3.1 addItem()	35
5.16.3.2 getItemAt()	35
5.17 DoomCopy::Player Class Reference	35
5.17.1 Detailed Description	36
5.17.2 Constructor & Destructor Documentation	36
5.17.2.1 Player() [1/3]	36
5.17.2.2 Player() [2/3]	36
5.17.2.3 Player() [3/3]	37
5.17.3 Member Function Documentation	37
5.17.3.1 getPosX()	37
5.17.3.2 getPosY()	38
5.17.3.3 move()	38
5.17.3.4 setPosX() [1/2]	38
5.17.3.5 setPosX() [2/2]	38
5.17.3.6 setPosY() [1/2]	40
5.17.3.7 setPosY() [2/2]	40
5.18 DoomCopy::Point Class Reference	40
5.19 DoomCopy::Projectile Class Reference	41

5.19.1 Detailed Description	41
5.19.2 Constructor & Destructor Documentation	41
5.19.2.1 Projectile()	41
5.19.3 Member Function Documentation	42
5.19.3.1 getStillExists()	42
5.19.3.2 update()	42
5.20 DoomCopy::ProjectileType Class Reference	42
5.20.1 Detailed Description	43
5.20.2 Member Function Documentation	43
5.20.2.1 loadProjectileType()	43
5.21 DoomCopy::Ray Class Reference	43
5.21.1 Detailed Description	44
5.21.2 Member Function Documentation	44
5.21.2.1 distanceFromCollision()	44
5.22 DoomCopy::ScreenSettings Class Reference	44
5.22.1 Detailed Description	45
5.23 DoomCopy::SetResHeight Class Reference	45
5.23.1 Detailed Description	45
5.24 DoomCopy::SetResWidth Class Reference	46
5.24.1 Detailed Description	46
5.25 DoomCopy::SetScreenHeight Class Reference	46
5.25.1 Detailed Description	47
5.26 DoomCopy::SetScreenWidth Class Reference	47
5.26.1 Detailed Description	47
5.27 DoomCopy::Settings Class Reference	47
5.27.1 Detailed Description	48
5.28 DoomCopy::Start Class Reference	48
5.28.1 Detailed Description	48
5.29 DoomCopy::StringManager Class Reference	48
5.29.1 Member Function Documentation	49
5.29.1.1 get_substring_after_first_occurence_of_and_last_occurence_of()	49
5.29.1.2 get_substring_after_last_occurence_of_and_first_occurence_of()	49
5.29.1.3 get_substring_btwn_first_and_next()	50
5.29.1.4 get_substring_btwn_first_occurences()	50
5.29.1.5 get_substring_btwn_last_occurences()	51
5.29.1.6 string_to_double()	51
5.30 DoomCopy::Texture Class Reference	51
5.30.1 Detailed Description	52
5.30.2 Constructor & Destructor Documentation	52
5.30.2.1 Texture()	52
5.30.3 Member Function Documentation	52
5.30.3.1 addImg()	53

5.30.3.2	getTextureSize()	53
5.30.3.3	sampleFrom()	53
5.31	DoomCopy::Weapon Class Reference	54
5.31.1	Detailed Description	54
5.31.2	Constructor & Destructor Documentation	54
5.31.2.1	Weapon() [1/2]	54
5.31.2.2	Weapon() [2/2]	54
5.31.3	Member Function Documentation	55
5.31.3.1	gShot()	55
5.31.3.2	loadWeapon()	55
5.31.3.3	set()	55
5.31.3.4	updateState()	56
Index		57

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

DoomCopy	7
-----------------	-------	---

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

DoomCopy::Array2D< T >	9
DoomCopy::Array2D< char >	9
DoomCopy::Array2D< int >	9
DoomCopy::Map	30
DoomCopy::Array2D< sf::Vertex * >	9
DoomCopy::Blocks	10
DoomCopy::BlockType	11
DoomCopy::cmpr	12
DoomCopy::Creature	13
DoomCopy::MeleeEnemy	33
DoomCopy::Game	18
DoomCopy::List< T >	25
DoomCopy::List< DoomCopy::BlockType >	25
DoomCopy::List< DoomCopy::Creature * >	25
DoomCopy::List< DoomCopy::Projectile * >	25
DoomCopy::ListItem< T >	29
DoomCopy::ListItem< DoomCopy::BlockType >	29
DoomCopy::ListItem< DoomCopy::Creature * >	29
DoomCopy::ListItem< DoomCopy::Projectile * >	29
DoomCopy::Menu	34
DoomCopy::Exit	18
DoomCopy::GStartGame	24
DoomCopy::HighScore	24
DoomCopy::MainMenu	30
DoomCopy::MapHighScore	32
DoomCopy::ScreenSettings	44
DoomCopy::SetResHeight	45
DoomCopy::SetResWidth	46
DoomCopy::SetScreenHeight	46
DoomCopy::SetScreenWidth	47
DoomCopy::Settings	47
DoomCopy::Start	48
DoomCopy::Player	35
DoomCopy::Point	40

DoomCopy::Projectile	41
DoomCopy::ProjectileType	42
DoomCopy::Ray	43
DoomCopy::StringManager	48
DoomCopy::Texture	51
DoomCopy::Weapon	54

Chapter 3

Class Index

3.1 Class List

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

DoomCopy::Array2D< T >	9
DoomCopy::Blocks	10
DoomCopy::BlockType	11
DoomCopy::cmp	12
DoomCopy::Creature	13
DoomCopy::Exit	
Kilepesi menu	18
DoomCopy::Game	
Maga a jatek, ez kezeli mindent, és éppen történik minden	18
DoomCopy::GStartGame	24
DoomCopy::HighScore	
MapHighScore (p. 32) menüpontokat tartalmazó menu	24
DoomCopy::List< T >	25
DoomCopy::ListItem< T >	29
DoomCopy::MainMenu	
Fomenu, ha a játék grafikus módban indul, akkor az ablakban ezt a menüt láthatjuk először	30
DoomCopy::Map	30
DoomCopy::MapHighScore	
Egy pályához tartozó high score-okat tölti be, majd kiválasztása után írja ki	32
DoomCopy::MeleeEnemy	33
DoomCopy::Menu	34
DoomCopy::Player	35
DoomCopy::Point	40
DoomCopy::Projectile	
Lovedek osztály, rendelkezik lovedek típusal, kezdőpozícióval, irányvektorral, és jelenlegi pozícióval	41
DoomCopy::ProjectileType	42
DoomCopy::Ray	43
DoomCopy::ScreenSettings	
Képernyő beállításait tartalmazó almenu	44
DoomCopy::SetResHeight	
Menüpont ami állítja a felbontás magasságát	45
DoomCopy::SetResWidth	
Menüpont ami állítja a felbontás szélességét	46
DoomCopy::SetScreenHeight	
A menüpont amelyik állítja az ablak magasságát	46

DoomCopy::SetScreenWidth	
A menüpont amelyik atallítja az ablak szélességet	47
DoomCopy::Settings	
Beállításokat tartalmazó menü	47
DoomCopy::Start	48
DoomCopy::StringManager	48
DoomCopy::Texture	51
DoomCopy::Weapon	54

Chapter 4

Namespace Documentation

4.1 DoomCopy Namespace Reference

Classes

- class **Array2D**
- class **Blocks**
- class **BlockType**
- class **cmpr**
- class **Creature**
- class **Exit**

Kilepesi menu.

- class **Game**

Maga a játék, ez kezeli mindent, és éppen történik minden.

- class **GStartGame**
- class **HighScore**

MapHighScore (p. 32) menüpontokat tartalmazó menu.

- class **List**
- class **ListItem**
- class **MainMenu**

Fomenu, ha a játék grafikus modban indul, akkor az ablakban ezt a menüt láthatjuk először.

- class **Map**
- class **MapHighScore**

Egy pályához tartozó high score-okat tölti be, majd kiválasztása után írja ki.

- class **MeleeEnemy**
- class **Menu**
- class **Player**
- class **Point**
- class **Projectile**

Lovedek osztály, rendelkezik lovedek típussal, kezdopozícióval, irányvektorral, és jelenlegi pozícióval.

- class **ProjectileType**
- class **Ray**
- class **ScreenSettings**

Kepernyő beállításait tartalmazó almenu.

- class **SetResHeight**

Menüpont ami atallítja a felbontás magasságát.

- class **SetResWidth**

Menupont ami atallítja a felbontás szélességet.

- class **SetScreenHeight**

A menupont amelyik atallítja az ablak magasságát.

- class **SetScreenWidth**

A menupont amelyik atallítja az ablak szélességet.

- class **Settings**

Beállításokat tartalmazó menu.

- class **Start**
- class **StringManager**
- class **Texture**
- class **Weapon**

Functions

- void **numpad** (sf::RenderWindow *window, int *variableToSaveInto, std::string textToWriteOut)

4.1.1 Detailed Description

Generikus ketdimenziós tömb

Parameters

<i>T</i>	- adattípus
----------	-------------

4.1.2 Function Documentation

4.1.2.1 numpad()

```
void DoomCopy::numpad (
    sf::RenderWindow * window,
    int * variableToSaveInto,
    std::string textToWriteOut )
```

4 menu is ezt a függvényt használja. Gyakorlatilag egy numpad, mint ahogy azt a neve is sugallja. Egy ablakon érzékeli a számok beütését, egy tetszőleges szöveg után kiírja azokat, és ha entert nyomnak, akkor az új számot (amit beütöttek) beírja egy a függvénynek átadott változóba.

Parameters

<i>window</i>	- ablak amibe a numpad üzemel
<i>textToWriteOut</i>	- a szöveg amiket a számok elé kiír
<i>variableToSaveInto</i>	- a változó amibe az új értéket elmenti.

Chapter 5

Class Documentation

5.1 DoomCopy::Array2D< T > Class Template Reference

Public Member Functions

- **Array2D** (int rows=0, int columns=0)
- void **free** ()
- virtual **~Array2D** ()

Public Attributes

- T ** **data**
- int **rows**
- int **columns**

5.1.1 Constructor & Destructor Documentation

5.1.1.1 Array2D()

```
template<class T >
DoomCopy::Array2D< T >:: Array2D (
    int rows = 0,
    int columns = 0 ) [inline]
```

A ketdimenzios tomb konstruktore

Parameters

<i>rows</i>	- a ketdimenzios tombben levo sorok szama
<i>columns</i>	- a ketdimenzios tombben levo oszlopok szama

5.1.1.2 ~Array2D()

```
template<class T >
virtual DoomCopy::Array2D< T >::~~ Array2D ( ) [inline], [virtual]
```

Virtualis destruktorkat Felszabadítja a dinamikusan lefoglalt memóriaterületeket

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Array2D.h

5.2 DoomCopy::Blocks Class Reference

```
#include <BlockType.h>
```

Public Member Functions

- **Blocks** (std::string path, **Texture** &text)
- void **load** (std::string path, **Texture** &)
- bool **isTypeSolid** (int type) const
- std::string **getTextureName** (int type)
- void **printTypes** ()
Szabványos kimenetre kiírja a blokkszámokat külön sorokba.

Public Attributes

- **List< BlockType > blockType**

5.2.1 Detailed Description

Blokk típusból álló láncolt lista, és a hozzá tartozó műveletek és függvények.

5.2.2 Member Function Documentation

5.2.2.1 getTextureName()

```
std::string DoomCopy::Blocks::getTextureName (
    int type )
```

Egy (blokk) számhoz megadja a hozzátartozó textúra nevét (felteve, hogy van)

Public Member Functions

- **BlockType** (int x, bool isSolid, const char *textName)

Public Attributes

- int **x**
- bool **isSolid**
- std::string **textureName**

5.3.1 Detailed Description

Blokktipus - eltarolja egy adott blokk nevet, szamat, texturajat, ha van, es azt, hogy az adott blokk szilard-e vagy sem.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/BlockType.h

5.4 DoomCopy::cmpr Class Reference

```
#include <Projectile.h>
```

Public Member Functions

- **cmpr** (**Point** a)

Static Public Member Functions

- static bool **nearby** (**Creature** *t)

Static Public Attributes

- static **Point** **p** = **Point**(0,0)

5.4.1 Detailed Description

Egy funktor arra, hogy meg lehessen vizsgalni, hogy egy lovedek egy szornyhoz eleg közel van-e ahhoz, hogy "eltalálja" azt.

5.4.2 Constructor & Destructor Documentation

5.4.2.1 cmpr()

```
DoomCopy::cmpr::cmpr (
    Point a ) [inline]
```

Beallítja a vizsgalando lovedek jelenlegi poziciojat.

Parameters

a	- lovedek pozicioja
-----	---------------------

5.4.3 Member Function Documentation

5.4.3.1 nearby()

```
bool DoomCopy::cmpr::nearby (
    DoomCopy::Creature * t ) [static]
```

Megvizsgalja adott szornyre, hogy eltalalta-e azt a lovedek

Parameters

t	- szorny
-----	----------

Returns

- talalt-e a lovedek

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Projectile.h
- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Projectile.cpp

5.5 DoomCopy::Creature Class Reference

```
#include <Creature.h>
```

Inherited by **DoomCopy::MeleeEnemy**.

Public Member Functions

- **Creature** (**Point** pos, double HP, **Point** direction, double FOV, double viewDistance)
- virtual void **setPos** (**Point** newPos)
- virtual **Point** **getPos** ()
- virtual void **damage** (double amount)
- virtual void **dmgPlayer** (**Player** &player, double amount)
- virtual bool **isAlive** ()
- virtual void **heal** (double amount)
- virtual double **getHP** ()
- virtual void **turn** (double radian)
- virtual double **getDirection** ()
- virtual void **move** (double relativeX, double relativeY, const **DoomCopy::Map** &map)
- virtual void **setSpeed** (double newSpeed)
- bool **isPlayerVisible** (const **Map** &map, **Point** playerPos)
- virtual void **update** (const **Map** &map, **Player** &player)

Public Attributes

- Status **currentStatus** = stand
- bool **visible** = false
- Texture **text**
- sf::VertexArray **vertexArray**
- double **viewDistance**
- double **FOV**
- double **distanceFromPlayer**

Protected Attributes

- double **HP**
- Point **position**
- Point **directionVector**

5.5.1 Detailed Description

A játékhoz tartozó szorny osztály

Parameters

<i>alive</i>	- el-e a szorny (életpontjai 0 fölött vannak)
<i>speed</i>	- mozgási sebesség
<i>currentHP</i>	- jelenlegi életpontok száma
<i>HP</i>	- eredeti/maximum életpontok száma
<i>currentStatus</i>	- jelenlegi statusza (mozgás/támadás/állás/stb)
<i>position</i>	- szorny pozíciója
<i>directionVector</i>	- irányvektor - ebbe az irányba néz, és mozog
<i>visible</i>	- látja-e a játékos
<i>distanceFromPlayer</i>	- játékostól való távolság - ez alapján van a szorny lista rendezve
<i>text</i>	- textúra
<i>FOV</i>	- szorny látószöge
<i>viewDistance</i>	- szorny látótávolsága

5.5.2 Constructor & Destructor Documentation

5.5.2.1 Creature()

```
DoomCopy::Creature::Creature (
    Point pos,
    double HP,
    Point direction,
    double FOV,
    double viewDistance ) [inline]
```

Szorny konstruktor. Beállítja a kezdeti helyzetet, kezdőirányt, látószöget, és látótávolságot.

Parameters

<i>pos</i>	- kezdó pozíció
<i>direction</i>	- kezdó irány
<i>viewDistance</i>	- látótávolság
<i>HP</i>	- kezdő/max életpontok száma

5.5.3 Member Function Documentation

5.5.3.1 damage()

```
virtual void DoomCopy::Creature::damage (
    double amount ) [inline], [virtual]
```

Adott mennyiséget sebez ha ezzel a szörny életpontja nulla, vagy annal kevesebb, akkor a szörny elpusztul

Parameters

<i>amount</i>	- sebzés merete
---------------	-----------------

5.5.3.2 dmgPlayer()

```
void DoomCopy::Creature::dmgPlayer (
    Player & player,
    double amount ) [virtual]
```

Megsebzi a játékost

Parameters

<i>player</i>	- játékos
<i>amount</i>	- sebzés merete

5.5.3.3 getPos()

```
virtual Point DoomCopy::Creature::getPos ( ) [inline], [virtual]
```

Visszaadja a szörny jelenlegi pozícióját

Returns

- szorny jelenlegi pozicioja

5.5.3.4 heal()

```
virtual void DoomCopy::Creature::heal (
    double amount ) [inline], [virtual]
```

Visszatolti a szorny egy adott mennyisegu eletpontjat, de ha az tobb vagy annyi, mint a max eletpontjainak szama , akkor szorny a kezdo eletpontjat kapja meg.

Parameters

<i>amount</i>	- eletviszzatoltes merteke
---------------	----------------------------

5.5.3.5 isPlayerVisble()

```
bool DoomCopy::Creature::isPlayerVisble (
    const Map & map,
    Point playerPos )
```

Latja-e a szorny a jatekost

Parameters

<i>map</i>	- palya
<i>playerPos</i>	- jatekos pozicioja

Returns

- lathato-e a jatekos

5.5.3.6 move()

```
void DoomCopy::Creature::move (
    double relativeX,
    double relativeY,
    const DoomCopy::Map & map ) [virtual]
```

A szorny jelenlegipoziciojahoz kepest elmozgatja azt valamilyen x, y irányban felteve, hogy kozben nem utkozik semmilyen falba.

Parameters

<i>map</i>	- palya (falakba utkozes vizsgalata miatt van ra szukseg)
<i>relativeX</i>	- pozicio valtozasa x iranyban
<i>relativY</i>	- pozicio valtozasa y iranyban

5.5.3.7 setPos()

```
virtual void DoomCopy::Creature::setPos (
    Point newPos ) [inline], [virtual]
```

Barmilyen vizsgalat nelkul csak atallitja a szorny poziciojat egy masikra.

Parameters

<i>newPos</i>	- uj pozicio
---------------	--------------

5.5.3.8 turn()

```
virtual void DoomCopy::Creature::turn (
    double radian ) [inline], [virtual]
```

A szorny iranyvektorat elforditja egy radianban megadott szoggel

Parameters

<i>radian</i>	- szog radianban
---------------	------------------

5.5.3.9 update()

```
void DoomCopy::Creature::update (
    const Map & map,
    Player & player ) [virtual]
```

Frissiti a szornyet, mozgatja, "tamadtatja", valtoztatja annak jelenlegi statuszat. Valojaban itt ez felel meg leginkabb a mesterseges intelligencianak, pedig valojaban csak mesterseges.

Parameters

<i>map</i>	- palya
<i>player</i>	- jatekos

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Creature.h
- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Creature.cpp

5.6 DoomCopy::Exit Class Reference

Kilepesi menu.

```
#include <Game.h>
```

Inherits **DoomCopy::Menu**.

Public Member Functions

- **Exit** (**Game** &game, **Menu** *back)
- void **doAction** ()

Az a fuggveny ami a menupont kivalasztasanal lefut. Ezt kell minden uj menunek felulirnia.

Public Attributes

- **Game** * **game**

5.6.1 Detailed Description

Kilepesi menu.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.h

5.7 DoomCopy::Game Class Reference

Maga a jatek, ez kezeli mindent, es ebben tortenik minden.

```
#include <Game.h>
```

Public Member Functions

- **Game** (bool cli)
- void **startGraphicalGame** (const char *mapName, **Point** resolution, **Point** screenSize)
- void **renderWalls** ()

Kiszamolja, hogy mit lat a jatekos: milyen blokknak, melyik reszet, melyik texturat.
- void **renderEnemies** ()
- void **renderProjectiles** ()
- void **renderWeapon** ()

Beallitja a jatekos fegyverenek allapota alapjan, hogy a fegyver melyik texturajat rajzolja majd ki.
- void **deleteDeadOrNonExistent** ()
- void **loadSettings** ()
- void **winCondition** (std::string mapName, int &x, int &y)
- void **startCLIGame** (const char *mapName)
- bool **command** (const std::string &cmd, bool &exit, int &aim)
- void **setScreenWidth** (int newWidth)
- void **setScreenHeight** (int newHeight)
- void **setResWidth** (int newRWidth)
- void **setResHeight** (int newRHeight)
- int **getScreenWidth** ()
- int **getScreenHeight** ()
- int **getResWidth** ()
- int **getResHeight** ()

Public Attributes

- sf::RenderWindow * **window** = NULL
- sf::Font **font**

5.7.1 Detailed Description

Maga a jatek, ez kezeli mindent, es ebben tortenik minden.

5.7.2 Constructor & Destructor Documentation

5.7.2.1 Game()

```
DoomCopy::Game::Game (
    bool cli )
```

Elindítja a jatekot vagy grafikusan vagy CLI-ben futva.

Parameters

<i>cli</i>	- cli-ben induljon-e a jatek
------------	------------------------------

5.7.3 Member Function Documentation

5.7.3.1 command()

```
bool DoomCopy::Game::command (
    const std::string & cmd,
    bool & exit,
    int & aim )
```

Ertelmezi a kovetkezo parancsokat: quit, shoot, aim, move, printmap, listmonsters, hp Akcio tipusu parancsok: shoot, move Nem akcio tipusu parancsok: aim, quit, printmap, hp, listmonsters

Parameters

<i>cmd</i>	- parancs
<i>aim</i>	- a becolazando szorny indexe
<i>exit</i>	- kilepjen-e a jatekbol

Returns

- akcio tipusu parancs lett-e beirva

5.7.3.2 deleteDeadOrNonExistent()

```
void DoomCopy::Game::deleteDeadOrNonExistent ( )
```

Kitorli a lovedek listabol azokat a lovedekeket, amik mar nem leteznek (bele utkoztek valamibe, vagy tul mentek a range-ukon). Es kitorli azokat a szornyeket a szorny listabol, melyek mar meghaltak.

5.7.3.3 getResHeight()

```
int DoomCopy::Game::getResHeight ( ) [inline]
```

Visszaadja a felbontas magassagat

Returns

- felbontas magassaga

5.7.3.4 getResWidth()

```
int DoomCopy::Game::getResWidth ( ) [inline]
```

Visszaadja a felbontás szélességet

Returns

- felbontás szélessége

5.7.3.5 getScreenHeight()

```
int DoomCopy::Game::getScreenHeight ( ) [inline]
```

Visszaadja az ablak magasságát

Returns

- ablak magassága

5.7.3.6 getScreenWidth()

```
int DoomCopy::Game::getScreenWidth ( ) [inline]
```

Visszaadja az ablak szélességet

Returns

- ablak szélessége

5.7.3.7 loadSettings()

```
void DoomCopy::Game::loadSettings ( )
```

Betölti a controls.conf konfigurációs fájlból a kezdő beállításokat, mint az irányítást, felbontást, és képernyő méretet

5.7.3.8 renderEnemies()

```
void DoomCopy::Game::renderEnemies ( )
```

Vegig fut a pályás szörny listáján, és megfelelő helyre, a szörny állapotának megfelelő texturájú szörnyet rajzol ki, ha az a játékos látószögben benne van, nem takarja ki blokk, és nincs távolabb, mint a játékos látótávolsága.

5.7.3.9 renderProjectiles()

```
void DoomCopy::Game::renderProjectiles ( )
```

Vegig fut a pályás lövedék listáján, és a megfelelő helyekre kirajzolja a lövedékeket, ha azok a játékos látószögbe esnek, nincsenek távolabb, mint a játékos látótávolsága, és nem takarja el őket egy (vagy több) blokk.

5.7.3.10 setResHeight()

```
void DoomCopy::Game::setResHeight (
    int newRHeight ) [inline]
```

Beállít a felbontásnak egy új magasságot (felteve hogy az nagyobb mint 100)

Parameters

<i>newRHeight</i>	- felbontas uj magassaga
-------------------	--------------------------

5.7.3.11 setResWidth()

```
void DoomCopy::Game::setResWidth (
    int newRWidth ) [inline]
```

Beallit a felbontasnak egy uj szelesseget (felteve hogy az nagyobb mint 100)

Parameters

<i>newRWidth</i>	- felbontas uj szelessege
------------------	---------------------------

5.7.3.12 setScreenHeight()

```
void DoomCopy::Game::setScreenHeight (
    int newHeight ) [inline]
```

Beallit grafikus futasban az ablaknak egy uj magassagot, felteve, hogy az nagyobb, mint 100

Parameters

<i>newHeight</i>	- az ablak uj magassaga
------------------	-------------------------

5.7.3.13 setScreenWidth()

```
void DoomCopy::Game::setScreenWidth (
    int newWidth ) [inline]
```

Beallit grafikus futasban az ablaknak egy uj szelesseget, felteve, hogy az nagyobb, mint 100

Parameters

<i>newWidth</i>	- az ablak uj szelessege
-----------------	--------------------------

5.7.3.14 startCLIGame()

```
void DoomCopy::Game::startCLIGame (
    const char * mapName )
```

Elindítja a jatekot a mapName nevű pályával CLI-s modban.

Parameters

<i>mapName</i>	- palya neve
----------------	--------------

5.7.3.15 startGraphicalGame()

```
void DoomCopy::Game::startGraphicalGame (
    const char * mapName,
    Point resolution,
    Point screenSize )
```

Elindítja a jatekot grafikusan egy palyan, adott ablakmerettel, es felbontassal.

Parameters

<i>mapName</i>	- palya neve
<i>resolution</i>	- felbontas
<i>screenSize</i>	- ablak merete

5.7.3.16 winCondition()

```
void DoomCopy::Game::winCondition (
    std::string mapName,
    int & x,
    int & y )
```

Betölti egy palya mappájában lévő whereto.conf konfigurációs fájlt, ami tartalmazza azt a pozíciót, ahova a játékosnak el kell jutnia azért, hogy a pályán végig jusson.

Parameters

x	- ide tolti be az elerendo cel x koordinatajat
y	- ide tolti be az elerendo cel y koordinatajat
mapName	- a palya neve, aminek a mappajaban a whereto.conf talalhato

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.h
- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.cpp

5.8 DoomCopy::GStartGame Class Reference

```
#include <Game.h>
```

Inherits **DoomCopy::Menu**.

Public Member Functions

- **GStartGame** (**Game** &game, std::string name, **Menu** *back=NULL)
- void **doAction** ()
Az a fuggveny ami a menupont kivalasztasanal lefut. Ezt kell minden uj menunek felulirnia.

Public Attributes

- **Game** * **game**

5.8.1 Detailed Description

Egy palya elinditasara szolgalo menupont. Elinditja a menuponttal azonos nevu palya grafikusan.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.h

5.9 DoomCopy::HighScore Class Reference

MapHighScore (p. 32) menupontokat tartalmaz menu.

```
#include <Game.h>
```

Inherits **DoomCopy::Menu**.

Public Member Functions

- **HighScore** (**Menu** *back, **Game** &game)

Public Attributes

- **Game** * **game**

5.9.1 Detailed Description

MapHighScore (p. 32) menupontokat tartalmaz menu.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.h

5.10 DoomCopy::List< T > Class Template Reference

```
#include <List.h>
```

Public Member Functions

- **List** (T first)
- void **setHead** (T first)
- void **addItem** (T newItem)
- void **sort** (bool fnc(T i, T i2))
- **ListItem**< T > * **getHead** () const
- T **at** (size_t idx)
- **ListItem**< T > * **addrAt** (size_t idx)
- T **searchBy** (bool fnc(T))
- **ListItem**< T > * **pSearchBy** (bool fnc(T))
- bool **doesItContain** (bool fnc(T))
- void **deleteAt** (size_t idx)
- void **deleteAt** (size_t idx, void fnc(T))
- void **setDestructFunction** (void fnc(T))

Public Attributes

- size_t **currentSize** = 0

5.10.1 Detailed Description

```
template<typename T>
class DoomCopy::List< T >
```

Automatikus lancolt lista sablon hasonlo az std::vector-hoz, csak minden szempontbol rosszabb.

Parameters

<i>T</i>	- adattípus
----------	-------------

5.10.2 Member Function Documentation

5.10.2.1 addItem()

```
template<typename T >
void DoomCopy::List< T >::addItem (
    T newItem ) [inline]
```

Hozzaad egy uj elemet a lancolt listahoz

Parameters

<i>newItem</i>	- az új elem
----------------	--------------

5.10.2.2 addrAt()

```
template<typename T >
ListItem<T>* DoomCopy::List< T >::addrAt (
    size_t idx ) [inline]
```

Visszaadja a lista egy adott indexen található memória címet

Parameters

<i>idx</i>	- index
------------	---------

Returns

- adott indexű elem memóriacíme

5.10.2.3 at()

```
template<typename T >
T DoomCopy::List< T >::at (
    size_t idx ) [inline]
```

Visszaadja a lista egy adott indexű elemét

Parameters

<i>idx</i>	- index
------------	---------

Returns

- adott indexű elem

5.10.2.4 deleteAt() [1/2]

```
template<typename T >
void DoomCopy::List< T >::deleteAt (
    size_t idx ) [inline]
```

Kitorli a lista egy adott indexen található elemét

Parameters

<i>idx</i>	- kitorlendo elem indexe
------------	--------------------------

5.10.2.5 deleteAt() [2/2]

```
template<typename T >
void DoomCopy::List< T >::deleteAt (
    size_t idx,
    void fncT ) [inline]
```

A lista egy adott indexu elemen lefuttat egy fuggveny (pl.: ha a lanc dinamikusan foglalt elemeket tartalmaz, akkor ez valamilyen delete fgvny) majd kitorli az elemet a listabol.

Parameters

<i>idx</i>	- kitorlendo elem indexe
<i>fnc</i>	- fuggveny ami a torlendo elemen lefut

5.10.2.6 doesItContain()

```
template<typename T >
bool DoomCopy::List< T >::doesItContain (
    bool fncT ) [inline]
```

Egy igaz/hamis fuggveny-t vegig futtat a lista minden elemen, es amint talal akar egyet is igazzal ter vissza.

Parameters

<i>fnc</i>	- fuggveny pointer elem vizsgalasahoz
------------	---------------------------------------

Returns

- tartalmaz-e adott tulajdonsagu elemet

5.10.2.7 pSearchBy()

```
template<typename T >
ListItem<T>* DoomCopy::List< T >::pSearchBy (
    bool fncT ) [inline]
```

Egy fuggveny alapan keres elemet a listabol. Az elso olyan elem memoria cimevel ter vissza ami teljesiti a kereso fuggveny feltetele(i)t.

Parameters

<i>fnc</i>	- függvény ami alapján adott elemet kereseni lehet
------------	--

Returns

- az első elem memóriája amire a biztosított függvény igazsággal tér vissza

5.10.2.8 searchBy()

```
template<typename T >
T DoomCopy::List< T >::searchBy (
    bool fncT ) [inline]
```

Egy függvény alapján keres elemet a listából. Az első olyan elemmel tér vissza ami teljesíti a kereső függvény feltétele(i)t.

Parameters

<i>fnc</i>	- függvény ami alapján adott elemet kereseni lehet
------------	--

Returns

- az első elem amire a biztosított függvény igazsággal tér vissza

5.10.2.9 setDestructFunction()

```
template<typename T >
void DoomCopy::List< T >::setDestructFunction (
    void fncT ) [inline]
```

Beállít egy függvényt ami majd lefut a lista minden elemére, amikor a listának a destruktora fut.

Parameters

<i>fnc</i>	- a függvény
------------	--------------

5.10.2.10 setHead()

```
template<typename T >
void DoomCopy::List< T >::setHead (
    T first ) [inline]
```

Beallitja a lancolt lista elso elemet felteve, hogy a lista eddig teljesen ures volt.

Parameters

<i>first</i>	- az elem amivel kezdeni szeretnenk
--------------	-------------------------------------

5.10.2.11 sort()

```
template<typename T >
void DoomCopy::List< T >::sort (
    bool fncT i, T i2 ) [inline]
```

Quicksort ami az egesz lista lefut

Parameters

<i>fnc</i>	- összehasonlito fuggveny ami alapjan a rendezes tortenik
------------	---

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/List.h

5.11 DoomCopy::ListItem< T > Class Template Reference

```
#include <List.h>
```

Public Member Functions

- **ListItem** (T item)

Public Attributes

- T item
- **ListItem** * next

5.11.1 Detailed Description

```
template<typename T>
class DoomCopy::ListItem< T >
```

Listitem - lancolt lista egysegeleme

Parameters

<i>T</i>	- adattipus
----------	-------------

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/List.h

5.12 DoomCopy::MainMenu Class Reference

Fomenu, ha a jatek grafikus modban indul, akkor az ablakban ezt a menut lathatjuk eloszor.

```
#include <Game.h>
```

Inherits **DoomCopy::Menu**.

Public Member Functions

- **MainMenu** (std::string name, **Game** &game)

Public Attributes

- **Game** * **game**

5.12.1 Detailed Description

Fomenu, ha a jatek grafikus modban indul, akkor az ablakban ezt a menut lathatjuk eloszor.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.h

5.13 DoomCopy::Map Class Reference

```
#include <Map.h>
```

Inherits **DoomCopy::Array2D< int >**.

Public Member Functions

- **Map** (std::string pathToMap)
- void **load** (std::string pathToMap)
- void **loadEnemies** (std::string pathToMap)

Public Attributes

- **Texture** `text`
- **Blocks** `blocks`
- **List**< **Creature** * > `enemies`
- **List**< **Projectile** * > `projectiles`
- **Array2D**< char > `CLIMap`

5.13.1 Detailed Description

A palya, egy ketdimenzios szam tomb, ahol minden szam egy blokkot jelent. Rendelkezik egy texturaval, amibe 9 blokktextura fer bele. Ezen kivul a palya rendelkezik egy szorny listval, es egy lovedeklistval.

5.13.2 Constructor & Destructor Documentation

5.13.2.1 Map()

```
DoomCopy::Map::Map (
    std::string pathToMap )
```

Betolti a palyat, es minden mast ami a palyahoz tartozik, pl a szornyeket a szorny listaba

Parameters

<i>pathToMap</i>	- a palya mappa eleresi utvonala
------------------	----------------------------------

5.13.3 Member Function Documentation

5.13.3.1 load()

```
void DoomCopy::Map::load (
    std::string pathToMap )
```

Betolti a palyat, es minden mast ami a palyahoz tartozik, pl a szornyeket a szorny listaba

Parameters

<i>pathToMap</i>	- a palya mappa eleresi utvonala
------------------	----------------------------------

5.13.3.2 loadEnemies()

```
void DoomCopy::Map::loadEnemies (
    std::string pathToMap )
```

Betolti a pályához tartozó szörnyek a pálya mappájának elérési útvonalára alapján.

Parameters

<i>pathToMap</i>	- a pályát tartalmazó mappa elérési útvonala
------------------	--

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Map.h
- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Map.cpp

5.14 DoomCopy::MapHighScore Class Reference

Egy pályához tartozó high score-okat tölti be, majd kiválasztása után írja ki.

```
#include <Game.h>
```

Inherits **DoomCopy::Menu**.

Public Member Functions

- **MapHighScore** (**Menu** *back, std::string mapName, **Game** &game)
- void **doAction** ()

Az a függvény ami a menüpont kiválasztásánál lefut. Ezt kell minden új menünek felülírnia.

Public Attributes

- **Game** * **game**
- sf::Text **scores** [10]
- int **lineNr** = 0
- std::string **mName**

5.14.1 Detailed Description

Egy pályához tartozó high score-okat tölti be, majd kiválasztása után írja ki.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.h

5.15 DoomCopy::MeleeEnemy Class Reference

Inherits **DoomCopy::Creature**.

Public Member Functions

- **MeleeEnemy** (**Point** position, **Point** direction, double HP, double dmg, double attackSpeed, const std::string &pathToText, const char *textname, double FOV, double viewDistance)

Public Attributes

- double **damage**
- double **attackSpeed**

Additional Inherited Members

5.15.1 Constructor & Destructor Documentation

5.15.1.1 MeleeEnemy()

```
DoomCopy::MeleeEnemy::MeleeEnemy (
    Point position,
    Point direction,
    double HP,
    double dmg,
    double attackSpeed,
    const std::string & pathToText,
    const char * textname,
    double FOV,
    double viewDistance ) [inline]
```

Letrehoz egy adott texturaju, pozicioju, iranyu, eletpontu, sebzesu MeleeEnemy-t.

Parameters

<i>HP</i>	- max/kezdo eletpontok szama
<i>position</i>	- kezdo pozicio
<i>direction</i>	- kezdo irany
<i>FOV</i>	- latoszog
<i>viewDistance</i>	- latotavolsag
<i>pathToText</i>	- textura eleresi utvonala
<i>textname</i>	- textura neve

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/MeleeEnemy.h

5.16 DoomCopy::Menu Class Reference

```
#include <Menu.h>
```

Inherited by `DoomCopy::Exit`, `DoomCopy::GStartGame`, `DoomCopy::HighScore`, `DoomCopy::Main`, `DoomCopy::MapHighScore`, `DoomCopy::ScreenSettings`, `DoomCopy::SetResHeight`, `DoomCopy::SetResWidth`, `DoomCopy::SetScreenHeight`, `DoomCopy::SetScreenWidth`, `DoomCopy::Settings`, and `DoomCopy::Start`.

Public Member Functions

- `Menu` (`size_t` arraySize, `std::string` name, `Menu *back=NULL`)
- virtual void `doAction` ()
Az a fuggveny ami a menupont kivlasztasanal lefut. Ezt kell minden uj menunek felulirnia.
- virtual void `addItem` (`Menu *newItem`)
- virtual `Menu *` `getItemAt` (`size_t` idx)

Public Attributes

- `Menu *` `back`
- `Menu **` `options`
- `size_t` `nr_of_menus` = 0
- `size_t` `arraySize`
- `std::string` `text`
- `sf::Text` `gText`

5.16.1 Detailed Description

A `Menu` (p.34) osztaly a kovetkezo modon mukodik: rendelkezik egy `doAction` fuggvennyel, mely egy adott menupont kivlasztasa utan lefut, felteve, hogy a kivlasztott menupont nem rendelkezik almenukkel. A `Menu` (p.34) osztaly rendelkezik egy ebbol az abstract osztalybol szarmazo heterogen kollekcioval. Ez a heterogen kollekcio valojaban az almenuk pointereibol allo dinamikus tomb. Ezenkivul minden menu rendelkezik meg egy menu pointerrel, ami vagy `NULL` (ez esetben a menu nem submenu), vagy pedig arra a menure mutat, amiben ez a menu egy submenu. Mas szavakkal: a `Menu` (p.34) osztaly valojaban egy funktor, es minden menu/menupont ennek a leszarmazottja, es minden szarmaztataskor a menu viselkedeset a `doAction` fuggveny atirasaval lehet megvaltoztatni. Es azert, hogy barmilyen menu tartalmazhasson barmilyen menupontot/almenut, ezert minden menu rendelkezik egy menukbol allo heterogen kollekcioval.

5.16.2 Constructor & Destructor Documentation

5.16.2.1 Menu()

```
DoomCopy::Menu::Menu (
    size_t arraySize,
    std::string name,
    Menu * back = NULL ) [inline]
```

Letrehoz egy menut/menupontot egy adott merettel (ha 0, akkor nincs almenuje -> ilyenkore menupont), adott nevel, es egy back pointerrel, ami az elozo menure mutat ami ezt tartalmazza, ha van ilyen.

Parameters

<i>arraySize</i>	- menu meret, ha 0, akkor nem lehet hozza menupontokat adni
<i>name</i>	- a menu neve
<i>back</i>	- pointer az elozo menure, ha van olyan

5.16.3 Member Function Documentation

5.16.3.1 addItem()

```
virtual void DoomCopy::Menu::addItem (
    Menu * newItem ) [inline], [virtual]
```

Hozzad egy almenut/menupontot a menuhoz

Parameters

<i>newItem</i>	- menupont/almenu pointer
----------------	---------------------------

5.16.3.2 getItemAt()

```
virtual Menu* DoomCopy::Menu::getItemAt (
    size_t idx ) [inline], [virtual]
```

Visszaadja egy adott indexu almenu memoriacimet

Parameters

<i>idx</i>	- almenu/menupont indexe
------------	--------------------------

Returns

- almenu/menupont memoriacime

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Menu.h

5.17 DoomCopy::Player Class Reference

```
#include <Player.h>
```

Public Member Functions

- **Player** (double x, double y)
- **Player** (const char *conf)
- **Player** (double x, double y, double FOV, double direction, double viewDistance)
- void **move** (double relativeX, double relativY, const **DoomCopy::Map** &map)
- void **setPosX** (double nPosX)
- void **setPosY** (double nPosY)
- void **setPosX** (double nPosX, const **DoomCopy::Map** &map)
- void **setPosY** (double nPosY, const **DoomCopy::Map** &map)
- double **getPosX** ()
- double **getPosY** ()

Public Attributes

- **Weapon weapon**
- double **HP** = 100
- double **FOV**
- double **direction**
- double **viewDistance**

5.17.1 Detailed Description

Jatekos - rendelkezik fegyverrel, eletpontokkal, latoszoggel, latotavolsaggal, pozicioval, irannyal. A jatekos az amit a felhasználó irányítani tud, a jatek veget erhet, ha a jatekos meghal.

5.17.2 Constructor & Destructor Documentation

5.17.2.1 Player() [1/3]

```
DoomCopy::Player::Player (
    double x,
    double y ) [inline]
```

Jatekos konstruktora, ami csak a kezdopoziciojat allitja be/allitja at, barmilyen vizsgalat nelkul.

Parameters

<i>x</i>	- jatekos uj x koordinataja
<i>y</i>	- jatekos uj y koordinataja

5.17.2.2 Player() [2/3]

```
DoomCopy::Player::Player (
```

```
const char * conf )
```

Jatekos konstruktora, ami megnyitja a conf eleresi utvonalu fajlt, majd a fajl tartalma alapjan betolti a jatekost, annak fegyvereivel, poziciojaval, irányával, latoszogevel, latotavolsagaval, eletpontjaival együtt. Azaz teljesen.

Parameters

<i>conf</i>	- a jatekos konfiguracios fajljanak eleresi utvonala
-------------	--

5.17.2.3 Player() [3/3]

```
DoomCopy::Player::Player (
    double x,
    double y,
    double FOV,
    double direction,
    double viewDistance ) [inline]
```

Jatekos konstruktora, ami a fontosabb dolgokat beallitja konfiguracios fajl nelkul (de pl fegyvert nem)

Parameters

<i>x</i>	- jatekos x koordinataja
<i>y</i>	- jatekos y koordinataja
<i>direction</i>	- jatekos iranya radianban (0 radian az x tengelyen valo pozitiv iranyu mozgassal egyenerteku)
<i>FOV</i>	- latoszog radianban
<i>viewDistance</i>	- latotavolsag (blokkok szamaban)

5.17.3 Member Function Documentation

5.17.3.1 getPosX()

```
double DoomCopy::Player::getPosX ( ) [inline]
```

Visszaadja a jatekos x koordinatajat

Returns

- jatekos x koordinataja

5.17.3.2 getPosY()

```
double DoomCopy::Player::getPosY ( ) [inline]
```

Visszaadja a jatekos y koordinatajat

Returns

- jatekos y koordinataja

5.17.3.3 move()

```
void DoomCopy::Player::move (
    double relativeX,
    double relativeY,
    const DoomCopy::Map & map )
```

Mozgas fuggveny, a jatekos a jelenlegi poziciojahoz kepest elmozdul relativeX-el es relativY-al, felteve, hogy ezzel a mozgassal nem utkozik blokkba (ezert van szukseg a mapre)

Parameters

<i>relativeX</i>	- mozgás x irányban a jelenlegi pozícióhoz kepest
<i>relativeY</i>	- mozgás y irányban a jelenlegi pozícióhoz kepest
<i>map</i>	- pályá

5.17.3.4 setPosX() [1/2]

```
void DoomCopy::Player::setPosX (
    double nPosX ) [inline]
```

Barmilyen vizsgalat nelkul atallitja a jatekos x poziciojat egy ujra.

Parameters

<i>nPosX</i>	- új x koordinataja a jatekosnak
--------------	----------------------------------

5.17.3.5 setPosX() [2/2]

```
void DoomCopy::Player::setPosX (
    double nPosX,
    const DoomCopy::Map & map )
```

Atallítja a játékos x koordinátáját egy újra, felteve, hogy így nem utkozik a játékos, vagy nem kerül blokk belsejébe.

Parameters

<i>nPosX</i>	- uj x koordinata
<i>map</i>	- palya

5.17.3.6 setPosY() [1/2]

```
void DoomCopy::Player::setPosY (
    double nPosY ) [inline]
```

Barmilyen vizsgalat nelkul atallitja a jatekos y poziciojat egy ujra.

Parameters

<i>nPosY</i>	- uj y koordinataja a jatekosnak
--------------	----------------------------------

5.17.3.7 setPosY() [2/2]

```
void DoomCopy::Player::setPosY (
    double nPosY,
    const DoomCopy::Map & map )
```

Atallitja a jatekos y koordinatajat egy ujra, felteve, hogy igy nem utkozik a jatekos, vagy nem kerul blokk belsejebe.

Parameters

<i>nPosY</i>	- uj y koordinata
<i>map</i>	- palya

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Player.h
- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Player.cpp

5.18 DoomCopy::Point Class Reference**Public Member Functions**

- **Point** (double x, double y)

Public Attributes

- double **x**
- double **y**

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Point.h

5.19 DoomCopy::Projectile Class Reference

Lovedek osztaly, rendelkezik lovedek tipussal, kezdopozicioval, iranyvektorral, es jelenlegi pozicioval.

```
#include <Projectile.h>
```

Public Member Functions

- **Projectile** (const **ProjectileType** &type, **Point** srcPosition, **Point** dir)
- bool **getStillExists** ()
- void **update** (**Map** &map, **Player** &player)

Public Attributes

- bool **visible** = false
- sf::Clock **clock**
- **ProjectileType** type
- **Point** direction
- **Point** currenPosition
- sf::VertexArray **vertexArray**

5.19.1 Detailed Description

Lovedek osztaly, rendelkezik lovedek tipussal, kezdopozicioval, iranyvektorral, es jelenlegi pozicioval.

5.19.2 Constructor & Destructor Documentation

5.19.2.1 Projectile()

```
DoomCopy::Projectile::Projectile (
    const ProjectileType & type,
    Point srcPosition,
    Point dir ) [inline]
```

Lovedek konstruktora - beallitja a lovedek tipusat, kezdopoziciojat, es iranyat.

Parameters

<i>srcPosition</i>	- kezdó pozíció
<i>dir</i>	- irányvektor
<i>type</i>	- lovedek típus

5.19.3 Member Function Documentation

5.19.3.1 getStillExists()

```
bool DoomCopy::Projectile::getStillExists ( ) [inline]
```

Azt vizsgálja, hogy a lovedek letezik-e.

Returns

- lovedek létezése

5.19.3.2 update()

```
void DoomCopy::Projectile::update (
    Map & map,
    Player & player )
```

Frissíti a lovedeket, azaz annak a pozícióját, és azt, hogy letezik-e a lovedek. Azért van szükség a map-re és a játékosra, hogy megvizsgálja, hogy eltalált-e bármit a lovedek, legyen az szörny, játékos, vagy akár csak a pályára egy fala.

Parameters

<i>map</i>	- pályára
<i>player</i>	- játékos

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Projectile.h
- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Projectile.cpp

5.20 DoomCopy::ProjectileType Class Reference

```
#include <Projectile.h>
```

Public Member Functions

- void **loadProjectileType** (std::string path, std::string line)

Public Attributes

- double **dmg**
- double **speed**
- double **range**
- **Texture text**

5.20.1 Detailed Description

Lovedektípus - a lovedekék gyakorlatilag ennek a példányai pozícióval, és különbozo muveletekkel/tulajdonsagokkal. A lovedektípusok rendelkeznek texturával, távolsággal (ameddig maximum elbirnak jutni), és sebeséggel.

5.20.2 Member Function Documentation

5.20.2.1 loadProjectileType()

```
void DoomCopy::ProjectileType::loadProjectileType (
    std::string path,
    std::string line )
```

Egy sorban található adatok alapján (line) betölti a lovedektipust és annak a texturáját az eleresi utvonal alapján.

Parameters

<i>line</i>	- string, amiben a lovedektipus adatai/tulajdonsagai szerepelnek
<i>path</i>	- a tipushoz tartozó textura elérési utvonala

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Projectile.h
- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Projectile.cpp

5.21 DoomCopy::Ray Class Reference

```
#include <Map.h>
```

Static Public Member Functions

- static double **distanceFromCollision** (const **Map** &map, **Point** source, **Point** direction, double stepSize, double maxDistance)

5.21.1 Detailed Description

A sugar, egy egyszeru funktor, ami ugy mukodik, hogy egy kezdopoziciobol egy adott irányba "kilovunk" egy sugarat, es megnezzuk, hogy blokkba utkozik-e, es visszaadjuk azt a tavolsagot amit ez a sugar megtett mig blokkba nem utkozott, vagy ha nem utkozott blokkba, akkor azt a maximalis tavolsagot, amit a sugar engedelyezetten meg tudott tenni.

5.21.2 Member Function Documentation

5.21.2.1 distanceFromCollision()

```
double DoomCopy::Ray::distanceFromCollision (
    const Map & map,
    Point source,
    Point direction,
    double stepSize,
    double maxDistance ) [inline], [static]
```

A sugar fuggvenye

Parameters

<i>source</i>	- a pozicio ahonnan a sugar indul
<i>direction</i>	- az irány amibe a sugar megy
<i>maxDistance</i>	- a maximum tavolsag amit a sugar megbir tenni
<i>stepSize</i>	- lepesemeret (mennyivel no a sugar azelott hogy megint megvizsgalnank, hogy blokkba utkozott-e)
<i>map</i>	- palya

Returns

- a tavolsag amit a sugar utkozes elott megtett.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Map.h

5.22 DoomCopy::ScreenSettings Class Reference

Kepernyo beallitasait tartalmazó almenu.

```
#include <Game.h>
```

Inherits **DoomCopy::Menu**.

Public Member Functions

- **ScreenSettings** (**Game** &game, **Menu** *back)

Public Attributes

- **Game** * game

5.22.1 Detailed Description

Kepernyo beallitasait tartalmaz almenu.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.h

5.23 DoomCopy::SetResHeight Class Reference

Menu pont ami atallitja a felbontas magassagat.

```
#include <Game.h>
```

Inherits **DoomCopy::Menu**.

Public Member Functions

- **SetResHeight** (**Game** &game, **Menu** *back)
- void **doAction** ()

Az a fuggvény ami a menu pont kivalasztasanal lefut. Ezt kell minden uj menunek felulirnia.

Public Attributes

- **Game** * game

5.23.1 Detailed Description

Menu pont ami atallitja a felbontas magassagat.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.h

5.24 DoomCopy::SetResWidth Class Reference

Menupont ami atallítja a felbontás szélességet.

```
#include <Game.h>
```

Inherits **DoomCopy::Menu**.

Public Member Functions

- **SetResWidth** (**Game** &game, **Menu** *back)
- void **doAction** ()

Az a függvény ami a menupont kiválasztásánál lefut. Ezt kell minden új menunek felulírnia.

Public Attributes

- **Game** * **game**

5.24.1 Detailed Description

Menupont ami atallítja a felbontás szélességet.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.h

5.25 DoomCopy::SetScreenHeight Class Reference

A menupont amelyik atallítja az ablak magasságát.

```
#include <Game.h>
```

Inherits **DoomCopy::Menu**.

Public Member Functions

- **SetScreenHeight** (**Game** &game, **Menu** *back)
- void **doAction** ()

Az a függvény ami a menupont kiválasztásánál lefut. Ezt kell minden új menunek felulírnia.

Public Attributes

- **Game** * **game**

5.25.1 Detailed Description

A menupont amelyik atallítja az ablak magasságát.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.h

5.26 DoomCopy::SetScreenWidth Class Reference

A menupont amelyik atallítja az ablak szélességét.

```
#include <Game.h>
```

Inherits **DoomCopy::Menu**.

Public Member Functions

- **SetScreenWidth** (**Game** &game, **Menu** *back)
- void **doAction** ()

Az a függvény ami a menupont kiválasztásánál lefut. Ezt kell minden új menunek felulírnia.

Public Attributes

- **Game** * **game**

5.26.1 Detailed Description

A menupont amelyik atallítja az ablak szélességét.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.h

5.27 DoomCopy::Settings Class Reference

Beállításokat tartalmazó menü.

```
#include <Game.h>
```

Inherits **DoomCopy::Menu**.

Public Member Functions

- **Settings** (**Game** &game, size_t arraySize, std::string name, **Menu** *back)

Public Attributes

- **Game * game**

5.27.1 Detailed Description

Beallitasokat tartalmazó menu.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.h

5.28 DoomCopy::Start Class Reference

```
#include <Game.h>
```

Inherits **DoomCopy::Menu**.

Public Member Functions

- **Start** (**Game** &game, size_t arraySize, std::string name, **Menu** *back=NULL)

Public Attributes

- **Game * game**

5.28.1 Detailed Description

Start (p. 48) menu, kiválasztása után felsorolja az összes map.conf-ban levo pályát. GStartGame-ekre mutat.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Game.h

5.29 DoomCopy::StringManager Class Reference

Static Public Member Functions

- static std::string **get_substring_btwn_first_occurences** (const std::string &source, const std::string &after_what, const std::string &until_what)
- static std::string **get_substring_btwn_last_occurences** (const std::string &source, const std::string &after_what, const std::string &until_what)
- static std::string **get_substring_after_first_occurence_of_and_last_occurence_of** (const std::string &source, const std::string &after_what, const std::string &until_what)
- static std::string **get_substring_after_last_occurence_of_and_first_occurence_of** (const std::string &source, const std::string &after_what, const std::string &until_what)
- static std::string **get_substring_btwn_first_and_next** (const std::string &source, const std::string &after_what, const std::string &until_what)
- static double **string_to_double** (std::string str)

5.29.1 Member Function Documentation

5.29.1.1 get_substring_after_first_occurence_of_and_last_occurence_of()

```
static std::string DoomCopy::StringManager::get_substring_after_first_occurence_of_and_last_↵  
occurence_of (   
    const std::string & source,   
    const std::string & after_what,   
    const std::string & until_what ) [inline], [static]
```

Egy adott stringbol visszaad egy olyan substringet, ami az after_what es az until_what stringek kozott talalhato a forras stringben. Ha tobbszor szerepel a forras stringben az after_what vagy az until_what, akkor az after_what-hoz hasonlo stringek kozul azt hasznalja, ami elsokent szerepel, mig az until_what-hoz hasonlo stringek kozul azt ami utoljara szerepel a forras stringben, es az ezek kozotti substringet adja vissza.

Parameters

<i>source</i>	- forras string - ebbol szeretnenk egy substringet
<i>after_what</i>	- milyen string utan kezdodik a substring
<i>until_what</i>	- milyen string elott fejezodik be a keresett substring

Returns

- a keresett substring

5.29.1.2 get_substring_after_last_occurence_of_and_first_occurence_of()

```
static std::string DoomCopy::StringManager::get_substring_after_last_occurence_of_and_first_↵  
occurence_of (   
    const std::string & source,   
    const std::string & after_what,   
    const std::string & until_what ) [inline], [static]
```

Egy adott stringbol visszaad egy olyan substringet, ami az after_what es az until_what stringek kozott talalhato a forras stringben. Ha tobbszor szerepel a forras stringben az after_what vagy az until_what, akkor az after_what-hoz hasonlo stringek kozul azt hasznalja, ami utolsokent szerepel, mig az until_what-hoz hasonlo stringek kozul azt ami elsokent szerepel a forras stringben, es az ezek kozotti substringet adja vissza.

Parameters

<i>source</i>	- forras string - ebbol szeretnenk egy substringet
<i>after_what</i>	- milyen string utan kezdodik a substring
<i>until_what</i>	- milyen string elott fejezodik be a keresett substring

Returns

- a keresett substring

5.29.1.3 get_substring_btwn_first_and_next()

```
static std::string DoomCopy::StringManager::get_substring_btwn_first_and_next (
    const std::string & source,
    const std::string & after_what,
    const std::string & until_what ) [inline], [static]
```

Egy adott stringbol visszaad egy substringet, ami az after_what es az until_what kozott talalhato. A visszaadott substring a forras stringben talalhato elso after_what utan kezdodik, es az elso after_what utani until_what utan er veget. Pelda: forras string: "test string", after_what: "test ", until_what: "t" Visszaadott ertekek: "s", mert habar a forrasban addig mar ketszer szerepelt a "t", minket az erdekel ami az after_what utan eloszor szerepel.

Parameters

<i>source</i>	- forras string - ebbol szeretnenk egy substringet
<i>after_what</i>	- milyen string utan kezdodik a substring
<i>until_what</i>	- milyen string elott fejezodik be a keresett substring

Returns

- a keresett substring

5.29.1.4 get_substring_btwn_first_occurences()

```
static std::string DoomCopy::StringManager::get_substring_btwn_first_occurences (
    const std::string & source,
    const std::string & after_what,
    const std::string & until_what ) [inline], [static]
```

Egy adott stringbol visszaad egy substringet, ami az after_what es az until_what kozott talalhato. Ha tobbszor szerepel a forras stringben az after_what vagy az until_what, akkor a tobbszor szereplo stringek kozul az(oka)t hasznalja, ami elsokent talalhato a forras stringben.

Parameters

<i>source</i>	- forras string - ebbol szeretnenk egy substringet
<i>after_what</i>	- milyen string utan kezdodik a substring
<i>until_what</i>	- milyen string elott fejezodik be a keresett substring

Returns

- a keresett substring/ures string ha nem talalhato a keresett substring

5.29.1.5 get_substring_btwn_last_occurences()

```
static std::string DoomCopy::StringManager::get_substring_btwn_last_occurences (
    const std::string & source,
    const std::string & after_what,
    const std::string & until_what ) [inline], [static]
```

Egy adott stringbol visszaad egy substringet, ami az after_what es az until_what kozott talalhato. Ha tobbszor szerepel a forras stringben az after_what vagy az until_what, akkor a tobbszor szereplo stringek kozul az(oka)t használja, ami utolsonak talalhato a forras stringben.

Parameters

<i>source</i>	- forras string - ebbol szeretnenk egy substringet
<i>after_what</i>	- milyen string utan kezdodik a substring
<i>until_what</i>	- milyen string elott fejezodik be a keresett substring

Returns

- a keresett substring/ures string ha nem talalhato a keresett substring

5.29.1.6 string_to_double()

```
static double DoomCopy::StringManager::string_to_double (
    std::string str ) [inline], [static]
```

Stringet valos szamma konvertal

Parameters

<i>str</i>	- string amibol szam lesz
------------	---------------------------

Returns

- a valos szam

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/StringManager.h

5.30 DoomCopy::Texture Class Reference

```
#include <Texture.h>
```

Public Member Functions

- **Texture** (int width, int height)
- void **addImg** (const char *pathToImg, const char *imgName)
- **Point** **sampleFrom** (std::string name)
- **Point** **getTextureSize** ()

Public Attributes

- std::string **names** [maxSize]
- sf::RenderStates **states**

5.30.1 Detailed Description

Textura Mivel az sf::VertexArray-nek csupan egy renderstate alapján van texturaja, így minden texturat ami egy VertexArray-hez tartozik érdemes egy nagy texturába összegyujteni, és így bizonyos tulajdonságok alapján (pl textura neve) lehet a VertexArray különböző texturáju részeihez különböző texturákat "kiválasztani". Mindez úgy működik, hogy a saját típusu textura class-nek van egy max merete (hány x,y meretu texturat bír eltarolni), egy jelenlegi merete (valójában hány textura van eddig bele rakva), és természetesen egy sf::Texture-je, aminek szélessége a maximum meret * x (ha x az egyeni texturak szélessége), és magassága: maximum meret * y. A texturak neve el van mentve, azzal együtt hogy a "textura matrixban" hol van a helyük, így egy sample függvénnyel a nev alapján visszakaphatjuk, hol a saját textura class sf::Texture-ben melyik x,y koordinataktól található a keresett texturánk.

5.30.2 Constructor & Destructor Documentation

5.30.2.1 Texture()

```
DoomCopy::Texture::Texture (
    int width,
    int height ) [inline]
```

Konstruktor, beállítja az egysegnyi texturak szélességeit, és magasságait, majd ez alapján létrehozza a textura megfelelő meretu sf::Texture-jét. Ebbe 9 darab a beállított szélességgel és magassággal rendelkező texturat/kepet lehet betölteni.

Parameters

<i>width</i>	- egyes texturak/kepek szélessége
<i>height</i>	- egyes texturak/kepek magassága

5.30.3 Member Function Documentation

5.30.3.1 addImg()

```
void DoomCopy::Texture::addImg (
    const char * pathToImg,
    const char * imgName ) [inline]
```

Betölt egy képet a "textura matrixba" a kép eleresi utvonala alapján

Parameters

<i>pathToImg</i>	- a kép eleresi utvonala
<i>imgName</i>	- a kép neve, ami alapján később meglehet találni.

5.30.3.2 getTextureSize()

```
Point DoomCopy::Texture::getTextureSize ( ) [inline]
```

Visszaadja a konstruktor által beállított egységnyi textúra szélességet, és magasságot.

Returns

- egységnyi textúra dimenziói

5.30.3.3 sampleFrom()

```
Point DoomCopy::Texture::sampleFrom (
    std::string name ) [inline]
```

Egy textúra neve alapján visszaad egy (x,y) koordinátát. A keresett textúrának a nagy "textura-matrixban" ez a bal felső pontja, és a textúra ettől tart az (x + textúra szélesség, y + textúra magassága) koordinátaig.

Parameters

<i>name</i>	- keresett textúrának a neve
-------------	------------------------------

Returns

- keresett textúra bal felső koordinátája a "textura-matrixban"

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Texture.h

5.31 DoomCopy::Weapon Class Reference

```
#include <Weapon.h>
```

Public Types

- enum **state** { **default_state**, **shoot** }

Public Member Functions

- **Weapon** ()
- **Weapon** (const **Point** *ownerPos, const double *ownerDirection)
- void **set** (const **Point** *ownerPos, const double *ownerDirection)
- bool **loadWeapon** (std::string path)
- void **gShot** (**Map** &map)
- void **updateState** ()

Public Attributes

- state **currentState** = default_state
- **ProjectileType** type
- **Texture** texture
- double **fireRate**
- double **range**
- const **Point** * **ownerPos**
- const double * **ownerDir**
- sf::VertexArray **vertexArray**

5.31.1 Detailed Description

Fegyver osztaly - rendelkezzeik texturaval, ket allapottal: normal, lott; a "tulajdonosa" poziciojaval es iranyvektoraval. Erre azert van szukseg, mert lovedeket loni a fegyver tud, nem a jatekos, igy a loveskor a fegyvernek "ismernie" kell a jatekos poziciojat es iranyat, hogy a lovedeket abba az iranyba inditsa el.

5.31.2 Constructor & Destructor Documentation

5.31.2.1 Weapon() [1/2]

```
DoomCopy::Weapon::Weapon ( ) [inline]
```

Letrehozza a fegyvert, de meg valojaban ez egy "ures" fegyver, minden tulajdonsage "ures".

5.31.2.2 Weapon() [2/2]

```
DoomCopy::Weapon::Weapon (
    const Point * ownerPos,
    const double * ownerDirection ) [inline]
```

Letrehoz egy fegyvert, aminek mar be van allitva a "pozicioja" es "iranya"

Parameters

<i>ownerPos</i>	- a fegyver tulajdonosának pozíciójára pointer, hogy mindig meg legyen a jelenlegi pozíció
<i>ownerDirection</i>	- irányvektorra pointer, hogy folyamatosan friss legyen az is.

5.31.3 Member Function Documentation

5.31.3.1 gShot()

```
void DoomCopy::Weapon::gShot (
    Map & map )
```

"Kilo" egy lovedeket. Azaz a pálya lovedeklistájához hozzáad egy lovedeket, aminek kezdopozíciója a fegyver "pozíciója" és iránya a fegyver "iránya".

Parameters

<i>map</i>	- pálya
------------	---------

5.31.3.2 loadWeapon()

```
bool DoomCopy::Weapon::loadWeapon (
    std::string path )
```

Betölti a fegyvert egy konfigurációs fájl alapján ami weapon.conf néven található az elérési útvonalon. Ez a fájl minden információt tartalmaz a fegyverről, mint lovedeknek típusa, tüzelési sebesség, textúra elérési útvonala, stb

Parameters

<i>path</i>	- elérési útvonal a weapon.conf-hoz
-------------	-------------------------------------

Returns

- betöltés sikeresége

5.31.3.3 set()

```
void DoomCopy::Weapon::set (
    const Point * ownerPos,
    const double * ownerDirection ) [inline]
```

Beállítja egy fegyver irányát és pozícióját

Parameters

<i>ownerPos</i>	- tulajdonos poziciojara pointer
<i>ownerDirection</i>	- tulajdonos irányara pointer

5.31.3.4 updateState()

```
void DoomCopy::Weapon::updateState ( ) [inline]
```

Frissíti a fegyver állapotát, a gShot fuggvény a fegyver állapotát shoot-ra állítja, és ez a fuggvény ezt frissíti.

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

- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Weapon.h
- /home/coldus/CLionProjects/CPP_Nagyhazi/DoomCopy/Weapon.cpp

Index

- ~Array2D
 - DoomCopy::Array2D< T >, 10
- addImg
 - DoomCopy::Texture, 52
- addItem
 - DoomCopy::List< T >, 25
 - DoomCopy::Menu, 35
- addrAt
 - DoomCopy::List< T >, 26
- Array2D
 - DoomCopy::Array2D< T >, 9
- at
 - DoomCopy::List< T >, 26
- cmpr
 - DoomCopy::cmpr, 12
- command
 - DoomCopy::Game, 20
- Creature
 - DoomCopy::Creature, 14
- damage
 - DoomCopy::Creature, 15
- deleteAt
 - DoomCopy::List< T >, 26, 27
- deleteDeadOrNonExistent
 - DoomCopy::Game, 20
- distanceFromCollision
 - DoomCopy::Ray, 44
- dmgPlayer
 - DoomCopy::Creature, 15
- doesItContain
 - DoomCopy::List< T >, 27
- DoomCopy, 7
 - numpad, 8
- DoomCopy::Array2D< T >, 9
 - ~Array2D, 10
 - Array2D, 9
- DoomCopy::Blocks, 10
 - getTextureName, 10
 - isTypeSolid, 11
 - load, 11
- DoomCopy::BlockType, 11
- DoomCopy::cmpr, 12
 - cmpr, 12
 - nearby, 13
- DoomCopy::Creature, 13
 - Creature, 14
 - damage, 15
 - dmgPlayer, 15
 - getPos, 15
 - heal, 16
 - isPlayerVisble, 16
 - move, 16
 - setPos, 17
 - turn, 17
 - update, 17
- DoomCopy::Exit, 18
- DoomCopy::Game, 18
 - command, 20
 - deleteDeadOrNonExistent, 20
 - Game, 19
 - getResHeight, 20
 - getResWidth, 20
 - getScreenHeight, 21
 - getScreenWidth, 21
 - loadSettings, 21
 - renderEnemies, 21
 - renderProjectiles, 21
 - setResHeight, 21
 - setResWidth, 22
 - setScreenHeight, 22
 - setScreenWidth, 22
 - startCLIGame, 22
 - startGraphicalGame, 23
 - winCondition, 23
- DoomCopy::GStartGame, 24
- DoomCopy::HighScore, 24
- DoomCopy::List< T >, 25
 - addItem, 25
 - addrAt, 26
 - at, 26
 - deleteAt, 26, 27
 - doesItContain, 27
 - pSearchBy, 27
 - searchBy, 28
 - setDestructFunction, 28
 - setHead, 28
 - sort, 29
- DoomCopy::ListItem< T >, 29
- DoomCopy::MainMenu, 30
- DoomCopy::Map, 30
 - load, 31
 - loadEnemies, 31
 - Map, 31
- DoomCopy::MapHighScore, 32
- DoomCopy::MeleeEnemy, 33
 - MeleeEnemy, 33

- DoomCopy::Menu, 34
 - addItem, 35
 - getItemAt, 35
 - Menu, 34
- DoomCopy::Player, 35
 - getPosX, 37
 - getPosY, 37
 - move, 38
 - Player, 36, 37
 - setPosX, 38
 - setPosY, 40
- DoomCopy::Point, 40
- DoomCopy::Projectile, 41
 - getStillExists, 42
 - Projectile, 41
 - update, 42
- DoomCopy::ProjectileType, 42
 - loadProjectileType, 43
- DoomCopy::Ray, 43
 - distanceFromCollision, 44
- DoomCopy::ScreenSettings, 44
- DoomCopy::SetResHeight, 45
- DoomCopy::SetResWidth, 46
- DoomCopy::SetScreenHeight, 46
- DoomCopy::SetScreenWidth, 47
- DoomCopy::Settings, 47
- DoomCopy::Start, 48
- DoomCopy::StringManager, 48
 - get_substring_after_first_occurrence_of_and_last_occurrence_of, 49
 - get_substring_after_last_occurrence_of_and_first_occurrence_of, 49
 - get_substring_btwn_first_and_next, 50
 - get_substring_btwn_first_occurrences, 50
 - get_substring_btwn_last_occurrences, 51
 - string_to_double, 51
- DoomCopy::Texture, 51
 - addImg, 52
 - getTextureSize, 53
 - sampleFrom, 53
 - Texture, 52
- DoomCopy::Weapon, 54
 - gShot, 55
 - loadWeapon, 55
 - set, 55
 - updateState, 56
 - Weapon, 54
- Game
 - DoomCopy::Game, 19
- get_substring_after_first_occurrence_of_and_last_occurrence_of, 49
 - DoomCopy::StringManager, 49
- get_substring_after_last_occurrence_of_and_first_occurrence_of, 49
 - DoomCopy::StringManager, 49
- get_substring_btwn_first_and_next
 - DoomCopy::StringManager, 50
- get_substring_btwn_first_occurrences
 - DoomCopy::StringManager, 50
- get_substring_btwn_last_occurrences
 - DoomCopy::StringManager, 51
- getItemAt
 - DoomCopy::Menu, 35
- getPos
 - DoomCopy::Creature, 15
- getPosX
 - DoomCopy::Player, 37
- getPosY
 - DoomCopy::Player, 37
- getResHeight
 - DoomCopy::Game, 20
- getResWidth
 - DoomCopy::Game, 20
- getScreenHeight
 - DoomCopy::Game, 21
- getScreenWidth
 - DoomCopy::Game, 21
- getStillExists
 - DoomCopy::Projectile, 42
- getTextureName
 - DoomCopy::Blocks, 10
- getTextureSize
 - DoomCopy::Texture, 53
- gShot
 - DoomCopy::Weapon, 55
- heal
 - DoomCopy::Creature, 16
- isPlayerVisble
 - DoomCopy::Creature, 16
- isTypeSolid
 - DoomCopy::Blocks, 11
- load
 - DoomCopy::Blocks, 11
 - DoomCopy::Map, 31
- loadEnemies
 - DoomCopy::Map, 31
- loadProjectileType
 - DoomCopy::ProjectileType, 43
- loadSettings
 - DoomCopy::Game, 21
- loadWeapon
 - DoomCopy::Weapon, 55
- Map
 - DoomCopy::Map, 31
- MeleeEnemy
 - DoomCopy::MeleeEnemy, 33
- Menu
 - DoomCopy::Menu, 34
- move
 - DoomCopy::Creature, 16
 - DoomCopy::Player, 38
- nearby
 - DoomCopy::cmpr, 13
- numpad

- DoomCopy, 8
- Player
 - DoomCopy::Player, 36, 37
- Projectile
 - DoomCopy::Projectile, 41
- pSearchBy
 - DoomCopy::List< T >, 27
- renderEnemies
 - DoomCopy::Game, 21
- renderProjectiles
 - DoomCopy::Game, 21
- sampleFrom
 - DoomCopy::Texture, 53
- searchBy
 - DoomCopy::List< T >, 28
- set
 - DoomCopy::Weapon, 55
- setDestructFunction
 - DoomCopy::List< T >, 28
- setHead
 - DoomCopy::List< T >, 28
- setPos
 - DoomCopy::Creature, 17
- setPosX
 - DoomCopy::Player, 38
- setPosY
 - DoomCopy::Player, 40
- setResHeight
 - DoomCopy::Game, 21
- setResWidth
 - DoomCopy::Game, 22
- setScreenHeight
 - DoomCopy::Game, 22
- setScreenWidth
 - DoomCopy::Game, 22
- sort
 - DoomCopy::List< T >, 29
- startCLIGame
 - DoomCopy::Game, 22
- startGraphicalGame
 - DoomCopy::Game, 23
- string_to_double
 - DoomCopy::StringManager, 51
- Texture
 - DoomCopy::Texture, 52
- turn
 - DoomCopy::Creature, 17
- update
 - DoomCopy::Creature, 17
 - DoomCopy::Projectile, 42
- updateState
 - DoomCopy::Weapon, 56
- Weapon
 - DoomCopy::Weapon, 54
- winCondition
 - DoomCopy::Game, 23