

PRAKTIKUM IZ PROGRAMIRANJA 2

ODSEK ZA SOFTVERSKO INŽENJERSTVO

Projektni zadatak broj 16

„1945: The Final Retribution“

Projektna dokumentacija

Projektni autori	Lazar Mitrović	15/0650	(lazanet96@gmail.com)
	Kosta Stojiljković	15/0095	(kostastojiljkovic@gmail.com)
	Nikola Jovanović	15/0081	(jovanovicn.96@gmail.com)



Beograd, jun 2016.



1945: Final Retribution

Projektna dokumentacija, tim IMEEE 802.11
PP2S 20.6.2016

Sadržaj

1	Spisak struktura/unija	1
1.1	Spisak struktura	1
2	Indeks datoteka	3
2.1	Spisak datoteka	3
3	Dokumentacija stuktura/unija	5
3.1	Dokumentacija strukture bulletListNodeStruct	5
3.1.1	Opširniji opis	5
3.1.2	Dokumentacija polja	5
3.1.2.1	bullet	5
3.1.2.2	next	5
3.1.2.3	prev	5
3.2	Dokumentacija strukture bulletStruct	6
3.2.1	Opširniji opis	6
3.2.2	Dokumentacija polja	6
3.2.2.1	counter	6
3.2.2.2	handleIteration	6
3.2.2.3	id	6
3.2.2.4	isPlayerOwned	6
3.2.2.5	position	6
3.2.2.6	texturePosition	7
3.2.2.7	x_pos	7
3.2.2.8	y_pos	7
3.3	Dokumentacija strukture effectListNodeStruct	7
3.3.1	Opširniji opis	7
3.3.2	Dokumentacija polja	7
3.3.2.1	effect	7
3.3.2.2	next	7
3.3.2.3	prev	7
3.4	Dokumentacija strukture effectStruct	8
3.4.1	Opširniji opis	8

3.4.2	Dokumentacija polja	8
3.4.2.1	counter	8
3.4.2.2	handleIteration	8
3.4.2.3	id	8
3.4.2.4	position	8
3.4.2.5	x_pos	8
3.4.2.6	y_pos	9
3.5	Dokumentacija strukture enemyListNodeStruct	9
3.5.1	Opširniji opis	9
3.5.2	Dokumentacija polja	9
3.5.2.1	enemy	9
3.5.2.2	next	9
3.5.2.3	prev	9
3.6	Dokumentacija strukture enemyStruct	9
3.6.1	Opširniji opis	10
3.6.2	Dokumentacija polja	10
3.6.2.1	coolDown	10
3.6.2.2	counter	10
3.6.2.3	handleIteration	10
3.6.2.4	health	10
3.6.2.5	id	11
3.6.2.6	path_id	11
3.6.2.7	position	11
3.6.2.8	score	11
3.6.2.9	texture_slot	11
3.6.2.10	texturePosition	11
3.6.2.11	x_pos	11
3.6.2.12	y_pos	11
3.7	Dokumentacija strukture gameStruct	11
3.7.1	Opširniji opis	13
3.7.2	Dokumentacija polja	13
3.7.2.1	background_offset	13
3.7.2.2	backgroundTiles	13
3.7.2.3	bomb_texture	13
3.7.2.4	boss_active	13
3.7.2.5	bullet_texture	14
3.7.2.6	bullets	14
3.7.2.7	bulletsCount	14
3.7.2.8	campaign_level_ind	14
3.7.2.9	campaign_row	14

3.7.2.10	difficulty	14
3.7.2.11	effects	14
3.7.2.12	effectsCount	14
3.7.2.13	enemies	14
3.7.2.14	enemiesCount	14
3.7.2.15	enemy_sprites	14
3.7.2.16	enemySpawnCounter	14
3.7.2.17	explosion_texture	15
3.7.2.18	final_boss_active	15
3.7.2.19	frames	15
3.7.2.20	framesFromCont	15
3.7.2.21	fumes_texture	15
3.7.2.22	game_mode	15
3.7.2.23	handleIteration	15
3.7.2.24	holeTexture	15
3.7.2.25	HSFileName	15
3.7.2.26	HSNames	15
3.7.2.27	HSScores	15
3.7.2.28	level_default	15
3.7.2.29	level_file	16
3.7.2.30	level_matrix	16
3.7.2.31	pauseSurface	16
3.7.2.32	pauseTexture	16
3.7.2.33	players	16
3.7.2.34	playersCount	16
3.7.2.35	playerTexture	16
3.7.2.36	playerTextureLeft	16
3.7.2.37	playerTextureRight	16
3.7.2.38	powerUps	16
3.7.2.39	powerUpTexture	16
3.7.2.40	score	16
3.7.2.41	shieldTexture	17
3.7.2.42	speedTexture	17
3.8	Dokumentacija strukture HSFileMemberStruct	17
3.8.1	Opširniji opis	17
3.8.2	Dokumentacija polja	17
3.8.2.1	name	17
3.8.2.2	score	17
3.9	Dokumentacija strukture maxLevelSaveStruct	17
3.9.1	Opširniji opis	18

3.9.2	Dokumentacija polja	18
3.9.2.1	level	18
3.10	Dokumentacija strukture playerListNodeStruct	18
3.10.1	Opširniji opis	18
3.10.2	Dokumentacija polja	18
3.10.2.1	next	18
3.10.2.2	player	18
3.10.2.3	prev	18
3.11	Dokumentacija strukture playerStruct	19
3.11.1	Opširniji opis	19
3.11.2	Dokumentacija polja	19
3.11.2.1	bombCoolDown	19
3.11.2.2	bulletCoolDown	19
3.11.2.3	counter	19
3.11.2.4	handleIteration	20
3.11.2.5	health	20
3.11.2.6	movement	20
3.11.2.7	position	20
3.11.2.8	shieldActive	20
3.11.2.9	speedActive	20
3.11.2.10	texturePosition	20
3.11.2.11	x_pos	20
3.11.2.12	y_pos	20
3.12	Dokumentacija strukture powerUpListNodeStruct	20
3.12.1	Opširniji opis	21
3.12.2	Dokumentacija polja	21
3.12.2.1	next	21
3.12.2.2	powerUp	21
3.12.2.3	prev	21
3.13	Dokumentacija strukture powerUpStruct	21
3.13.1	Opširniji opis	21
3.13.2	Dokumentacija polja	22
3.13.2.1	counter	22
3.13.2.2	handleIteration	22
3.13.2.3	id	22
3.13.2.4	position	22
3.13.2.5	x_pos	22
3.13.2.6	y_pos	22
3.14	Dokumentacija strukture saveHeaderStruct	22
3.14.1	Dokumentacija polja	22

3.14.1.1	bulletNum	22
3.14.1.2	effectNum	22
3.14.1.3	enemyNum	22
3.14.1.4	powerUpNum	23
3.15	Dokumentacija strukture textStruct	23
3.15.1	Opširniji opis	23
3.15.2	Dokumentacija polja	23
3.15.2.1	color	23
3.15.2.2	font	23
3.15.2.3	rect	23
3.15.2.4	show	23
3.15.2.5	str	23
3.15.2.6	texture	24
4	Dokumentacija datoteke	25
4.1	Opis datoteke defines.h	25
4.2	Opis datoteke draw.c	25
4.2.1	Dokumentacija funkcije	25
4.2.1.1	drawBackground	25
4.2.1.2	drawBullets	26
4.2.1.3	drawEffects	26
4.2.1.4	drawEnemies	26
4.2.1.5	drawPlayers	26
4.2.1.6	drawPowerUps	26
4.2.1.7	drawText	26
4.3	Opis datoteke draw.h	26
4.3.1	Dokumentacija funkcije	27
4.3.1.1	drawBackground	27
4.3.1.2	drawBullets	27
4.3.1.3	drawEffects	27
4.3.1.4	drawEnemies	27
4.3.1.5	drawPlayers	27
4.3.1.6	drawPowerUps	27
4.3.1.7	drawText	27
4.3.2	Dokumentacija promenljive	27
4.3.2.1	backgroundRect	27
4.3.2.2	bgSound	28
4.3.2.3	bottomStripRect	28
4.3.2.4	explosionSound	28
4.3.2.5	fire	28

4.3.2.6	font	28
4.3.2.7	game	28
4.3.2.8	lastCampaignLevel	28
4.3.2.9	renderer	28
4.3.2.10	topStripRect	28
4.3.2.11	window	28
4.4	Opis datoteke game/campaign.c	28
4.4.1	Dokumentacija promenljive	29
4.4.1.1	campaign_background	29
4.5	Opis datoteke game/campaign.h	29
4.5.1	Dokumentacija promenljive	29
4.5.1.1	campaign_background	29
4.6	Opis datoteke game/collisions.c	29
4.6.1	Dokumentacija makro zamene	29
4.6.1.1	MIN	29
4.6.2	Dokumentacija funkcije	29
4.6.2.1	detectBulletEnemyCollision	29
4.6.2.2	detectBulletPlayerCollision	30
4.6.2.3	detectCollisions	30
4.6.2.4	detectPlayerEnemyCollision	30
4.6.2.5	detectPowerUpEnemyCollision	30
4.6.2.6	detectPowerUpPlayerCollision	30
4.7	Opis datoteke game/collisions.h	30
4.7.1	Dokumentacija funkcije	31
4.7.1.1	detectBulletEnemyCollision	31
4.7.1.2	detectBulletPlayerCollision	31
4.7.1.3	detectCollisions	31
4.7.1.4	detectPlayerEnemyCollision	31
4.7.1.5	detectPowerUpEnemyCollision	31
4.7.1.6	detectPowerUpPlayerCollision	31
4.7.2	Dokumentacija promenljive	32
4.7.2.1	backgroundRect	32
4.7.2.2	bgSound	32
4.7.2.3	bottomStripRect	32
4.7.2.4	explosionSound	32
4.7.2.5	fire	32
4.7.2.6	font	32
4.7.2.7	game	32
4.7.2.8	lastCampaignLevel	32
4.7.2.9	renderer	32

4.7.2.10	topStripRect	32
4.7.2.11	window	32
4.8	Opis datoteke game/game.c	33
4.8.1	Dokumentacija funkcije	33
4.8.1.1	background_campaign	33
4.8.1.2	background_load_row	33
4.8.1.3	background_random	33
4.8.1.4	background_swap	33
4.8.1.5	doBackgroundJob	33
4.8.1.6	renderGame	33
4.8.1.7	runGame	34
4.8.1.8	swap_tile_rows	34
4.8.1.9	updateEffects	34
4.9	Opis datoteke game/game.h	34
4.9.1	Dokumentacija makro zamene	35
4.9.1.1	PLAYERS_NUM	35
4.9.1.2	TEXTS_NUM	35
4.9.2	Dokumentacija definicije tipa	35
4.9.2.1	HSFileMemberPtr	35
4.9.2.2	maxLevelSavePtr	36
4.9.3	Dokumentacija funkcije	36
4.9.3.1	renderGame	36
4.9.3.2	runGame	36
4.9.4	Dokumentacija promenljive	36
4.9.4.1	backgroundRect	36
4.9.4.2	backgroundTexture	36
4.9.4.3	bgSound	36
4.9.4.4	bottomStripRect	36
4.9.4.5	events	36
4.9.4.6	explosionSound	36
4.9.4.7	fire	37
4.9.4.8	font	37
4.9.4.9	game	37
4.9.4.10	gameOverRect	37
4.9.4.11	gameOverTexture	37
4.9.4.12	HSBackgroundRect	37
4.9.4.13	HSBackgroundTexture	37
4.9.4.14	lastCampaignLevel	37
4.9.4.15	levelClearTexture	37
4.9.4.16	players	37

4.9.4.17	renderer	37
4.9.4.18	texts	37
4.9.4.19	topStripRect	37
4.9.4.20	window	38
4.10	Opis datoteke game/handlers.h	38
4.10.1	Dokumentacija funkcije	39
4.10.1.1	bulletIterationHandlerBomb	39
4.10.1.2	bulletIterationHandlerBullet	39
4.10.1.3	effectIterationHandlerAirExplosion	39
4.10.1.4	effectIterationHandlerExhaust	39
4.10.1.5	effectIterationHandlerFallingPlane	40
4.10.1.6	effectIterationHandlerHole	41
4.10.1.7	effectIterationHandlerLandExplosion	41
4.10.1.8	effectIterationHandlerPowerUp	41
4.10.1.9	enemyIterationHandlerBoss	41
4.10.1.10	enemyIterationHandlerDefault	41
4.10.1.11	enemyIterationHandlerFollower	42
4.10.1.12	enemyIterationHandlerGroundTarget	42
4.10.1.13	enemyIterationHandlerKamikaze	42
4.10.1.14	gameIterationHandlerDefault	42
4.10.1.15	playerIterationHandlerDefault	42
4.10.1.16	playerIterationHandlerDemo	43
4.10.1.17	powerUpIterationHandler	43
4.10.2	Dokumentacija promenljive	43
4.10.2.1	backgroundRect	43
4.10.2.2	bgSound	43
4.10.2.3	bottomStripRect	43
4.10.2.4	explosionSound	43
4.10.2.5	fire	43
4.10.2.6	font	44
4.10.2.7	game	44
4.10.2.8	gameOverRect	44
4.10.2.9	HSBackgroundRect	44
4.10.2.10	lastCampaignLevel	44
4.10.2.11	renderer	44
4.10.2.12	topStripRect	44
4.10.2.13	window	44
4.11	Opis datoteke game/iterationHandlers.c	44
4.11.1	Dokumentacija funkcije	45
4.11.1.1	bulletIterationHandlerBomb	45

4.11.1.2	bulletIterationHandlerBullet	45
4.11.1.3	effectIterationHandlerAirExplosion	45
4.11.1.4	effectIterationHandlerExhaust	45
4.11.1.5	effectIterationHandlerFallingPlane	45
4.11.1.6	effectIterationHandlerHole	46
4.11.1.7	effectIterationHandlerLandExplosion	46
4.11.1.8	effectIterationHandlerPowerUp	46
4.11.1.9	enemyIterationHandlerBoss	46
4.11.1.10	enemyIterationHandlerDefault	46
4.11.1.11	enemyIterationHandlerFollower	47
4.11.1.12	enemyIterationHandlerGroundTarget	47
4.11.1.13	enemyIterationHandlerKamikaze	47
4.11.1.14	gameIterationHandlerDefault	47
4.11.1.15	playerIterationHandlerDefault	47
4.11.1.16	playerIterationHandlerDemo	48
4.11.1.17	powerUpIterationHandler	48
4.12	Opis datoteke game/savingGame.c	48
4.12.1	Dokumentacija funkcije	48
4.12.1.1	loadGame	48
4.12.1.2	saveGame	48
4.13	Opis datoteke game/savingGame.h	49
4.13.1	Dokumentacija definicije tipa	50
4.13.1.1	saveHeaderPtr	50
4.13.2	Dokumentacija funkcije	50
4.13.2.1	loadGame	50
4.13.2.2	saveGame	50
4.13.3	Dokumentacija promenljive	50
4.13.3.1	backgroundRect	50
4.13.3.2	bgSound	50
4.13.3.3	bottomStripRect	50
4.13.3.4	explosionSound	50
4.13.3.5	fire	50
4.13.3.6	font	50
4.13.3.7	game	50
4.13.3.8	lastCampaignLevel	51
4.13.3.9	renderer	51
4.13.3.10	topStripRect	51
4.13.3.11	window	51
4.14	Opis datoteke instances/constructors.c	51
4.14.1	Dokumentacija funkcije	51

4.14.1.1	editText	51
4.14.1.2	initBullet	51
4.14.1.3	initEffect	52
4.14.1.4	initEnemy	52
4.14.1.5	initGame	52
4.14.1.6	initLevelMatrix	52
4.14.1.7	initPlayer	53
4.14.1.8	initPowerUp	53
4.14.1.9	initText	53
4.14.1.10	initTextures	53
4.14.1.11	saveHighScore	53
4.15	Opis datoteke instances/destructors.c	53
4.15.1	Dokumentacija funkcije	54
4.15.1.1	freeBulletNode	54
4.15.1.2	freeEffectNode	54
4.15.1.3	freeEnemyNode	54
4.15.1.4	freeGame	54
4.15.1.5	freePlayerNode	54
4.15.1.6	freePowerUpNode	55
4.15.1.7	freeText	55
4.16	Opis datoteke instances/instances.h	55
4.16.1	Dokumentacija funkcije	56
4.16.1.1	editText	56
4.16.1.2	freeBulletNode	56
4.16.1.3	freeEffectNode	56
4.16.1.4	freeEnemyNode	57
4.16.1.5	freeGame	57
4.16.1.6	freePlayerNode	57
4.16.1.7	freePowerUpNode	57
4.16.1.8	freeText	57
4.16.1.9	initBullet	57
4.16.1.10	initEffect	58
4.16.1.11	initEnemy	58
4.16.1.12	initGame	58
4.16.1.13	initLevelMatrix	58
4.16.1.14	initLoad_curr_lvl	59
4.16.1.15	initPlayer	59
4.16.1.16	initPowerUp	59
4.16.1.17	initText	59
4.16.1.18	initTextures	59

4.16.1.19	saveHighScore	59
4.16.2	Dokumentacija promenljive	59
4.16.2.1	backgroundRect	59
4.16.2.2	bgSound	60
4.16.2.3	bottomStripRect	60
4.16.2.4	explosionSound	60
4.16.2.5	fire	60
4.16.2.6	font	60
4.16.2.7	game	60
4.16.2.8	lastCampaignLevel	60
4.16.2.9	renderer	60
4.16.2.10	topStripRect	60
4.16.2.11	window	60
4.17	Opis datoteke main.c	60
4.17.1	Dokumentacija funkcije	61
4.17.1.1	initProgram	61
4.17.1.2	killGame	61
4.17.1.3	main	61
4.18	Opis datoteke main.h	61
4.18.1	Dokumentacija promenljive	62
4.18.1.1	backgroundRect	62
4.18.1.2	bgSound	62
4.18.1.3	bottomStripRect	62
4.18.1.4	explosionSound	62
4.18.1.5	fire	62
4.18.1.6	font	62
4.18.1.7	game	62
4.18.1.8	lastCampaignLevel	62
4.18.1.9	renderer	62
4.18.1.10	topStripRect	62
4.18.1.11	window	63
4.19	Opis datoteke menu/menu.c	63
4.19.1	Dokumentacija funkcije	63
4.19.1.1	getLevelFileName	63
4.19.1.2	getSavedGameName	63
4.19.1.3	selectCampaignLevel	63
4.19.1.4	showMenu	64
4.20	Opis datoteke menu/menu.h	64
4.20.1	Dokumentacija makro zamene	65
4.20.1.1	TEXT_TEXTURE_NUM	65

4.20.2 Dokumentacija funkcije	65
4.20.2.1 showMenu	65
4.20.3 Dokumentacija promenljive	65
4.20.3.1 backgroundOverlay	65
4.20.3.2 backgroundRect	65
4.20.3.3 backgroundTexture	66
4.20.3.4 bgSound	66
4.20.3.5 bottomStripRect	66
4.20.3.6 difficulty	66
4.20.3.7 events	66
4.20.3.8 explosionSound	66
4.20.3.9 fire	66
4.20.3.10 font	66
4.20.3.11 game	66
4.20.3.12 helpRect	66
4.20.3.13 helpTexture	66
4.20.3.14 helpTipRect	66
4.20.3.15 lastCampaignLevel	66
4.20.3.16 logoRect	66
4.20.3.17 logoTexture	66
4.20.3.18 mainMetalFrameRect	67
4.20.3.19 menuMusic	67
4.20.3.20 metalFrameRect	67
4.20.3.21 metalFrameTexture	67
4.20.3.22 renderer	67
4.20.3.23 selector2Pos	67
4.20.3.24 selector2Rect	67
4.20.3.25 selectorPos	67
4.20.3.26 selectorRect	67
4.20.3.27 selectorTexture	67
4.20.3.28 textTextures	67
4.20.3.29 topStripRect	67
4.20.3.30 window	67
Sadržaj	69

Glava 1

Spisak struktura/unija

1.1 Spisak struktura

Spisak struktura sa kratkim opisima:

bulletListNodeStruct	5
bulletStruct	6
effectListNodeStruct	7
effectStruct	8
enemyListNodeStruct	9
enemyStruct	9
gameStruct	11
HSFileMemberStruct	17
maxLevelSaveStruct	17
playerListNodeStruct	18
playerStruct	19
powerUpListNodeStruct	20
powerUpStruct	21
saveHeaderStruct	22
textStruct	23

Glava 2

Indeks datoteka

2.1 Spisak datoteka

Spisak svih datoteka, sa kratkim opisima:

defines.h	25
draw.c	25
draw.h	26
main.c	60
main.h	61
game/campaign.c	28
game/campaign.h	29
game/collisions.c	29
game/collisions.h	30
game/game.c	33
game/game.h	34
game/handlers.h	38
game/iterationHandlers.c	44
game/savingGame.c	48
game/savingGame.h	49
instances/constructors.c	51
instances/destructors.c	53
instances/instances.h	55
menu/menu.c	63
menu/menu.h	64

Glava 3

Dokumentacija stuktura/unija

3.1 Dokumentacija strukture bulletListNodeStruct

```
#include <defines.h>
```

Klasni dijagram za bulletListNodeStruct:

Polja

- [bulletPtr bullet](#)
pokazivač na strukturu metka
- struct [bulletListNodeStruct](#) * [next](#)
sledeći element liste
- struct [bulletListNodeStruct](#) * [prev](#)
prethodni element liste

3.1.1 Opširniji opis

Struktura člana liste metaka

3.1.2 Dokumentacija polja

3.1.2.1 [bulletPtr bulletListNodeStruct::bullet](#)

pokazivač na strukturu metka

3.1.2.2 struct [bulletListNodeStruct](#)* [bulletListNodeStruct::next](#)

sledeći element liste

3.1.2.3 struct [bulletListNodeStruct](#)* [bulletListNodeStruct::prev](#)

prethodni element liste

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [defines.h](#)

3.2 Dokumentacija strukture bulletStruct

```
#include <defines.h>
```

Polja

- double `x_pos`
x koordinata
- double `y_pos`
y koordinata
- int `counter`
brojač po frejmovima
- int `id`
identifikacioni broj tipa metka (1:običan 2:bomba)
- short `isPlayerOwned`
flag da li je igrač vlasnik metka
- SDL_Rect `position`
pravougaonik iscrtavanja
- SDL_Rect `texturePosition`
pozicija teksture
- void(* `handleIteration`)(`bulletListNodePtr`)
pokazivač na odgovarajuću funkciju koja se poziva pri svakoj iteraciji

3.2.1 Opširniji opis

Struktura instance metka

3.2.2 Dokumentacija polja

3.2.2.1 int bulletStruct::counter

brojač po frejmovima

3.2.2.2 void(* bulletStruct::handleIteration)(bulletListNodePtr)

pokazivač na odgovarajuću funkciju koja se poziva pri svakoj iteraciji

3.2.2.3 int bulletStruct::id

identifikacioni broj tipa metka (1:običan 2:bomba)

3.2.2.4 short bulletStruct::isPlayerOwned

flag da li je igrač vlasnik metka

3.2.2.5 SDL_Rect bulletStruct::position

pravougaonik iscrtavanja

3.2.2.6 SDL_Rect bulletStruct::texturePosition

pozicija teksture

3.2.2.7 double bulletStruct::x_pos

x koordinata

3.2.2.8 double bulletStruct::y_pos

y koordinata

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [defines.h](#)

3.3 Dokumentacija strukture effectListNodeStruct

```
#include <defines.h>
```

Klasni dijagram za effectListNodeStruct:

Polja

- [effectPtr effect](#)
pokazivač na strukturu efekta
- struct [effectListNodeStruct](#) * [next](#)
sledeći element liste
- struct [effectListNodeStruct](#) * [prev](#)
prethodni element liste

3.3.1 Opširniji opis

Struktura člana liste efekata

3.3.2 Dokumentacija polja

3.3.2.1 effectPtr effectListNodeStruct::effect

pokazivač na strukturu efekta

3.3.2.2 struct effectListNodeStruct* effectListNodeStruct::next

sledeći element liste

3.3.2.3 struct effectListNodeStruct* effectListNodeStruct::prev

prethodni element liste

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [defines.h](#)

3.4 Dokumentacija strukture effectStruct

```
#include <defines.h>
```

Polja

- double `x_pos`
x koordinata
- double `y_pos`
y koordinata
- SDL_Rect `position`
pravougaonik iscrtavanja
- int `counter`
brojač frejmova od inicijalizacije
- int `id`
identifikacioni broj tipa efekta (1:dim 2:eksplozija u vazduhu, 3:eksplozija na zemlji, 4:rupa na zemlji, 5:pad običnog aviona, 6:pad štuke, 7:pad bossa followera, 8:pad kamikaza, 9:pad glavnog bossa, 10:štit, 11:brzina)
- int(* `handleIteration`)(`effectListNodePtr`)
pokazivač na odgovarajuću funkciju koja se poziva pri svakoj iteraciji

3.4.1 Opširniji opis

Struktura instance efekta

3.4.2 Dokumentacija polja

3.4.2.1 int effectStruct::counter

brojač frejmova od inicijalizacije

3.4.2.2 int(* effectStruct::handleIteration)(effectListNodePtr)

pokazivač na odgovarajuću funkciju koja se poziva pri svakoj iteraciji

3.4.2.3 int effectStruct::id

identifikacioni broj tipa efekta (1:dim 2:eksplozija u vazduhu, 3:eksplozija na zemlji, 4:rupa na zemlji, 5:pad običnog aviona, 6:pad štuke, 7:pad bossa followera, 8:pad kamikaza, 9:pad glavnog bossa, 10:štit, 11:brzina)

3.4.2.4 SDL_Rect effectStruct::position

pravougaonik iscrtavanja

3.4.2.5 double effectStruct::x_pos

x koordinata

3.4.2.6 double effectStruct::y_pos

y koordinata

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [defines.h](#)

3.5 Dokumentacija strukture enemyListNodeStruct

```
#include <defines.h>
```

Klasni dijagram za enemyListNodeStruct:

Polja

- [enemyPtr enemy](#)
pokazivač na strukturu neprijatelja
- struct [enemyListNodeStruct](#) * [next](#)
sledeći element liste
- struct [enemyListNodeStruct](#) * [prev](#)
prethodni element liste

3.5.1 Opširniji opis

Struktura clana liste neprijatelja

3.5.2 Dokumentacija polja

3.5.2.1 enemyPtr enemyListNodeStruct::enemy

pokazivač na strukturu neprijatelja

3.5.2.2 struct enemyListNodeStruct* enemyListNodeStruct::next

sledeći element liste

3.5.2.3 struct enemyListNodeStruct* enemyListNodeStruct::prev

prethodni element liste

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [defines.h](#)

3.6 Dokumentacija strukture enemyStruct

```
#include <defines.h>
```


Polja

- double `x_pos`
x koordinata
- double `y_pos`
y koordinata
- int `counter`
brojač po frejmovima
- int `coolDown`
cooldown ispaljivanja metka
- int `id`
identifikacioni broj tipa neprijatelja (0:obicna, 1:avion sa više health-a, 2:boss follower, 3:kamikaza, 4:glavni boss, 5:meta)
- int `path_id`
identifikacioni broj putanje po kojoj neprijatelj treba da se kreće
- int `texture_slot`
textura koja treba da se iscrtava za neprijatelja
- int `health`
trenutni health neprijatelja
- int `score`
vrednost neprijatelja u poenima
- SDL_Rect `position`
pravougaonik iscrtavanja
- SDL_Rect `texturePosition`
pozicija teksture
- void(* `handleIteration`)(enemyListNodePtr)
pokazivač na odgovarajuću funkciju koja se poziva pri svakoj iteraciji

3.6.1 Opširniji opis

Struktura instance neprijatelja, id {0:obicna neprijatelj, 1:dupli health, 2:follower, 3:meta , 4:kamikaze}

3.6.2 Dokumentacija polja

3.6.2.1 int enemyStruct::coolDown

cooldown ispaljivanja metka

3.6.2.2 int enemyStruct::counter

brojač po frejmovima

3.6.2.3 void(* enemyStruct::handleIteration) (enemyListNodePtr)

pokazivač na odgovarajuću funkciju koja se poziva pri svakoj iteraciji

3.6.2.4 int enemyStruct::health

trenutni health neprijatelja

3.6.2.5 int enemyStruct::id

identifikacioni broj tipa neprijatelja (0:obicaj, 1:avion sa više health-a, 2:boss follower, 3:kamikaza, 4:glavni boss, 5:meta)

3.6.2.6 int enemyStruct::path_id

identifikacioni broj putanje po kojoj neprijatelj treba da se kreće

3.6.2.7 SDL_Rect enemyStruct::position

pravougaonik iscrtavanja

3.6.2.8 int enemyStruct::score

vrednost neprijatelja u poenima

3.6.2.9 int enemyStruct::texture_slot

textura koja treba da se iscrtava za neprijatelja

3.6.2.10 SDL_Rect enemyStruct::texturePosition

pozicija tekstone

3.6.2.11 double enemyStruct::x_pos

x koordinata

3.6.2.12 double enemyStruct::y_pos

y koordinata

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [defines.h](#)

3.7 Dokumentacija strukture gameStruct

```
#include <defines.h>
```

Klasni dijagram za gameStruct:

Polja

- [playerListNodePtr players](#)
pokazivač na aktivne igrače
- [enemyListNodePtr enemies](#)
pokazivač na aktivne neprijatelje
- [effectListNodePtr effects](#)

- pokazivač na aktivne efekte*

 - `bulletListNodePtr` `bullets`
- pokazivač na aktivne metke*

 - `powerUpListNodePtr` `powerUps`
- pokazivač na aktivne powerUp-ove*

 - `int` `playersCount`
- broj igrača*

 - `int` `enemiesCount`
- broj neprijatelja*

 - `int` `effectsCount`
- broj efekata*

 - `int` `bulletsCount`
- broj metkova*

 - `int` `score`
- trenutni broj poena igrača*

 - `int` `frames`
- broj frejmova*

 - `int` `framesFromCont`
- broj frejmova od menija*

 - `int` `game_mode`
- aktivan tip igre*

 - `int` `difficulty`
- težina igre (0:demo, 1:laka, 2:normalna, 3:teška)*

 - `short` `boss_active`
- flag da li je boss aktivan*

 - `short` `final_boss_active`
- flag da li je glavni boss aktivan*

 - `int` `enemySpawnCounter`
- broj stovernih običnih neprijatelja*

 - `SDL_Texture *` `explosion_texture`
- pokazivač na teksturu eksplozije*

 - `SDL_Texture *` `bullet_texture`
- pokazivač na teksturu metka*

 - `SDL_Texture *` `bomb_texture`
- pokazivač na teksturu bombe*

 - `SDL_Texture *` `fumes_texture`
- pokazivač na teksturu dima*

 - `SDL_Texture *` `holeTexture`
- pokazivač na teksturu rupe*

 - `SDL_Texture *` `powerUpTexture`
- pokazivač na teksturu powerUp-a*

 - `SDL_Texture *` `playerTexture`
- pokazivač na teksturu igrača*

 - `SDL_Texture *` `playerTextureLeft`
- pokazivač na teksturu igrača okrenutog ka levo*

 - `SDL_Texture *` `playerTextureRight`
- pokazivač na teksturu igrača okrenutog ka desno*

 - `SDL_Texture *` `pauseTexture`
- pokazivač na teksturu prikazanu u pauzi*

 - `SDL_Texture *` `shieldTexture`
- pokazivač na teksturu štita*

- SDL_Texture * [speedTexture](#)
pokazivač na teksturu koja označava da igrač ima ubrzanje
- SDL_Surface * [pauseSurface](#)
pokazivač na teksturu površine prikazane u pauzi
- FILE * [level_file](#)
pokazivač na fajl za ulitavanje
- SDL_Texture ** [backgroundTiles](#)
matrica delova pozadine
- SDL_Texture ** [enemy_sprites](#)
teksture neprijatelja
- char [HSNames](#) [10][51]
imena u HighScores
- char [HSFileName](#) [256]
ime HighScore fajla
- int [HSScores](#) [10]
poeni u HighScore tabeli
- int [level_matrix](#) [BACKGROUND_GRID_Y][BACKGROUND_GRID_X]
matrica nivoa
- int [level_default](#)
flag da li je podrazumevan nivo aktivan
- int [background_offset](#)
offset pozadine
- int [campaign_level_ind](#)
indeks nivoa kampanje
- int [campaign_row](#)
red u kampanji
- int(* [handleIteration](#))(void)
pokazivač na odgovarajuću funkciju koja se poziva pri svakoj iteraciji

3.7.1 Opširniji opis

Struktura instance igre

3.7.2 Dokumentacija polja

3.7.2.1 int gameStruct::background_offset

offset pozadine

3.7.2.2 SDL_Texture** gameStruct::backgroundTiles

matrica delova pozadine

3.7.2.3 SDL_Texture* gameStruct::bomb_texture

pokazivač na teksturu bombe

3.7.2.4 short gameStruct::boss_active

flag da li je boss aktivan

3.7.2.5 SDL_Texture* gameStruct::bullet_texture

pokazivač na teksturu metka

3.7.2.6 bulletListNodePtr gameStruct::bullets

pokazivač na aktivne metke

3.7.2.7 int gameStruct::bulletsCount

broj metkova

3.7.2.8 int gameStruct::campaign_level_ind

indeks nivoa kampanje

3.7.2.9 int gameStruct::campaign_row

red u kampanji

3.7.2.10 int gameStruct::difficulty

težina igre (0:demo, 1:laka, 2:normalna, 3:teška)

3.7.2.11 effectListNodePtr gameStruct::effects

pokazivač na aktivne efekte

3.7.2.12 int gameStruct::effectsCount

broj efekata

3.7.2.13 enemyListNodePtr gameStruct::enemies

pokazivač na aktivne neprijatelje

3.7.2.14 int gameStruct::enemiesCount

broj neprijatelja

3.7.2.15 SDL_Texture gameStruct::enemy_sprites**

teksture neprijatelja

3.7.2.16 int gameStruct::enemySpawnCounter

broj stovernih običnih neprijatelja

3.7.2.17 SDL_Texture* gameStruct::explosion_texture

pokazivač na teksturu eksplozije

3.7.2.18 short gameStruct::final_boss_active

flag da li je glavni boss aktivan

3.7.2.19 int gameStruct::frames

broj frejmova

3.7.2.20 int gameStruct::framesFromCont

broj frejmova od menija

3.7.2.21 SDL_Texture* gameStruct::fumes_texture

pokazivač na teksturu dima

3.7.2.22 int gameStruct::game_mode

aktivan tip igre

3.7.2.23 int(* gameStruct::handleIteration) (void)

pokazivač na odgovarajuću funkciju koja se poziva pri svakoj iteraciji

3.7.2.24 SDL_Texture* gameStruct::holeTexture

pokazivač na teksturu rupe

3.7.2.25 char gameStruct::HSFileName[256]

ime HighScore fajla

3.7.2.26 char gameStruct::HSNames[10][51]

imena u HighScores

3.7.2.27 int gameStruct::HSScores[10]

poeni u HighScore tabeli

3.7.2.28 int gameStruct::level_default

flag da li je podrazumevan nivo aktivan

3.7.2.29 FILE* gameStruct::level_file

pokazivač na fajl za ulitavanje

3.7.2.30 int gameStruct::level_matrix[BACKGROUND_GRID_Y][BACKGROUND_GRID_X]

matrica nivoa

3.7.2.31 SDL_Surface* gameStruct::pauseSurface

pokazivač na teksturu površine prikazane u pauzi

3.7.2.32 SDL_Texture* gameStruct::pauseTexture

pokazivač na teksturu prikazanu u pauzi

3.7.2.33 playerListNodePtr gameStruct::players

pokazivač na aktivne igrače

3.7.2.34 int gameStruct::playersCount

broj igrača

3.7.2.35 SDL_Texture* gameStruct::playerTexture

pokazivač na teksturu igrača

3.7.2.36 SDL_Texture* gameStruct::playerTextureLeft

pokazivač na teksturu igrača okrenutog ka levo

3.7.2.37 SDL_Texture* gameStruct::playerTextureRight

pokazivač na teksturu igrača okrenutog ka desno

3.7.2.38 powerUpListNodePtr gameStruct::powerUps

pokazivač na aktivne powerUp-ove

3.7.2.39 SDL_Texture* gameStruct::powerUpTexture

pokazivač na teksturu powerUp-a

3.7.2.40 int gameStruct::score

trenutni broj poena igrača

3.7.2.41 SDL_Texture* gameStruct::shieldTexture

pokazivač na teksturu štita

3.7.2.42 SDL_Texture* gameStruct::speedTexture

pokazivač na teksturu koja označava da igrač ima ubrzanje

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [defines.h](#)

3.8 Dokumentacija strukture HSFileMemberStruct

```
#include <game.h>
```

Polja

- char [name](#) [256]
Ime igrača.
- int [score](#)
Rezultat koji je igrač ostvario.

3.8.1 Opširniji opis

Strukturura koja se koristi prilikom snimanja datoteke sa najboljim rezultatima

3.8.2 Dokumentacija polja

3.8.2.1 char HSFileMemberStruct::name[256]

Ime igrača.

3.8.2.2 int HSFileMemberStruct::score

Rezultat koji je igrač ostvario.

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [game/game.h](#)

3.9 Dokumentacija strukture maxLevelSaveStruct

```
#include <game.h>
```

Polja

- double [level](#)
Nivo koji je potrebno zapamtiti.

3.9.1 Opširniji opis

Struktura koja se koristi prilikom snimanja nivoa do kog je predjena kampanja

3.9.2 Dokumentacija polja

3.9.2.1 `double maxLevelSaveStruct::level`

Nivo koji je potrebno zapamtiti.

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [game/game.h](#)

3.10 Dokumentacija strukture `playerListNodeStruct`

```
#include <defines.h>
```

Klasni dijagram za `playerListNodeStruct`:

Polja

- [playerPtr player](#)
pokazivač na strukturu igrača
- `struct playerListNodeStruct * next`
sledeći element liste
- `struct playerListNodeStruct * prev`
prethodni element liste

3.10.1 Opširniji opis

Struktura člana liste igrača

3.10.2 Dokumentacija polja

3.10.2.1 `struct playerListNodeStruct* playerListNodeStruct::next`

sledeći element liste

3.10.2.2 `playerPtr playerListNodeStruct::player`

pokazivač na strukturu igrača

3.10.2.3 `struct playerListNodeStruct* playerListNodeStruct::prev`

prethodni element liste

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [defines.h](#)

3.11 Dokumentacija strukture playerStruct

```
#include <defines.h>
```

Polja

- double `x_pos`
x koordinata
- double `y_pos`
y koordinata
- int `counter`
brojač po frejmovima
- int `bulletCoolDown`
cooldown ispaljivanja metka
- int `bombCoolDown`
cooldown ispaljivanja bombe
- int `health`
trenutni health igrača
- int `movement`
strana na koju se igrač kreće (0:sredina, -1:levo, 1:desno)
- short `shieldActive`
flag da li je aktivan štit
- short `speedActive`
flag da li je aktivno ubrzanje
- SDL_Rect `position`
pravougaonik iscrtavanja
- SDL_Rect `texturePosition`
pozicija tekstone
- void(* `handleIteration`)(playerListNodePtr)
pokazivač na odgovarajuću funkciju koja se poziva pri svakoj iteraciji

3.11.1 Opširniji opis

Struktura instance igrača

3.11.2 Dokumentacija polja

3.11.2.1 int playerStruct::bombCoolDown

cooldown ispaljivanja bombe

3.11.2.2 int playerStruct::bulletCoolDown

cooldown ispaljivanja metka

3.11.2.3 int playerStruct::counter

brojač po frejmovima

3.11.2.4 void(* playerStruct::handleIteration) (playerListNodePtr)

pokazivač na odgovarajuću funkciju koja se poziva pri svakoj iteraciji

3.11.2.5 int playerStruct::health

trenutni health igrača

3.11.2.6 int playerStruct::movement

strana na koju se igrač kreće (0:sredina, -1:levo, 1:desno)

3.11.2.7 SDL_Rect playerStruct::position

pravougaonik iscrtavanja

3.11.2.8 short playerStruct::shieldActive

flag da li je aktivan štit

3.11.2.9 short playerStruct::speedActive

flag da li je aktivno ubrzanje

3.11.2.10 SDL_Rect playerStruct::texturePosition

pozicija teksture

3.11.2.11 double playerStruct::x_pos

x koordinata

3.11.2.12 double playerStruct::y_pos

y koordinata

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [defines.h](#)

3.12 Dokumentacija strukture powerUpListNodeStruct

```
#include <defines.h>
```

Klasni dijagram za powerUpListNodeStruct:

Polja

- [powerUpPtr powerUp](#)
pokazivač na strukturu powerUp-a

- struct [powerUpListNodeStruct](#) * [next](#)
sledeći element liste
- struct [powerUpListNodeStruct](#) * [prev](#)
prethodni element liste

3.12.1 Opširniji opis

Struktura clana liste powerUp-a

3.12.2 Dokumentacija polja

3.12.2.1 struct powerUpListNodeStruct* powerUpListNodeStruct::next

sledeći element liste

3.12.2.2 powerUpPtr powerUpListNodeStruct::powerUp

pokazivač na strukturu powerUp-a

3.12.2.3 struct powerUpListNodeStruct* powerUpListNodeStruct::prev

prethodni element liste

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [defines.h](#)

3.13 Dokumentacija strukture powerUpStruct

```
#include <defines.h>
```

Polja

- double [x_pos](#)
x koordinata
- double [y_pos](#)
y koordinata
- SDL_Rect [position](#)
pravougaonik iscrtavanja
- int [counter](#)
brojač frejmova od inicijalizacije
- int [id](#)
identifikacioni broj tipa powerUp-a (1:štit 2:brzina)
- int(* [handleIteration](#))(powerUpListNodeStruct)
pokazivač na odgovarajuću funkciju koja se poziva pri svakoj iteraciji

3.13.1 Opširniji opis

Struktura instance powerUp-a

3.13.2 Dokumentacija polja

3.13.2.1 `int powerUpStruct::counter`

brojač frejmova od inicijalizacije

3.13.2.2 `int(* powerUpStruct::handleIteration) (powerUpListNodeStruct)`

pokazivač na odgovarajuću funkciju koja se poziva pri svakoj iteraciji

3.13.2.3 `int powerUpStruct::id`

identifikacioni broj tipa powerUp-a (1:štit 2:brzina)

3.13.2.4 `SDL_Rect powerUpStruct::position`

pravougaonik iscrtavanja

3.13.2.5 `double powerUpStruct::x_pos`

x koordinata

3.13.2.6 `double powerUpStruct::y_pos`

y koordinata

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [defines.h](#)

3.14 Dokumentacija strukture saveHeaderStruct

```
#include <savingGame.h>
```

Polja

- `int enemyNum`
- `int bulletNum`
- `int effectNum`
- `int powerUpNum`

3.14.1 Dokumentacija polja

3.14.1.1 `int saveHeaderStruct::bulletNum`

3.14.1.2 `int saveHeaderStruct::effectNum`

3.14.1.3 `int saveHeaderStruct::enemyNum`

3.14.1.4 int saveHeaderStruct::powerUpNum

Dokumentacija ove strukture je napravljena na osnovu datoteke

- game/[savingGame.h](#)

3.15 Dokumentacija strukture textStruct

```
#include <defines.h>
```

Polja

- SDL_Rect [rect](#)
pravougaonik iscrtavanja
- SDL_Color [color](#)
boja teksta
- char [str](#) [5000]
tekst
- int [show](#)
flag da li da se prikazuje tekst
- TTF_Font * [font](#)
font
- SDL_Texture * [texture](#)
tekstura koja se iscrtava

3.15.1 Opširniji opis

Struktura instance koja omogućava ispis teksta

3.15.2 Dokumentacija polja

3.15.2.1 SDL_Color textStruct::color

boja teksta

3.15.2.2 TTF_Font* textStruct::font

font

3.15.2.3 SDL_Rect textStruct::rect

pravougaonik iscrtavanja

3.15.2.4 int textStruct::show

flag da li da se prikazuje tekst

3.15.2.5 char textStruct::str[5000]

tekst

3.15.2.6 SDL_Texture* textStruct::texture

tekstura koja se iscrtava

Dokumentacija ove strukture je napravljena na osnovu datoteke

- [defines.h](#)

Glava 4

Dokumentacija datoteke

4.1 Opis datoteke defines.h

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <string.h>
#include <math.h>
#include <dirent.h>
#include <SDL2/SDL.h>
#include <SDL2/SDL_image.h>
#include <SDL2/SDL_ttf.h>
#include <SDL2/SDL_mixer.h>
```

Graf zavisnosti datoteka za defines.h:

4.2 Opis datoteke draw.c

```
#include "draw.h"
```

Graf zavisnosti datoteka za draw.c:

Funkcije

- void [drawText](#) ([textPtr](#) textTexture)
- void [drawPlayers](#) ()
- void [drawEnemies](#) ()
- void [drawEffects](#) ()
- void [drawBullets](#) ()
- void [drawPowerUps](#) ()
- void [drawBackground](#) ()

4.2.1 Dokumentacija funkcije

4.2.1.1 void drawBackground ()

Funkcija za iscrtavanje tile-ova

4.2.1.2 void drawBullets ()

Funkcija za iscrtavanje metaka

4.2.1.3 void drawEffects ()

Funkcija za iscrtavanje efekata

4.2.1.4 void drawEnemies ()

Funkcija za iscrtavanje neprijatelja

4.2.1.5 void drawPlayers ()

Funkcija za iscrtavanje igrača

4.2.1.6 void drawPowerUps ()

Funkcija za iscrtavanje powerup-ova

4.2.1.7 void drawText (textPtr textTexture)

Funkcija za iscrtavanje teksta pomoću strukture za iscrtavanje teksta

4.3 Opis datoteke draw.h

```
#include "defines.h"  
#include "instances/instances.h"
```

Graf zavisnosti datoteka za draw.h: Ovaj graf pokazuje koje datoteke direktno ili indirektno uključuju ovu datoteku:

Funkcije

- void [drawText](#) (textPtr)
- void [drawPlayers](#) ()
- void [drawEnemies](#) ()
- void [drawPowerUps](#) ()
- void [drawEffects](#) ()
- void [drawBullets](#) ()
- void [drawBackground](#) ()

Promenljive

- SDL_Rect [backgroundRect](#)
Pokazivač na pravougaonik u koji se iscrtava pozadina.
- SDL_Rect [topStripRect](#)
Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.
- SDL_Rect [bottomStripRect](#)
Pokazivač na pravougaonik u koji se iscrtava donji bar.
- SDL_Renderer * [renderer](#)

- Globalni pokazivač na SDL renderer.*
- SDL_Window * [window](#)
Globalni pokazivač na SDL window.
- TTF_Font * [font](#)
Globalni pokazivač na SDL font.
- [gamePtr](#) game
Globalni pokazivač na trenutnu započetu ili pauziranu igru.
- int [lastCampaignLevel](#)
Indeks nivoa do koga je pređena igra.
- Mix_Music * [bgSound](#)
Zvuk motora aviona.
- Mix_Chunk * [fire](#)
Zvuk pucnja.
- Mix_Chunk * [explosionSound](#)
Zvuk eksplozije.

4.3.1 Dokumentacija funkcije

4.3.1.1 void drawBackground ()

Funkcija za iscrtavanje tile-ova

4.3.1.2 void drawBullets ()

Funkcija za iscrtavanje metaka

4.3.1.3 void drawEffects ()

Funkcija za iscrtavanje efekata

4.3.1.4 void drawEnemies ()

Funkcija za iscrtavanje neprijatelja

4.3.1.5 void drawPlayers ()

Funkcija za iscrtavanje igrača

4.3.1.6 void drawPowerUps ()

Funkcija za iscrtavanje powerup-ova

4.3.1.7 void drawText (textPtr textTexture)

Funkcija za iscrtavanje teksta pomoću strukture za iscrtavanje teksta

4.3.2 Dokumentacija promenljive

4.3.2.1 SDL_Rect backgroundRect

Pokazivač na pravougaonik u koji se iscrtava pozadina.

4.3.2.2 `Mix_Music*` `bgSound`

Zvuk motora aviona.

4.3.2.3 `SDL_Rect` `bottomStripRect`

Pokazivač na pravougaonik u koji se iscrtava donji bar.

4.3.2.4 `Mix_Chunk*` `explosionSound`

Zvuk eksplozije.

4.3.2.5 `Mix_Chunk*` `fire`

Zvuk pucnja.

4.3.2.6 `TTF_Font*` `font`

Globalni pokazivač na SDL font.

4.3.2.7 `gamePtr` `game`

Globalni pokazivač na trenutnu započetu ili pauziranu igru.

4.3.2.8 `int` `lastCampaignLevel`

Indeks nivoa do koga je pređena igra.

4.3.2.9 `SDL_Renderer*` `renderer`

Globalni pokazivač na SDL renderer.

4.3.2.10 `SDL_Rect` `topStripRect`

Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.

4.3.2.11 `SDL_Window*` `window`

Globalni pokazivač na SDL window.

4.4 Opis datoteke `game/campaign.c`

```
#include "campaign.h"  
Graf zavisnosti datoteka za campaign.c:
```

Promenljive

- `int` `campaign_background` [3][30][`BACKGROUND_GRID_X+1`]

4.4.1 Dokumentacija promenljive

4.4.1.1 `int campaign_background[3][30][BACKGROUND_GRID_X+1]`

4.5 Opis datoteke game/campaign.h

```
#include "../defines.h"
```

Graf zavisnosti datoteka za campaign.h: Ovaj graf pokazuje koje datoteke direktno ili indirektno uključuju ovu datoteku:

Promenljive

- `int campaign_background [3][30][5]`
Niz matrica u kojima se čuvaju definicije nivoa iz kampanje.

4.5.1 Dokumentacija promenljive

4.5.1.1 `int campaign_background[3][30][5]`

Niz matrica u kojima se čuvaju definicije nivoa iz kampanje.

4.6 Opis datoteke game/collisions.c

```
#include "collisions.h"
```

Graf zavisnosti datoteka za collisions.c:

Makro zamene

- `#define MIN(X, Y) (((X) < (Y)) ? (X) : (Y))`

Funkcije

- `void detectCollisions ()`
- `int detectPlayerEnemyCollision (playerPtr player, enemyPtr enemy)`
- `int detectBulletEnemyCollision (bulletPtr bullet, enemyPtr enemy)`
- `int detectBulletPlayerCollision (bulletPtr bullet, playerPtr player)`
- `int detectPowerUpPlayerCollision (powerUpPtr powerUp, playerPtr player)`
- `int detectPowerUpEnemyCollision (powerUpPtr powerUp, enemyPtr enemy)`

4.6.1 Dokumentacija makro zamene

4.6.1.1 `#define MIN(X, Y) (((X) < (Y)) ? (X) : (Y))`

4.6.2 Dokumentacija funkcije

4.6.2.1 `int detectBulletEnemyCollision (bulletPtr bullet, enemyPtr enemy)`

Funkcija za proveru i hendlovanje sudara metka i protivnika

4.6.2.2 int detectBulletPlayerCollision (bulletPtr *bullet*, playerPtr *player*)

Funkcija za proveru i hendlovanje sudara metka i igraca

4.6.2.3 void detectCollisions (void)

Funkcija za detekciju kolizije izmedju objekata na ekranu

- igrač - neprijatelj
- igrač - metak
- neprijatelj - metak
- igrač - pojačanje

4.6.2.4 int detectPlayerEnemyCollision (playerPtr *player*, enemyPtr *enemy*)

Funkcija za proveru i hendlovanje sudara igraca i protivnika

4.6.2.5 int detectPowerUpEnemyCollision (powerUpPtr *powerUp*, enemyPtr *enemy*)

Funkcija za proveru i hendlovanje kolizije izmedju powerUp-a i neprijatelja

4.6.2.6 int detectPowerUpPlayerCollision (powerUpPtr *powerUp*, playerPtr *player*)

Funkcija za proveru i hendlovanje kolizije izmedju powerUp-a i igraca

4.7 Opis datoteke game/collisions.h

```
#include "../defines.h"
#include "game.h"
#include "../instances/instances.h"
```

Graf zavisnosti datoteka za collisions.h: Ovaj graf pokazuje koje datoteke direktno ili indirektno uključuju ovu datoteku:

Funkcije

- void [detectCollisions](#) (void)
- int [detectPlayerEnemyCollision](#) (playerPtr, enemyPtr)
- int [detectBulletEnemyCollision](#) (bulletPtr, enemyPtr)
- int [detectBulletPlayerCollision](#) (bulletPtr, playerPtr)
- int [detectPowerUpPlayerCollision](#) (powerUpPtr, playerPtr)
- int [detectPowerUpEnemyCollision](#) (powerUpPtr, enemyPtr)

Promenljive

- SDL_Rect [backgroundRect](#)
Pokazivač na pravougaonik u koji se iscrtaava pozadina.
- SDL_Rect [topStripRect](#)
Pokazivač na pravougaonik u koji se iscrtaava gornji bar u igri.

- SDL_Rect [bottomStripRect](#)
Pokazivač na pravougaonik u koji se is crtava donji bar.
- SDL_Renderer * [renderer](#)
Globalni pokazivač na SDL renderer.
- SDL_Window * [window](#)
Globalni pokazivač na SDL window.
- TTF_Font * [font](#)
Globalni pokazivač na SDL font.
- [gamePtr](#) [game](#)
Globalni pokazivač na trenutnu započetu ili pauziranu igru.
- int [lastCampaignLevel](#)
Indeks nivoa do koga je pređena igra.
- Mix_Music * [bgSound](#)
Zvuk motora aviona.
- Mix_Chunk * [fire](#)
Zvuk pucnja.
- Mix_Chunk * [explosionSound](#)
Zvuk eksplozije.

4.7.1 Dokumentacija funkcije

4.7.1.1 int detectBulletEnemyCollision ([bulletPtr](#) *bullet*, [enemyPtr](#) *enemy*)

Funkcija za proveru i hendlovanje sudara metka i protivnika

4.7.1.2 int detectBulletPlayerCollision ([bulletPtr](#) *bullet*, [playerPtr](#) *player*)

Funkcija za proveru i hendlovanje sudara metka i igrača

4.7.1.3 void detectCollisions (void)

Funkcija za detekciju kolizije izmedju objekata na ekranu

- igrač - neprijatelj
- igrač - metak
- neprijatelj - metak
- igrač - pojačanje

4.7.1.4 int detectPlayerEnemyCollision ([playerPtr](#) *player*, [enemyPtr](#) *enemy*)

Funkcija za proveru i hendlovanje sudara igrača i protivnika

4.7.1.5 int detectPowerUpEnemyCollision ([powerUpPtr](#) *powerUp*, [enemyPtr](#) *enemy*)

Funkcija za proveru i hendlovanje kolizije izmedju powerUp-a i neprijatelja

4.7.1.6 int detectPowerUpPlayerCollision ([powerUpPtr](#) *powerUp*, [playerPtr](#) *player*)

Funkcija za proveru i hendlovanje kolizije izmedju powerUp-a i igrača

4.7.2 Dokumentacija promenljive

4.7.2.1 SDL_Rect backgroundRect

Pokazivač na pravougaonik u koji se iscrtava pozadina.

4.7.2.2 Mix_Music* bgSound

Zvuk motora aviona.

4.7.2.3 SDL_Rect bottomStripRect

Pokazivač na pravougaonik u koji se iscrtava donji bar.

4.7.2.4 Mix_Chunk* explosionSound

Zvuk eksplozije.

4.7.2.5 Mix_Chunk* fire

Zvuk pucnja.

4.7.2.6 TTF_Font* font

Globalni pokazivač na SDL font.

4.7.2.7 gamePtr game

Globalni pokazivač na trenutnu započetu ili pauziranu igru.

4.7.2.8 int lastCampaignLevel

Indeks nivoa do koga je pređena igra.

4.7.2.9 SDL_Renderer* renderer

Globalni pokazivač na SDL renderer.

4.7.2.10 SDL_Rect topStripRect

Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.

4.7.2.11 SDL_Window* window

Globalni pokazivač na SDL window.

4.8 Opis datoteke game/game.c

```
#include "game.h"
```

Graf zavisnosti datoteka za game.c:

Funkcije

- void `renderGame` ()
- void `updateEffects` ()
- void `swap_tile_rows` (int a, int b)
- int * `background_load_row` ()
- int * `background_campaign` ()
- int * `background_random` ()
- void `background_swap` (int *row)
- void `doBackgroundJob` ()
- void `runGame` ()

4.8.1 Dokumentacija funkcije

4.8.1.1 int* `background_campaign` ()

Ucitava sledeci red indeksa tileova iz kampanje

4.8.1.2 int* `background_load_row` ()

Ucitava sledeci red indeksa tileova za nivo iz fajla u prvi red matrice

4.8.1.3 int* `background_random` ()

Generise random novi red indeksa tileova

4.8.1.4 void `background_swap` (int * row)

Ubacuje novi red tileova u nivo (paziti na deklaracije u [game.h](#))

4.8.1.5 void `doBackgroundJob` ()

U svakom frejmu poziva funkcije koje updateuju pozadinu

4.8.1.6 void `renderGame` (void)

Funkcija koja iscrtava frejm igre

- objekti se iscrtavaju po redosledu: efekti, neprijatelji, metci, pojačanja i igrač

4.8.1.7 void runGame (void)

Funkcija sa glavnom petljom igre

- prepoznaje događaje i cuva ih u nizu events
- redom poziva funkcije za ažuriranje: igrača, neprijatelja, efekata, metaka i pojačanja
- poziva funkciju koja detektuje i hendluje sudate objekata
- poziva funkciju koja ažurira generalno stanje igre
- poziva funkciju koja iscrtava stanje igre na ekran
- održava ujednačen broj frejmova po sekundi

4.8.1.8 void swap_tile_rows (int a, int b)

Zamenjuje dva reda backgrounda u matrici ucitanog nivoa (po indeksima)

4.8.1.9 void updateEffects ()

Funkcija koja poziva handleIteration funkciju za svaki efekat na ekranu

4.9 Opis datoteke game/game.h

```
#include "campaign.h"
#include "../defines.h"
#include "../draw.h"
#include "../instances/instances.h"
#include "collisions.h"
```

Graf zavisnosti datoteka za game.h: Ovaj graf pokazuje koje datoteke direktno ili indirektno uključuju ovu datoteku:

Strukture i unije

- struct [maxLevelSaveStruct](#)
- struct [HSFileMemberStruct](#)

Makro zamene

- #define [TEXTS_NUM](#) 15
- #define [PLAYERS_NUM](#) 1

Definicije tipa

- typedef struct [maxLevelSaveStruct](#) * [maxLevelSavePtr](#)
- typedef struct [HSFileMemberStruct](#) * [HSFileMemberPtr](#)

Funkcije

- void [renderGame](#) (void)
- void [runGame](#) (void)

Promenljive

- SDL_Rect [backgroundRect](#)
Pokazivač na pravougaonik u koji se iscrtava pozadina.
- SDL_Rect [topStripRect](#)
Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.
- SDL_Rect [bottomStripRect](#)
Pokazivač na pravougaonik u koji se iscrtava donji bar.
- SDL_Renderer * [renderer](#)
Globalni pokazivač na SDL renderer.
- SDL_Window * [window](#)
Globalni pokazivač na SDL window.
- TTF_Font * [font](#)
Globalni pokazivač na SDL font.
- [gamePtr](#) [game](#)
Globalni pokazivač na trenutnu započetu ili pauziranu igru.
- int [lastCampaignLevel](#)
Indeks nivoa do koga je pređena igra.
- Mix_Music * [bgSound](#)
Zvuk motora aviona.
- Mix_Chunk * [fire](#)
Zvuk pucnja.
- Mix_Chunk * [explosionSound](#)
Zvuk eksplozije.
- SDL_Texture * [backgroundTexture](#)
- SDL_Texture * [gameOverTexture](#)
Tekstura koja se prikazuje nakon izgubljene partije.
- SDL_Texture * [levelClearTexture](#)
Tekstura koja se prikazuje nakon predjenog nivoa.
- SDL_Texture * [HSBackgroundTexture](#)
Tekstura za pozadinu high score liste.
- SDL_Rect [gameOverRect](#)
- SDL_Rect [HSBackgroundRect](#)
- [textPtr](#) [texts](#) [[TEXTS_NUM](#)]
Niz struktura za ispis teksta.
- [playerPtr](#) [players](#) [[PLAYERS_NUM](#)]
Niz pokazivača na igrače.
- char [events](#) [256]
Niz zabeleženih događaja sa tastature.

4.9.1 Dokumentacija makro zamene

4.9.1.1 `#define PLAYERS_NUM 1`

4.9.1.2 `#define TEXTS_NUM 15`

4.9.2 Dokumentacija definicije tipa

4.9.2.1 `typedef struct HSFileMemberStruct * HSFileMemberPtr`

Struktura koja se koristi prilikom snimanja datoteke sa najboljim rezultatima

4.9.2.2 `typedef struct maxLevelSaveStruct * maxLevelSavePtr`

Struktura koja se koristi prilikom snimanja nivoa do kog je predjena kampanja

4.9.3 Dokumentacija funkcije

4.9.3.1 `void renderGame (void)`

Funkcija koja iscrtava frejm igre

- objekti se iscrtavaju po redosledu: efekti, neprijatelji, metci, pojačanja i igrač

4.9.3.2 `void runGame (void)`

Funkcija sa glavnom petljom igre

- prepoznaje događaje i cuva ih u nizu events
- redom poziva funkcije za ažuriranje: igrača, neprijatelja, efekata, metaka i pojačanja
- poziva funkciju koja detektuje i hendluje sudate objekata
- poziva funkciju koja ažurira generalno stanje igre
- poziva funkciju koja iscrtava stanje igre na ekran
- održava ujednačen broj frejmova po sekundi

4.9.4 Dokumentacija promenljive

4.9.4.1 `SDL_Rect backgroundRect`

Pokazivač na pravougaonik u koji se iscrtava pozadina.

4.9.4.2 `SDL_Texture* backgroundTexture`

4.9.4.3 `Mix_Music* bgSound`

Zvuk motora aviona.

4.9.4.4 `SDL_Rect bottomStripRect`

Pokazivač na pravougaonik u koji se iscrtava donji bar.

4.9.4.5 `char events[256]`

Niz zabeleženih događaja sa tastature.

4.9.4.6 `Mix_Chunk* explosionSound`

Zvuk eksplozije.

4.9.4.7 Mix_Chunk* fire

Zvuk pucnja.

4.9.4.8 TTF_Font* font

Globalni pokazivač na SDL font.

4.9.4.9 gamePtr game

Globalni pokazivač na trenutnu započetu ili pauziranu igru.

4.9.4.10 SDL_Rect gameOverRect**4.9.4.11 SDL_Texture* gameOverTexture**

Tekstura koja se prikazuje nakon izgubljene partije.

4.9.4.12 SDL_Rect HSBackgroundRect**4.9.4.13 SDL_Texture* HSBackgroundTexture**

Tekstura za pozadinu high score liste.

4.9.4.14 int lastCampaignLevel

Indeks nivoa do koga je pređena igra.

4.9.4.15 SDL_Texture* levelClearTexture

Tekstura koja se prikazuje nakon predjenog nivoa.

4.9.4.16 playerPtr players[PLAYERS_NUM]

Niz pokazivača na igrače.

4.9.4.17 SDL_Renderer* renderer

Globalni pokazivač na SDL renderer.

4.9.4.18 textPtr texts[TEXTS_NUM]

Niz struktura za ispist teksta.

4.9.4.19 SDL_Rect topStripRect

Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.

4.9.4.20 SDL_Window* window

Globalni pokazivač na SDL window.

4.10 Opis datoteke game/handlers.h

```
#include "../defines.h"
#include "game.h"
#include "../instances/instances.h"
```

Graf zavisnosti datoteka za handlers.h: Ovaj graf pokazuje koje datoteke direktno ili indirektno uključuju ovu datoteku:

Funkcije

- int [gameIterationHandlerDefault](#) ()
- int [playerIterationHandlerDefault](#) (playerListNodePtr)
- int [playerIterationHandlerDemo](#) (playerListNodePtr)
- int [enemyIterationHandlerDefault](#) (enemyListNodePtr)
- int [enemyIterationHandlerFollower](#) (enemyListNodePtr)
- int [enemyIterationHandlerKamikaze](#) (enemyListNodePtr)
- int [enemyIterationHandlerBoss](#) (enemyListNodePtr)
- int [enemyIterationHandlerGroundTarget](#) (enemyListNodePtr)
- int [powerUpIterationHandler](#) (powerUpListNodePtr)
- int [bulletIterationHandlerBullet](#) (bulletListNodePtr)
- int [bulletIterationHandlerBomb](#) (bulletListNodePtr)
- int [effectIterationHandlerExhaust](#) (effectListNodePtr)
- int [effectIterationHandlerAirExplosion](#) (effectListNodePtr)
- int [effectIterationHandlerLandExplosion](#) (effectListNodePtr)
- int [effectIterationHandlerHole](#) (effectListNodePtr)
- int [effectIterationHandlerFallingPlane](#) (effectListNodePtr)
- int [effectIterationHandlerPowerUp](#) (effectListNodePtr)

Promenljive

- SDL_Rect [backgroundRect](#)
Pokazivač na pravougaonik u koji se iscrtava pozadina.
- SDL_Rect [topStripRect](#)
Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.
- SDL_Rect [bottomStripRect](#)
Pokazivač na pravougaonik u koji se iscrtava donji bar.
- SDL_Renderer * [renderer](#)
Globalni pokazivač na SDL renderer.
- SDL_Window * [window](#)
Globalni pokazivač na SDL window.
- TTF_Font * [font](#)
Globalni pokazivač na SDL font.
- [gamePtr](#) [game](#)
Globalni pokazivač na trenutnu započetu ili pauziranu igru.
- int [lastCampaignLevel](#)
Indeks nivoa do koga je pređena igra.
- Mix_Music * [bgSound](#)

Zvuk motora aviona.

- Mix_Chunk * [fire](#)

Zvuk pucnja.

- Mix_Chunk * [explosionSound](#)

Zvuk eksplozije.

- SDL_Rect [gameOverRect](#)
- SDL_Rect [HSBackgroundRect](#)

4.10.1 Dokumentacija funkcije

4.10.1.1 int bulletIterationHandlerBomb (bulletListNodePtr *bulletNode*)

Osnovno hendlovanje pozicije bombe u svakom frejmu

- bomba se sama unistava posle nekoliko frejmove kako bi simulirala efekat "padanja" na zemlju

Parametri

<i>bulletNode</i>	Element liste metkova koji sadrzi podatke o trenutnoj bombi
-------------------	---

4.10.1.2 int bulletIterationHandlerBullet (bulletListNodePtr *bulletNode*)

Osnovno hendlovanje pozicije metka u svakom frejmu

- metak se pomera ka gore ili dole u zavisnosti od objekta koji ga je ispalio
- u slucaju da metak ispadne sa ekrana, memorija se oslobadja

Parametri

<i>bulletNode</i>	Element liste metkova koji sadrzi podatke o trenutnom metku
-------------------	---

4.10.1.3 int effectIterationHandlerAirExplosion (effectListNodePtr *effectNode*)

Funkcija koja radi update pojedinacnog efekta eksplozije u vazduhu, u svakom frejmu

- efekat eksplozije se smanjuje svaka tri frejma kako bi simulirao "nestanak"

Parametri

<i>effectNode</i>	Element liste efekata koji sadrzi podatke o trenutnom efektu
-------------------	--

4.10.1.4 int effectIterationHandlerExhaust (effectListNodePtr *effectNode*)

Funkcija koja radi update pojedinacnog efekta dima iza aviona svakog frejma

- efekat dima se smanjuje svaka cetiri frejma kako bi simulirao efekat "nestajanja"

Parametri

<i>effectNode</i>	Element liste efekata koji sadrzi podatke o trenutnom efektu
-------------------	--

4.10.1.5 int effectIterationHandlerFallingPlane (effectListNodePtr effectNode)

Funkcija koja radi update pojedinacnog efekta padanja aviona, u svakom frejmu

- efekat padanja aviona se smanjuje u svakom frejmu kako bi se relano simuliralo rusenje aviona na zemlju/u vodu
- efekat se pomera po y osi brzion pomreanja pozadine

Parametri

<i>effectNode</i>	Element liste efekata koji sadrzi podatke o trenutnom efektu
-------------------	--

4.10.1.6 int effectIterationHandlerHole (effectListNodePtr effectNode)

Funkcija koja radi update pojedinacnog efekta rupe na zemlji, u svakom frejmu

- efekat rupe na zemlji se pomera po y osi brzinom pomeranja pozadine

Parametri

<i>effectNode</i>	Element liste efekata koji sadrzi podatke o trenutnom efektu
-------------------	--

4.10.1.7 int effectIterationHandlerLandExplosion (effectListNodePtr effectNode)

Funkcija koja radi update pojedinacnog efekta eksplozije na zemlji, u svakom frejmu

- efekat eksplozije se smanjuje svaka tri frejma kako bi simulirao "nestanak"

Parametri

<i>effectNode</i>	Element liste efekata koji sadrzi podatke o trenutnom efektu
-------------------	--

4.10.1.8 int effectIterationHandlerPowerUp (effectListNodePtr effectNode)

Funkcija koja radi update pojedinacnog efekta powerUp-a

- efekat powerUp-a je vezan za igrača i pomera se zajedno sa igračem
- ako je igrač unisten dok je efekat aktivan, efekat se takodje unistava

Parametri

<i>effectNode</i>	Element liste efekata koji sadrzi podatke o trenutnom efektu
-------------------	--

4.10.1.9 int enemyIterationHandlerBoss (enemyListNodePtr enemyNode)

Funkcija koja radi update pojedinacnog boss neprijatelja, a poziva se u svakom frejmu

- boss ima predefinisanu trajektoriju po y osi, a po x osi prati igrača
- boss ima dve faze, normalnu u kojoj puca na svakih 75/50/25 frejmova, i hyper fazu u kojoj puca na svakih 10 frejmova i koja traje nekoliko sekundi (cooldown pucanja zavisi od težine igre)

Parametri

<i>enemyNode</i>	Element liste neprijatelja koji sadrzi podatke o trenutnom neprijatelju
------------------	---

4.10.1.10 int enemyIterationHandlerDefault (enemyListNodePtr enemyNode)

Funkcija koja radi update pojedinacnog default neprijatelja, a poziva se u svakom frejmu. U default neprijatelje spadaju obični neprijatelji i neprijatelji u formaciji

- obični neprijatelji se kreću po 4 putanje u zavisnosti od njihovog path_id-a
- neprijatelji u formaciji se kreću po zajedničkoj putanji i path_id im je 4

Parametri

<i>enemyNode</i>	Element liste neprijatelja koji sadrzi podatke o trenutnom neprijatelju
------------------	---

4.10.1.11 int enemyIterationHandlerFollower (enemyListNodePtr enemyNode)

Funkcija koja radi update pojedinacnog follower neprijatelja (koji prati igrača i puca u njega), a poziva se u svakom frejmu

- follower neprijatelj ima predefinisanu trajektoriju po y osi, a po x osi prati igrača i puca

Parametri

<i>enemyNode</i>	Element liste neprijatelja koji sadrzi podatke o trenutnom neprijatelju
------------------	---

4.10.1.12 int enemyIterationHandlerGroundTarget (enemyListNodePtr enemyNode)

Funkcija koja radi update pojedinacne mete na zemlji/u vodi, a poziva se u svakom frejmu

- meta je fiksirana po x osi, a po y osi se pomera brzinom pomeranja y ose i tako stvara iluziju kretanja sa pozadinom
- funkcija proverava da li je trenutna meta izašla sa ekrana (y koordinata joj je veca od visine prozora za igru)

Parametri

<i>enemyNode</i>	Element liste neprijatelja koji sadrzi podatke o trenutnoj meti
------------------	---

4.10.1.13 int enemyIterationHandlerKamikaze (enemyListNodePtr enemyNode)

Funkcija koja radi update pojedinacnog kamikaza neprijatelja (koji pokušava da udari u igrača), a poziva se u svakom frejmu -U zavisnosti od trenutne pozicije igrača na x i y osi, neprijatelj se pomera direktno ka igraču bez pucanja

Parametri

<i>enemyNode</i>	Element liste neprijatelja koji sadrzi podatke o trenutnom neprijatelju
------------------	---

4.10.1.14 int gameIterationHandlerDefault ()

Funkcija koja radi update stanja igre u svakom frejmu (poziva hendlere za sve objekte u igrici, update-uje gui, spawnuje powerup, enemy-je, ispisuje HS ako je ispunjen uslov kraja igre)

- u zavisnosti od podesene težine stvaranje neprijatelja je cesce/redje
- proverava trenutnu pozadinu kako bi se mete stvarale na razlicitim mestima u zavisnosti od pozadine (npr. na mapi sa vodom na vodi ce se stvoriti brod a na ostrvu tenk)
- u slucaju unosenja odgovarajuće komande vraca vrednost 0 kako bi se igra pauzirala

4.10.1.15 int playerIterationHandlerDefault (playerListNodePtr playerNode)

Funkcija koja na osnovu komandi sa tastature radi update pozicije igrača, stvara metkove ako su odgovarajuće komande unete i vodi racuna o trenutno aktivnim power-upovima

Parametri

<i>playerNode</i>	Element liste igrača koji sadrži podatke o trenutnom igraču
-------------------	---

4.10.1.16 int playerIterationHandlerDemo (*playerListNodePtr playerNode*)

Funkcija koja radi update igrača za demo igrice, sa predefinisanim vestackom inteligencijom, a poziva se u svakom frejmu

- prvi prioritet: pozicioniranje ka meti
- drugi prioritet: pozicioniranje prema powerUp-ovima
- treci prioritet: unistavanje najblizeg neprijatelja

Parametri

<i>playerNode</i>	Element liste igrača koji sadrži podatke o trenutnom igraču
-------------------	---

4.10.1.17 int powerUpIterationHandler (*powerUpListNodePtr powerUpNode*)

Osnovno hendlovanje pozicije powerupa u svakom frejmu

- powerUp se pomera ka dole brzinom pomeranja pozadine kako bi se simulirao efekat pracenja pozadine
- u slucaju da y koordinata powerUp-a bude veca od visine prozora igre, powerUp se brise

Parametri

<i>powerUpNode</i>	Element liste powerUp-ova koji sadrži podatke o trenutnom powerUp-u
--------------------	---

4.10.2 Dokumentacija promenljive

4.10.2.1 SDL_Rect backgroundRect

Pokazivač na pravougaonik u koji se iscrtava pozadina.

4.10.2.2 Mix_Music* bgSound

Zvuk motora aviona.

4.10.2.3 SDL_Rect bottomStripRect

Pokazivač na pravougaonik u koji se iscrtava donji bar.

4.10.2.4 Mix_Chunk* explosionSound

Zvuk eksplozije.

4.10.2.5 Mix_Chunk* fire

Zvuk pucnja.

4.10.2.6 TTF_Font* font

Globalni pokazivač na SDL font.

4.10.2.7 gamePtr game

Globalni pokazivač na trenutnu započetu ili pauziranu igru.

4.10.2.8 SDL_Rect gameOverRect

4.10.2.9 SDL_Rect HSBackgroundRect

4.10.2.10 int lastCampaignLevel

Indeks nivoa do koga je pređena igra.

4.10.2.11 SDL_Renderer* renderer

Globalni pokazivač na SDL renderer.

4.10.2.12 SDL_Rect topStripRect

Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.

4.10.2.13 SDL_Window* window

Globalni pokazivač na SDL window.

4.11 Opis datoteke game/iterationHandlers.c

```
#include "handlers.h"
```

Graf zavisnosti datoteka za iterationHandlers.c:

Funkcije

- int [gameIterationHandlerDefault](#) ()
- int [playerIterationHandlerDefault](#) (playerListNodePtr playerNode)
- int [playerIterationHandlerDemo](#) (playerListNodePtr playerNode)
- int [enemyIterationHandlerDefault](#) (enemyListNodePtr enemyNode)
- int [enemyIterationHandlerFollower](#) (enemyListNodePtr enemyNode)
- int [enemyIterationHandlerKamikaze](#) (enemyListNodePtr enemyNode)
- int [enemyIterationHandlerBoss](#) (enemyListNodePtr enemyNode)
- int [enemyIterationHandlerGroundTarget](#) (enemyListNodePtr enemyNode)
- int [bulletIterationHandlerBullet](#) (bulletListNodePtr bulletNode)
- int [bulletIterationHandlerBomb](#) (bulletListNodePtr bulletNode)
- int [powerUpIterationHandler](#) (powerUpListNodePtr powerUpNode)
- int [effectIterationHandlerExhaust](#) (effectListNodePtr effectNode)
- int [effectIterationHandlerAirExplosion](#) (effectListNodePtr effectNode)
- int [effectIterationHandlerLandExplosion](#) (effectListNodePtr effectNode)
- int [effectIterationHandlerHole](#) (effectListNodePtr effectNode)
- int [effectIterationHandlerFallingPlane](#) (effectListNodePtr effectNode)
- int [effectIterationHandlerPowerUp](#) (effectListNodePtr effectNode)

4.11.1 Dokumentacija funkcije

4.11.1.1 int bulletIterationHandlerBomb (bulletListNodePtr bulletNode)

Osnovno hendlovanje pozicije bombe u svakom frejmu

- bomba se sama unistava posle nekoliko frejmove kako bi simulirala efekat "padanja" na zemlju

Parametri

<i>bulletNode</i>	Element liste metkova koji sadrzi podatke o trenutnoj bombi
-------------------	---

4.11.1.2 int bulletIterationHandlerBullet (bulletListNodePtr bulletNode)

Osnovno hendlovanje pozicije metka u svakom frejmu

- metak se pomera ka gore ili dole u zavisnosti od objekta koji ga je ispalio
- u slucaju da metak ispadne sa ekrana, memorija se oslobadja

Parametri

<i>bulletNode</i>	Element liste metkova koji sadrzi podatke o trenutnom metku
-------------------	---

4.11.1.3 int effectIterationHandlerAirExplosion (effectListNodePtr effectNode)

Funkcija koja radi update pojedinacnog efekta eksplozije u vazduhu, u svakom frejmu

- efekat eksplozije se smanjuje svaka tri frejma kako bi simulirao "nestanak"

Parametri

<i>effectNode</i>	Element liste efekata koji sadrzi podatke o trenutnom efektu
-------------------	--

4.11.1.4 int effectIterationHandlerExhaust (effectListNodePtr effectNode)

Funkcija koja radi update pojedinacnog efekta dima iza aviona svakog frejma

- efekat dima se smanjuje svaka cetiri frejma kako bi simulirao efekat "nestajanja"

Parametri

<i>effectNode</i>	Element liste efekata koji sadrzi podatke o trenutnom efektu
-------------------	--

4.11.1.5 int effectIterationHandlerFallingPlane (effectListNodePtr effectNode)

Funkcija koja radi update pojedinacnog efekta padanja aviona, u svakom frejmu

- efekat padanja aviona se smanjuje u svakom frejmu kako bi se relano simuliralo rusenje aviona na zemlju/u vodu
- efekat se pomera po y osi brzion pomreanja pozadine

Parametri

<i>effectNode</i>	Element liste efekata koji sadrzi podatke o trenutnom efektu
-------------------	--

4.11.1.6 `int effectIterationHandlerHole (effectListNodePtr effectNode)`

Funkcija koja radi update pojedinačnog efekta rupe na zemlji, u svakom frejmu

- efekat rupe na zemlji se pomera po y osi brzinom pomeranja pozadine

Parametri

<i>effectNode</i>	Element liste efekata koji sadrzi podatke o trenutnom efektu
-------------------	--

4.11.1.7 `int effectIterationHandlerLandExplosion (effectListNodePtr effectNode)`

Funkcija koja radi update pojedinačnog efekta eksplozije na zemlji, u svakom frejmu

- efekat eksplozije se smanjuje svaka tri frejma kako bi simulirao "nestanak"

Parametri

<i>effectNode</i>	Element liste efekata koji sadrzi podatke o trenutnom efektu
-------------------	--

4.11.1.8 `int effectIterationHandlerPowerUp (effectListNodePtr effectNode)`

Funkcija koja radi update pojedinačnog efekta powerUp-a

- efekat powerUp-a je vezan za igrača i pomera se zajedno sa igračem
- ako je igrač uništen dok je efekat aktivan, efekat se takodje unistava

Parametri

<i>effectNode</i>	Element liste efekata koji sadrzi podatke o trenutnom efektu
-------------------	--

4.11.1.9 `int enemyIterationHandlerBoss (enemyListNodePtr enemyNode)`

Funkcija koja radi update pojedinačnog boss neprijatelja, a poziva se u svakom frejmu

- boss ima predefinisane trajektorije po y osi, a po x osi prati igrača
- boss ima dve faze, normalnu u kojoj puca na svakih 75/50/25 frejmova, i hyper fazu u kojoj puca na svakih 10 frejmova i koja traje nekoliko sekundi (cooldown pucanja zavisi od težine igre)

Parametri

<i>enemyNode</i>	Element liste neprijatelja koji sadrzi podatke o trenutnom neprijatelju
------------------	---

4.11.1.10 `int enemyIterationHandlerDefault (enemyListNodePtr enemyNode)`

Funkcija koja radi update pojedinačnog default neprijatelja, a poziva se u svakom frejmu. U default neprijatelje spadaju obični neprijatelji i neprijatelji u formaciji

- obični neprijatelji se kreću po 4 putanje u zavisnosti od njihovog path_id-a
- neprijatelji u formaciji se kreću po zajedničkoj putanji i path_id im je 4

Parametri

<i>enemyNode</i>	Element liste neprijatelja koji sadrzi podatke o trenutnom neprijatelju
------------------	---

4.11.1.11 int enemyIterationHandlerFollower (enemyListNodePtr enemyNode)

Funkcija koja radi update pojedinacnog follower neprijatelja (koji prati igrača i puca u njega), a poziva se u svakom frejmu

- follower neprijatelj ima predefinisanu trajektoriju po y osi, a po x osi prati igrača i puca

Parametri

<i>enemyNode</i>	Element liste neprijatelja koji sadrzi podatke o trenutnom neprijatelju
------------------	---

4.11.1.12 int enemyIterationHandlerGroundTarget (enemyListNodePtr enemyNode)

Funkcija koja radi update pojedinacne mete na zemlji/u vodi, a poziva se u svakom frejmu

- meta je fiksirana po x osi, a po y osi se pomera brzinom pomeranja y ose i tako stvara iluziju kretanja sa pozadinom
- funkcija proverava da li je trenutna meta izašla sa ekrana (y koordinata joj je veca od visine prozora za igru)

Parametri

<i>enemyNode</i>	Element liste neprijatelja koji sadrzi podatke o trenutnoj meti
------------------	---

4.11.1.13 int enemyIterationHandlerKamikaze (enemyListNodePtr enemyNode)

Funkcija koja radi update pojedinacnog kamikaza neprijatelja (koji pokušava da udari u igrača), a poziva se u svakom frejmu -U zavisnosti od trenutne pozicije igrača na x i y osi, neprijatelj se pomera direktno ka igraču bez pucanja

Parametri

<i>enemyNode</i>	Element liste neprijatelja koji sadrzi podatke o trenutnom neprijatelju
------------------	---

4.11.1.14 int gameIterationHandlerDefault ()

Funkcija koja radi update stanja igre u svakom frejmu (poziva hendlere za sve objekte u igrici, update-uje gui, spawnuje powerup, enemy-je, ispisuje HS ako je ispunjen uslov kraja igre)

- u zavisnosti od podesene težine stvaranje neprijatelja je cesce/redje
- proverava trenutnu pozadinu kako bi se mete stvarale na razlicitim mestima u zavisnosti od pozadine (npr. na mapi sa vodom na vodi ce se stvoriti brod a na ostrvu tenk)
- u slucaju unosenja odgovarajuće komande vraća vrednost 0 kako bi se igra pauzirala

4.11.1.15 int playerIterationHandlerDefault (playerListNodePtr playerNode)

Funkcija koja na osnovu komandi sa tastature radi update pozicije igrača, stvara metkove ako su odgovarajuće komande unete i vodi racuna o trenutno aktivnim power-upovima

Parametri

<i>playerNode</i>	Element liste igrača koji sadrži podatke o trenutnom igraču
-------------------	---

4.11.1.16 int playerIterationHandlerDemo (*playerListNodePtr playerNode*)

Funkcija koja radi update igrača za demo igrice, sa predefinisanim vestackom inteligencijom, a poziva se u svakom frejmu

- prvi prioritet: pozicioniranje ka meti
- drugi prioritet: pozicioniranje prema powerUp-ovima
- treci prioritet: unistavanje najblizeg neprijatelja

Parametri

<i>playerNode</i>	Element liste igrača koji sadrži podatke o trenutnom igraču
-------------------	---

4.11.1.17 int powerUpIterationHandler (*powerUpListNodePtr powerUpNode*)

Osnovno hendlovanje pozicije powerupa u svakom frejmu

- powerUp se pomera ka dole brzinom pomeranja pozadine kako bi se simulirao efekat pracenja pozadine
- u slucaju da y koordinata powerUp-a bude veca od visine prozora igre, powerUp se brise

Parametri

<i>powerUpNode</i>	Element liste powerUp-ova koji sadrži podatke o trenutnom powerUp-u
--------------------	---

4.12 Opis datoteke game/savingGame.c

```
#include "savingGame.h"
```

Graf zavisnosti datoteka za savingGame.c:

Funkcije

- void [saveGame](#) (char *gamePath)
- void [loadGame](#) (char *gamePath)

4.12.1 Dokumentacija funkcije

4.12.1.1 void loadGame (char * *gamePath*)

Funkcija koja ucitava igru iz fajla

Parametri

<i>gamePath</i>	putanja do fajla iz koga treba učitati igru
-----------------	---

4.12.1.2 void saveGame (char * *gamePath*)

Funkcija koja snima igru u fajl

- u posebnom zaglavlju se čuva broj različitih objekata, kako bi datoteka kasnije bila korektno pročitana

Parametri

<code>gamePath</code>	putanja do fajla u koji treba snimiti igru
-----------------------	--

4.13 Opis datoteke game/savingGame.h

```
#include "../defines.h"
#include "game.h"
```

Graf zavisnosti datoteka za savingGame.h: Ovaj graf pokazuje koje datoteke direktno ili indirektno uključuju ovu datoteku:

Strukture i unije

- struct [saveHeaderStruct](#)

Definicije tipa

- typedef struct [saveHeaderStruct](#) * [saveHeaderPtr](#)

Funkcije

- void [saveGame](#) (char *)
- void [loadGame](#) (char *)

Promenljive

- SDL_Rect [backgroundRect](#)
Pokazivač na pravougaonik u koji se iscrtava pozadina.
- SDL_Rect [topStripRect](#)
Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.
- SDL_Rect [bottomStripRect](#)
Pokazivač na pravougaonik u koji se iscrtava donji bar.
- SDL_Renderer * [renderer](#)
Globalni pokazivač na SDL renderer.
- SDL_Window * [window](#)
Globalni pokazivač na SDL window.
- TTF_Font * [font](#)
Globalni pokazivač na SDL font.
- [gamePtr](#) [game](#)
Globalni pokazivač na trenutnu započetu ili pauziranu igru.
- int [lastCampaignLevel](#)
Indeks nivoa do koga je pređena igra.
- Mix_Music * [bgSound](#)
Zvuk motora aviona.
- Mix_Chunk * [fire](#)
Zvuk pucnja.
- Mix_Chunk * [explosionSound](#)
Zvuk eksplozije.

4.13.1 Dokumentacija definicije tipa

4.13.1.1 typedef struct saveHeaderStruct * saveHeaderPtr

4.13.2 Dokumentacija funkcije

4.13.2.1 void loadGame (char * gamePath)

Funkcija koja učitava igru iz fajla

Parametri

<i>gamePath</i>	putanja do fajla iz koga treba učitati igru
-----------------	---

4.13.2.2 void saveGame (char * gamePath)

Funkcija koja snima igru u fajl

- u posebnoj zaglavlju se čuva broj različitih objekata, kako bi datoteka kasnije bila korektno pročitana

Parametri

<i>gamePath</i>	putanja do fajla u koji treba snimiti igru
-----------------	--

4.13.3 Dokumentacija promenljive

4.13.3.1 SDL_Rect backgroundRect

Pokazivač na pravougaonik u koji se iscrtava pozadina.

4.13.3.2 Mix_Music* bgSound

Zvuk motora aviona.

4.13.3.3 SDL_Rect bottomStripRect

Pokazivač na pravougaonik u koji se iscrtava donji bar.

4.13.3.4 Mix_Chunk* explosionSound

Zvuk eksplozije.

4.13.3.5 Mix_Chunk* fire

Zvuk pucnja.

4.13.3.6 TTF_Font* font

Globalni pokazivač na SDL font.

4.13.3.7 gamePtr game

Globalni pokazivač na trenutnu započetu ili pauziranu igru.

4.13.3.8 int lastCampaignLevel

Indeks nivoa do koga je pređena igra.

4.13.3.9 SDL_Renderer* renderer

Globalni pokazivač na SDL renderer.

4.13.3.10 SDL_Rect topStripRect

Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.

4.13.3.11 SDL_Window* window

Globalni pokazivač na SDL window.

4.14 Opis datoteke instances/constructors.c

```
#include "instances.h"
```

Graf zavisnosti datoteka za constructors.c:

Funkcije

- void [initTextures](#) ()
- void [initGame](#) (int game_mode, char *s, int [difficulty](#))
- void [initText](#) (textPtr *textTexture, int x, int y, int col1, int col2, int col3, int col4, TTF_Font *font, char *str)
- void [editText](#) (textPtr *textTexture, char *str)
- void [initPlayer](#) ()
- void [initEnemy](#) (double x_pos, double y_pos, int id, int path_id)
- void [initBullet](#) (double x, double y, short isPlayerOwned, int type)
- void [initEffect](#) (double x, double y, int type)
- void [initPowerUp](#) (double x, double y, int type)
- void [initLevelMatrix](#) ()
- void [saveHighScore](#) ()

4.14.1 Dokumentacija funkcije

4.14.1.1 void editText (textPtr * textTexture, char * str)

Funkcija za izmenu sadržaja teksta i teksture strukture za prikaz teksta

Parametri

<i>textTexture</i>	Pokazivač na teksturu teksta koju treba iscrtati
<i>str</i>	Tekst koji treba iscrtati

4.14.1.2 void initBullet (double x, double y, short isPlayerOwned, int type)

Funkcija za inicijalizaciju strukture metaka i njeno dodavanje u listu metaka

- vrše se inicijalna podešavanja strukture metka, resetovanje brojača i setovanja parametara za iscrtavanje

Parametri

<i>x</i>	X koordinata mesta na koje treba postaviti metak pri inicijalizaciji
<i>y</i>	Y koordinata mesta na koje treba postaviti metak pri inicijalizaciji
<i>isPlayerOwned</i>	Flag koji govori da li je metak ispaljen od strane igrača ili neprijatelja
<i>type</i>	Identifikacioni broj tipa metka (1:običan 2:bomba)

4.14.1.3 void initEffect (double x, double y, int type)

Funkcija za inicijalizaciju strukture efekta i njeno dodavanje u listu efekata

- vrše se inicijalna podešavanja strukture efekta, resetovanje brojača i setovanja parametara za iscrtavanje

Parametri

<i>x</i>	X koordinata mesta na koje treba postaviti efekat pri inicijalizaciji
<i>y</i>	Y koordinata mesta na koje treba postaviti efekat pri inicijalizaciji
<i>type</i>	Identifikacioni broj tipa efekta (1:dim 2:eksplozija u vazduhu, 3:eksplozija na zemlji, 4:rupa na zemlji, 5:pad običnog aviona, 6:pad štuke, 7:pad bossa followera, 8:pad kamikaza, 9:pad glavnog bossa, 10:štit, 11:brzina)

4.14.1.4 void initEnemy (double x_pos, double y_pos, int id, int path_id)

Funkcija za inicijalizaciju strukture neprijatelja i njeno dodavanje u dvostruko ulančanu listu neprijatelja

- vrše se inicijalna podešavanja strukture neprijatelja, resetovanje brojača i setovanja parametara za iscrtavanje

Parametri

<i>x_pos</i>	X koordinata mesta na koje treba postaviti neprijatelja pri inicijalizaciji
<i>y_pos</i>	Y koordinata mesta na koje treba postaviti neprijatelja pri inicijalizaciji
<i>id</i>	Identifikacioni broj tipa neprijatelja (0:obicaj, 1:avion sa više health-a, 2:boss follower, 3:kamikaza, 4:glavni boss, 5:meta)
<i>path_id</i>	Identifikacioni broj putanje po kojoj neprijatelj treba da se kreće

4.14.1.5 void initGame (int game_mode, char * s, int difficulty)

Funkcija za inicijalizaciju strukture igre

- inicijalna podešavanja broja objekata i raznih brojača
- u zavisnosti od tipa igre koji se inicijalizuje otvara se određena HighScore tabela
- zove se inicijalizacija tekstura i pozadine

Parametri

<i>game_mode</i>	Indikator tipa igre koja se inicijalizuje (0:free game, 1:kampanja, 2:ucitavanje iz fajla)
<i>difficulty</i>	Indikator težine igre koja se inicijalizuje (0:demo igra, 1:lako, 2:normalno, 3:teško)

4.14.1.6 void initLevelMatrix ()

Funkcija za inicijalizaciju matrice nivoa nulama

4.14.1.7 void initPlayer ()

Funkcija za inicijalizaciju strukture igrača i njeno dodavanje u listu igrača

- struktura novog igrača se dodaje u dvostruko ulančanu listu svih igrača u okviru strukture igre
- vrše se inicijalna podešavanja brojača strukture, pozicije i veličine objekta

4.14.1.8 void initPowerUp (double x, double y, int type)

Funkcija za inicijalizaciju strukture powerUp-a i njeno dodavanje u listu efekata

- vrše se inicijalna podešavanja strukture powerUp-a, resetovanje brojača i setovanja parametara za iscrtavanje

Parametri

<i>x</i>	X koordinata mesta na koje treba postaviti powerUp pri inicijalizaciji
<i>y</i>	Y koordinata mesta na koje treba postaviti powerUp pri inicijalizaciji
<i>type</i>	Identifikacioni broj tipa powerUp-a (1:štit 2:brzina)

4.14.1.9 void initText (textPtr * textTexture, int x, int y, int col1, int col2, int col3, int col4, TTF_Font * font, char * str)

Funkcija za inicijalizaciju strukture za prikaz teksta

Parametri

<i>textTexture</i>	Pokazivač na teksturu teksta koju treba iscrtati
<i>x</i>	X koordinata mesta na koje treba iscrtati teksturu
<i>y</i>	Y koordinata mesta na koje treba iscrtati teksturu
<i>col1</i>	Jačina crvene boje teksture RGBA formata
<i>col2</i>	Jačina zelene boje teksture RGBA formata
<i>col3</i>	Jačina plave boje teksture RGBA formata
<i>col4</i>	Jačina providnosti teksture RGBA formata
<i>TTF_Font</i>	Pokazivač na font koji će biti korišćen za iscrtavanje
<i>str</i>	Tekst koji treba iscrtati

4.14.1.10 void initTextures ()

Funkcija koja inicijalizuje texture za igru

- učitava texture iz foldera sa resursima u niz tekstura koji se koristi za iscrtavanje objekata

4.14.1.11 void saveHighScore ()

Funkcija za snimanje trenutnog stanja tabele najboljih rezultata u fajl sacuvan u strukturi igre

4.15 Opis datoteke instances/destructors.c

```
#include "instances.h"
```

Graf zavisnosti datoteka za destructors.c:

Funkcije

- void `freeGame` ()
- void `freeText` (`textPtr` *`textTexture`)
- void `freePlayerNode` (`playerListNodePtr` `playerNode`)
- void `freeEnemyNode` (`enemyListNodePtr` `enemyNode`)
- void `freeEffectNode` (`effectListNodePtr` `effectNode`)
- void `freeBulletNode` (`bulletListNodePtr` `bulletNode`)
- void `freePowerUpNode` (`powerUpListNodePtr` `powerUpNode`)

4.15.1 Dokumentacija funkcije

4.15.1.1 void `freeBulletNode` (`bulletListNodePtr` `bulletNode`)

Funkcija za deinicijalizaciju strukture metka

- funkcija korektno oslobadja memoriju koju prosleđen argument zauzima i prevezuje dvostruko ulančanu listu

Parametri

<i><code>bulletNode</code></i>	Pokazivač na element liste metaka koji treba osloboditi
--------------------------------	---

4.15.1.2 void `freeEffectNode` (`effectListNodePtr` `effectNode`)

Funkcija za deinicijalizaciju strukture efekta

- funkcija korektno oslobadja memoriju koju prosleđen argument zauzima i prevezuje dvostruko ulančanu listu

Parametri

<i><code>effectNode</code></i>	Pokazivač na element liste efekata koji treba osloboditi
--------------------------------	--

4.15.1.3 void `freeEnemyNode` (`enemyListNodePtr` `enemyNode`)

Funkcija za deinicijalizaciju strukture neprijatelja

- funkcija korektno oslobadja memoriju koju prosleđen argument zauzima i prevezuje dvostruko ulančanu listu

Parametri

<i><code>enemyNode</code></i>	Pokazivač na element liste neprijatelja koji treba osloboditi
-------------------------------	---

4.15.1.4 void `freeGame` ()

Funkcija za deinicijalizaciju strukture igre

- zovu se destruktori igrača, neprijatelja, efekata i powerUp-ova
- oslobadjaju se zapamćene teksture i zatvaraju otvoreni fajlovi

4.15.1.5 void `freePlayerNode` (`playerListNodePtr` `playerNode`)

Funkcija za deinicijalizaciju strukture igrača

- funkcija korektno oslobadja memoriju koju prosleđen argument zauzima i prevezuje dvostruko ulančanu listu

Parametri

<i>playerNode</i>	Pokazivač na element liste igrača koji treba osloboditi
-------------------	---

4.15.1.6 void freePowerUpNode (powerUpListNodePtr powerUpNode)

Funkcija za deinicijalizaciju strukture powerUp-a

- funkcija korektno oslobadja memoriju koju prosleđen argument zauzima i prevezuje dvostruko ulančanu listu

Parametri

<i>powerUpNode</i>	Pokazivač na element liste powerUp-ova koji treba osloboditi
--------------------	--

4.15.1.7 void freeText (textPtr * textTexture)

Funkcija za deinicijalizaciju strukture za prikaz teksta

Parametri

<i>textTexture</i>	Pokazivač na element liste textura koji treba osloboditi
--------------------	--

4.16 Opis datoteke instances/instances.h

```
#include "../defines.h"
#include "../game/handlers.h"
#include "../game/campaign.h"
```

Graf zavisnosti datoteka za instances.h: Ovaj graf pokazuje koje datoteke direktno ili indirektno uključuju ovu datoteku:

Funkcije

- void [initTextures](#) ()
- void [initGame](#) (int, char *, int)
- void [initText](#) (textPtr *, int, int, int, int, int, int, TTF_Font *, char *)
- void [editText](#) (textPtr *, char *)
- void [initPlayer](#) ()
- void [initEnemy](#) (double, double, int, int)
- void [initEffect](#) (double, double, int)
- void [initBullet](#) (double, double, short, int)
- void [initPowerUp](#) (double, double, int)
- void [freeGame](#) ()
- void [freeText](#) (textPtr *)
- void [freePlayerNode](#) (playerListNodePtr)
- void [freeEnemyNode](#) (enemyListNodePtr)
- void [freeEffectNode](#) (effectListNodePtr)
- void [freeBulletNode](#) (bulletListNodePtr)
- void [freePowerUpNode](#) (powerUpListNodePtr)
- void [initLoad_curr_lvl](#) ()
- void [initLevelMatrix](#) ()
- void [saveHighScore](#) ()

Promenljive

- SDL_Rect [backgroundRect](#)
Pokazivač na pravougaonik u koji se iscrtava pozadina.
- SDL_Rect [topStripRect](#)
Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.
- SDL_Rect [bottomStripRect](#)
Pokazivač na pravougaonik u koji se iscrtava donji bar.
- SDL_Renderer * [renderer](#)
Globalni pokazivač na SDL renderer.
- SDL_Window * [window](#)
Globalni pokazivač na SDL window.
- TTF_Font * [font](#)
Globalni pokazivač na SDL font.
- [gamePtr](#) [game](#)
Globalni pokazivač na trenutnu započetu ili pauziranu igru.
- int [lastCampaignLevel](#)
Indeks nivoa do koga je pređena igra.
- Mix_Music * [bgSound](#)
Zvuk motora aviona.
- Mix_Chunk * [fire](#)
Zvuk pucnja.
- Mix_Chunk * [explosionSound](#)
Zvuk eksplozije.

4.16.1 Dokumentacija funkcije

4.16.1.1 void editText (textPtr * textTexture, char * str)

Funkcija za izmenu sadržaja teksta i teksture strukture za prikaz teksta

Parametri

<i>textTexture</i>	Pokazivač na teksturu teksta koju treba iscrtati
<i>str</i>	Tekst koji treba iscrtati

4.16.1.2 void freeBulletNode (bulletListNodePtr bulletNode)

Funkcija za deinicijalizaciju strukture metka

- funkcija korektno oslobadja memoriju koju prosleđen argument zauzima i prevezuje dvostruko ulančanu listu

Parametri

<i>bulletNode</i>	Pokazivač na element liste metaka koji treba osloboditi
-------------------	---

4.16.1.3 void freeEffectNode (effectListNodePtr effectNode)

Funkcija za deinicijalizaciju strukture efekta

- funkcija korektno oslobadja memoriju koju prosleđen argument zauzima i prevezuje dvostruko ulančanu listu

Parametri

<i>effectNode</i>	Pokazivač na element liste efekata koji treba osloboditi
-------------------	--

4.16.1.4 void freeEnemyNode (enemyListNodePtr enemyNode)

Funkcija za deinicijalizaciju strukture neprijatelja

- funkcija korektno oslobadja memoriju koju prosleđen argument zauzima i prevezuje dvostruko ulančanu listu

Parametri

<i>enemyNode</i>	Pokazivač na element liste neprijatelja koji treba osloboditi
------------------	---

4.16.1.5 void freeGame ()

Funkcija za deinicijalizaciju strukture igre

- zovu se destruktori igrača, neprijatelja, efekata i powerUp-ova
- oslobadjaju se zapamćene teksture i zatvaraju otvoreni fajlovi

4.16.1.6 void freePlayerNode (playerListNodePtr playerNode)

Funkcija za deinicijalizaciju strukture igrača

- funkcija korektno oslobadja memoriju koju prosleđen argument zauzima i prevezuje dvostruko ulančanu listu

Parametri

<i>playerNode</i>	Pokazivač na element liste igrača koji treba osloboditi
-------------------	---

4.16.1.7 void freePowerUpNode (powerUpListNodePtr powerUpNode)

Funkcija za deinicijalizaciju strukture powerUp-a

- funkcija korektno oslobadja memoriju koju prosleđen argument zauzima i prevezuje dvostruko ulančanu listu

Parametri

<i>powerUpNode</i>	Pokazivač na element liste powerUp-ova koji treba osloboditi
--------------------	--

4.16.1.8 void freeText (textPtr * textTexture)

Funkcija za deinicijalizaciju strukture za prikaz teksta

Parametri

<i>textTexture</i>	Pokazivač na element liste textura koji treba osloboditi
--------------------	--

4.16.1.9 void initBullet (double x, double y, short isPlayerOwned, int type)

Funkcija za inicijalizaciju strukture metaka i njeno dodavanje u listu metaka

- vrše se inicijalna podešavanja strukture metka, resetovanje brojača i setovanja parametara za iscrtavanje

Parametri

<i>x</i>	X koordinata mesta na koje treba postaviti metak pri inicijalizaciji
<i>y</i>	Y koordinata mesta na koje treba postaviti metak pri inicijalizaciji
<i>isPlayerOwned</i>	Flag koji govori da li je metak ispaljen od strane igrača ili neprijatelja
<i>type</i>	Identifikacioni broj tipa metka (1:običan 2:bomba)

4.16.1.10 void initEffect (double x, double y, int type)

Funkcija za inicijalizaciju strukture efekta i njeno dodavanje u listu efekata

- vrše se inicijalna podešavanja strukture efekta, resetovanje brojača i setovanja parametara za iscrtavanje

Parametri

<i>x</i>	X koordinata mesta na koje treba postaviti efekat pri inicijalizaciji
<i>y</i>	Y koordinata mesta na koje treba postaviti efekat pri inicijalizaciji
<i>type</i>	Identifikacioni broj tipa efekta (1:dim 2:eksplozija u vazduhu, 3:eksplozija na zemlji, 4:rupa na zemlji, 5:pad običnog aviona, 6:pad štuke, 7:pad bossa followera, 8:pad kamikaza, 9:pad glavnog bossa, 10:štit, 11:brzina)

4.16.1.11 void initEnemy (double x_pos, double y_pos, int id, int path_id)

Funkcija za inicijalizaciju strukture neprijatelja i njeno dodavanje u dvostruko ulančanu listu neprijatelja

- vrše se inicijalna podešavanja strukture neprijatelja, resetovanje brojača i setovanja parametara za iscrtavanje

Parametri

<i>x_pos</i>	X koordinata mesta na koje treba postaviti neprijatelja pri inicijalizaciji
<i>y_pos</i>	Y koordinata mesta na koje treba postaviti neprijatelja pri inicijalizaciji
<i>id</i>	Identifikacioni broj tipa neprijatelja (0:obicaj, 1:avion sa više health-a, 2:boss follower, 3: kamikaza, 4:glavni boss, 5:meta)
<i>path_id</i>	Identifikacioni broj putanje po kojoj neprijatelj treba da se kreće

4.16.1.12 void initGame (int game_mode, char * s, int difficulty)

Funkcija za inicijalizaciju strukture igre

- inicijalna podešavanja broja objekata i raznih brojača
- u zavisnosti od tipa igre koji se inicijalizuje otvara se određena HighScore tabela
- zove se inicijalizacija tekstura i pozadine

Parametri

<i>game_mode</i>	Indikator tipa igre koja se inicijalizuje (0:free game, 1:kampanja, 2:ucitavanje iz fajla)
<i>difficulty</i>	Indikator težine igre koja se inicijalizuje (0:demo igra, 1:lako, 2:normalno, 3:teško)

4.16.1.13 void initLevelMatrix ()

Funkcija za inicijalizaciju matrice nivoa nulama

4.16.1.14 void initLoad_curr_lvl ()

4.16.1.15 void initPlayer ()

Funkcija za inicijalizaciju strukture igrača i njeno dodavanje u listu igrača

- struktura novog igrača se dodaje u dvostruko ulančanu listu svih igrača u okviru strukture igre
- vrše se inicijalna podešavanja brojača strukture, pozicije i veličine objekta

4.16.1.16 void initPowerUp (double x, double y, int type)

Funkcija za inicijalizaciju strukture powerUp-a i njeno dodavanje u listu efekata

- vrše se inicijalna podešavanja strukture powerUp-a, resetovanje brojača i setovanja parametara za iscrtavanje

Parametri

<i>x</i>	X koordinata mesta na koje treba postaviti powerUp pri inicijalizaciji
<i>y</i>	Y koordinata mesta na koje treba postaviti powerUp pri inicijalizaciji
<i>type</i>	Identifikacioni broj tipa powerUp-a (1:štit 2:brzina)

4.16.1.17 void initText (textPtr * textTexture, int x, int y, int col1, int col2, int col3, int col4, TTF_Font * font, char * str)

Funkcija za inicijalizaciju strukture za prikaz teksta

Parametri

<i>textTexture</i>	Pokazivač na teksturu teksta koju treba iscrtati
<i>x</i>	X koordinata mesta na koje treba iscrtati teksturu
<i>y</i>	Y koordinata mesta na koje treba iscrtati teksturu
<i>col1</i>	Jačina crvene boje teksture RGBA formata
<i>col2</i>	Jačina zelene boje teksture RGBA formata
<i>col3</i>	Jačina plave boje teksture RGBA formata
<i>col4</i>	Jačina providnosti teksture RGBA formata
<i>TTF_Font</i>	Pokazivač na font koji će biti korišćen za iscrtavanje
<i>str</i>	Tekst koji treba iscrtati

4.16.1.18 void initTextures ()

Funkcija koja inicijalizuje teksture za igru

- učitava teksture iz foldera sa resursima u niz tekstura koji se koristi za iscrtavanje objekata

4.16.1.19 void saveHighScore ()

Funkcija za snimanje trenutnog stanja tabele najboljih rezultata u fajl sacuvan u strukturi igre

4.16.2 Dokumentacija promenljive

4.16.2.1 SDL_Rect backgroundRect

Pokazivač na pravougaonik u koji se iscrtava pozadina.

4.16.2.2 Mix_Music* bgSound

Zvuk motora aviona.

4.16.2.3 SDL_Rect bottomStripRect

Pokazivač na pravougaonik u koji se iscrtava donji bar.

4.16.2.4 Mix_Chunk* explosionSound

Zvuk eksplozije.

4.16.2.5 Mix_Chunk* fire

Zvuk pucnja.

4.16.2.6 TTF_Font* font

Globalni pokazivač na SDL font.

4.16.2.7 gamePtr game

Globalni pokazivač na trenutnu započetu ili pauziranu igru.

4.16.2.8 int lastCampaignLevel

Indeks nivoa do koga je pređena igra.

4.16.2.9 SDL_Renderer* renderer

Globalni pokazivač na SDL renderer.

4.16.2.10 SDL_Rect topStripRect

Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.

4.16.2.11 SDL_Window* window

Globalni pokazivač na SDL window.

4.17 Opis datoteke main.c

```
#include "main.h"
```

Graf zavisnosti datoteka za main.c:

Funkcije

- void `initProgram` ()
- void `killGame` ()
- int `main` (int argc, char *argv[])

4.17.1 Dokumentacija funkcije

4.17.1.1 void initProgram ()

Funkcija za inicijalizaciju igre

4.17.1.2 void killGame ()

Funkcija za deinicijalizaciju igre

4.17.1.3 int main (int argc, char * argv[])

Glavna funkcija koja vrti meni i igru

4.18 Opis datoteke main.h

```
#include "defines.h"
#include "instances/instances.h"
#include "game/handlers.h"
#include "game/collisions.h"
#include "game/game.h"
#include "menu/menu.h"
```

Graf zavisnosti datoteka za main.h: Ovaj graf pokazuje koje datoteke direktno ili indirektno uključuju ovu datoteku:

Promenljive

- SDL_Rect `backgroundRect`
Pokazivač na pravougaonik u koji se iscrtava pozadina.
- SDL_Rect `topStripRect`
Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.
- SDL_Rect `bottomStripRect`
Pokazivač na pravougaonik u koji se iscrtava donji bar.
- SDL_Renderer * `renderer`
Globalni pokazivač na SDL renderer.
- SDL_Window * `window`
Globalni pokazivač na SDL window.
- TTF_Font * `font`
Globalni pokazivač na SDL font.
- `gamePtr` `game`
Globalni pokazivač na trenutnu započetu ili pauziranu igru.
- int `lastCampaignLevel`
Indeks nivoa do koga je pređena igra.
- Mix_Music * `bgSound`
Zvuk motora aviona.

- Mix_Chunk * [fire](#)
Zvuk pucnja.
- Mix_Chunk * [explosionSound](#)
Zvuk eksplozije.

4.18.1 Dokumentacija promenljive

4.18.1.1 SDL_Rect backgroundRect

Pokazivač na pravougaonik u koji se iscrtava pozadina.

4.18.1.2 Mix_Music* bgSound

Zvuk motora aviona.

4.18.1.3 SDL_Rect bottomStripRect

Pokazivač na pravougaonik u koji se iscrtava donji bar.

4.18.1.4 Mix_Chunk* explosionSound

Zvuk eksplozije.

4.18.1.5 Mix_Chunk* fire

Zvuk pucnja.

4.18.1.6 TTF_Font* font

Globalni pokazivač na SDL font.

4.18.1.7 gamePtr game

Globalni pokazivač na trenutnu započetu ili pauziranu igru.

4.18.1.8 int lastCampaignLevel

Indeks nivoa do koga je pređena igra.

4.18.1.9 SDL_Renderer* renderer

Globalni pokazivač na SDL renderer.

4.18.1.10 SDL_Rect topStripRect

Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.

4.18.1.11 SDL_Window* window

Globalni pokazivač na SDL window.

4.19 Opis datoteke menu/menu.c

```
#include "menu.h"
```

Graf zavisnosti datoteka za menu.c:

Funkcije

- int `selectCampaignLevel` ()
- void `getLevelFileName` (char *fileName)
- void `getSavedGameName` (char *fileName)
- int `showMenu` ()

4.19.1 Dokumentacija funkcije

4.19.1.1 void `getLevelFileName` (char * *fileName*)

Funkcija koja omogućava unos imena dodatnog nivoa

- učitavanje imena svih fajlova sa opisima nivoa u niz
- korišćenjem dugmeta TAB moguće je iterirati kroz učitani niz

Parametri

<i>fileName</i>	Niz karaktera u koji se upisuje uneto ime
-----------------	---

4.19.1.2 void `getSavedGameName` (char * *fileName*)

Funkcija koja omogućava unos imena snimljene igre

- učitavanje imena svih fajlova sa opisima nivoa u niz
- korišćenjem dugmeta TAB moguće je iterirati kroz učitani niz

Parametri

<i>fileName</i>	Niz karaktera u koji se upisuje uneto ime
-----------------	---

4.19.1.3 int `selectCampaignLevel` ()

Funkcija koja prikazuje podmeni za odabir željenog dela kampanje

- na početku funkcije se uključuje prikazivanje podmenija
- omogućava odabiranje željene opcije podešavanjem selektoora
- pri izlasku prekida ispisivanje podmenija

4.19.1.4 int showMenu ()

Funkcija sa glavnom petljom menija

- prepoznaje događaje i ažurira niz events
- poziva funkciju handleMenuEvents koja upravlja menijem
- vraća 1 ukoliko je potrebno pokrenuti igru, ili 0 ukoliko je potrebno izaći iz prorama

4.20 Opis datoteke menu/menu.h

```
#include "../defines.h"
#include "../draw.h"
#include "../instances/instances.h"
#include "../game/savingGame.h"
```

Graf zavisnosti datoteka za menu.h: Ovaj graf pokazuje koje datoteke direktno ili indirektno uključuju ovu datoteku:

Makro zamene

- #define TEXT_TEXTURE_NUM 24

Funkcije

- int showMenu ()

Promenljive

- SDL_Rect backgroundRect
Pokazivač na pravougaonik u koji se iscrtava pozadina.
- SDL_Rect topStripRect
Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.
- SDL_Rect bottomStripRect
Pokazivač na pravougaonik u koji se iscrtava donji bar.
- SDL_Renderer * renderer
Globalni pokazivač na SDL renderer.
- SDL_Window * window
Globalni pokazivač na SDL window.
- TTF_Font * font
Globalni pokazivač na SDL font.
- gamePtr game
Globalni pokazivač na trenutnu započetu ili pauziranu igru.
- int lastCampaignLevel
Indeks nivoa do koga je pređena igra.
- Mix_Music * bgSound
Zvuk motora aviona.
- Mix_Chunk * fire
Zvuk pucnja.
- Mix_Chunk * explosionSound
Zvuk eksplozije.
- SDL_Texture * backgroundTexture

- Tekstura pozadine u meniju.*
- SDL_Texture * [selectorTexture](#)
- Tekstura selektora u meniju.*
- SDL_Texture * [metalFrameTexture](#)
- Tekstura metalnog okvira.*
- SDL_Texture * [helpTexture](#)
- Tekstura koja sadrži sliku sa savetima.*
- SDL_Texture * [logoTexture](#)
- Tekstura koja sadrži logo igre.*
- SDL_Texture * [backgroundOverlay](#)
- Tekstura koja služi da zatamni pozadinu.*
- SDL_Rect [selectorRect](#)
- SDL_Rect [selector2Rect](#)
- SDL_Rect [metalFrameRect](#)
- SDL_Rect [logoRect](#)
- SDL_Rect [mainMetalFrameRect](#)
- SDL_Rect [helpRect](#)
- SDL_Rect [helpTipRect](#)
- int [selectorPos](#)
- int [selector2Pos](#)
- int [difficulty](#)
- Mix_Music * [menuMusic](#)
- Muzika u glavnom meniju.*
- [textPtr](#) [textTextures](#) [[TEXT_TEXTURE_NUM](#)]
- Niz tekstova za ispis na ekranu.*
- char [events](#) [256]
- Niz zabeleženih događaja.*

4.20.1 Dokumentacija makro zamene

4.20.1.1 #define TEXT_TEXTURE_NUM 24

4.20.2 Dokumentacija funkcije

4.20.2.1 int showMenu ()

Funkcija sa glavnom petljom menija

- prepoznaje događaje i ažurira niz events
- poziva funkciju handleMenuEvents koja upravlja menijem
- vraća 1 ukoliko je potrebno pokrenuti igru, ili 0 ukoliko je potrebno izaći iz prorama

4.20.3 Dokumentacija promenljive

4.20.3.1 SDL_Texture* backgroundOverlay

Tekstura koja služi da zatamni pozadinu.

4.20.3.2 SDL_Rect backgroundRect

Pokazivač na pravougaonik u koji se iscrtava pozadina.

4.20.3.3 SDL_Texture* backgroundTexture

Tekstura pozadine u meniju.

4.20.3.4 Mix_Music* bgSound

Zvuk motora aviona.

4.20.3.5 SDL_Rect bottomStripRect

Pokazivač na pravougaonik u koji se iscrtava donji bar.

4.20.3.6 int difficulty**4.20.3.7 char events[256]**

Niz zabeleženih događaja.

4.20.3.8 Mix_Chunk* explosionSound

Zvuk eksplozije.

4.20.3.9 Mix_Chunk* fire

Zvuk pucnja.

4.20.3.10 TTF_Font* font

Globalni pokazivač na SDL font.

4.20.3.11 gamePtr game

Globalni pokazivač na trenutnu započetu ili pauziranu igru.

4.20.3.12 SDL_Rect helpRect**4.20.3.13 SDL_Texture* helpTexture**

Tekstura koja sadrži sliku sa savetima.

4.20.3.14 SDL_Rect helpTipRect**4.20.3.15 int lastCampaignLevel**

Indeks nivoa do koga je pređena igra.

4.20.3.16 SDL_Rect logoRect**4.20.3.17 SDL_Texture* logoTexture**

Tekstura koja sadrži logo igre.

4.20.3.18 `SDL_Rect mainMetalFrameRect`

4.20.3.19 `Mix_Music* menuMusic`

Muzika u glavnom meniju.

4.20.3.20 `SDL_Rect metalFrameRect`

4.20.3.21 `SDL_Texture* metalFrameTexture`

Tekstura metalnog okvira.

4.20.3.22 `SDL_Renderer* renderer`

Globalni pokazivač na SDL renderer.

4.20.3.23 `int selector2Pos`

4.20.3.24 `SDL_Rect selector2Rect`

4.20.3.25 `int selectorPos`

4.20.3.26 `SDL_Rect selectorRect`

4.20.3.27 `SDL_Texture* selectorTexture`

Tekstura selektora u meniju.

4.20.3.28 `textPtr textTextures[TEXT_TEXTURE_NUM]`

Niz tekstova za ispis na ekranu.

4.20.3.29 `SDL_Rect topStripRect`

Pokazivač na pravougaonik u koji se iscrtava gornji bar u igri.

4.20.3.30 `SDL_Window* window`

Globalni pokazivač na SDL window.

Indeks

- background_campaign
 - game.c, [33](#)
- background_load_row
 - game.c, [33](#)
- background_offset
 - gameStruct, [13](#)
- background_random
 - game.c, [33](#)
- background_swap
 - game.c, [33](#)
- backgroundOverlay
 - menu.h, [65](#)
- backgroundRect
 - collisions.h, [32](#)
 - draw.h, [27](#)
 - game.h, [36](#)
 - handlers.h, [43](#)
 - instances.h, [59](#)
 - main.h, [62](#)
 - menu.h, [65](#)
 - savingGame.h, [50](#)
- backgroundTexture
 - game.h, [36](#)
 - menu.h, [65](#)
- backgroundTiles
 - gameStruct, [13](#)
- bgSound
 - collisions.h, [32](#)
 - draw.h, [27](#)
 - game.h, [36](#)
 - handlers.h, [43](#)
 - instances.h, [59](#)
 - main.h, [62](#)
 - menu.h, [66](#)
 - savingGame.h, [50](#)
- bomb_texture
 - gameStruct, [13](#)
- bombCoolDown
 - playerStruct, [19](#)
- boss_active
 - gameStruct, [13](#)
- bottomStripRect
 - collisions.h, [32](#)
 - draw.h, [28](#)
 - game.h, [36](#)
 - handlers.h, [43](#)
 - instances.h, [60](#)
 - main.h, [62](#)
 - menu.h, [66](#)
- savingGame.h, [50](#)
- bullet
 - bulletListNodeStruct, [5](#)
- bullet_texture
 - gameStruct, [13](#)
- bulletCoolDown
 - playerStruct, [19](#)
- bulletIterationHandlerBomb
 - handlers.h, [39](#)
 - iterationHandlers.c, [45](#)
- bulletIterationHandlerBullet
 - handlers.h, [39](#)
 - iterationHandlers.c, [45](#)
- bulletListNodeStruct, [5](#)
 - bullet, [5](#)
 - next, [5](#)
 - prev, [5](#)
- bulletNum
 - saveHeaderStruct, [22](#)
- bulletStruct, [6](#)
 - counter, [6](#)
 - handleIteration, [6](#)
 - id, [6](#)
 - isPlayerOwned, [6](#)
 - position, [6](#)
 - texturePosition, [6](#)
 - x_pos, [7](#)
 - y_pos, [7](#)
- bullets
 - gameStruct, [14](#)
- bulletsCount
 - gameStruct, [14](#)
- campaign.c
 - campaign_background, [29](#)
- campaign.h
 - campaign_background, [29](#)
- campaign_background
 - campaign.c, [29](#)
 - campaign.h, [29](#)
- campaign_level_ind
 - gameStruct, [14](#)
- campaign_row
 - gameStruct, [14](#)
- collisions.c
 - detectBulletEnemyCollision, [29](#)
 - detectBulletPlayerCollision, [29](#)
 - detectCollisions, [30](#)
 - detectPlayerEnemyCollision, [30](#)
 - detectPowerUpEnemyCollision, [30](#)

- detectPowerUpPlayerCollision, 30
- MIN, 29
- collisions.h
 - backgroundRect, 32
 - bgSound, 32
 - bottomStripRect, 32
 - detectBulletEnemyCollision, 31
 - detectBulletPlayerCollision, 31
 - detectCollisions, 31
 - detectPlayerEnemyCollision, 31
 - detectPowerUpEnemyCollision, 31
 - detectPowerUpPlayerCollision, 31
 - explosionSound, 32
 - fire, 32
 - font, 32
 - game, 32
 - lastCampaignLevel, 32
 - renderer, 32
 - topStripRect, 32
 - window, 32
- color
 - textStruct, 23
- constructors.c
 - editText, 51
 - initBullet, 51
 - initEffect, 52
 - initEnemy, 52
 - initGame, 52
 - initLevelMatrix, 52
 - initPlayer, 52
 - initPowerUp, 53
 - initText, 53
 - initTextures, 53
 - saveHighScore, 53
- coolDown
 - enemyStruct, 10
- counter
 - bulletStruct, 6
 - effectStruct, 8
 - enemyStruct, 10
 - playerStruct, 19
 - powerUpStruct, 22
- defines.h, 25
- destructors.c
 - freeBulletNode, 54
 - freeEffectNode, 54
 - freeEnemyNode, 54
 - freeGame, 54
 - freePlayerNode, 54
 - freePowerUpNode, 55
 - freeText, 55
- detectBulletEnemyCollision
 - collisions.c, 29
 - collisions.h, 31
- detectBulletPlayerCollision
 - collisions.c, 29
 - collisions.h, 31
- detectCollisions
 - collisions.c, 30
 - collisions.h, 31
- detectPlayerEnemyCollision
 - collisions.c, 30
 - collisions.h, 31
- detectPowerUpEnemyCollision
 - collisions.c, 30
 - collisions.h, 31
- detectPowerUpPlayerCollision
 - collisions.c, 30
 - collisions.h, 31
- difficulty
 - gameStruct, 14
 - menu.h, 66
- doBackgroundJob
 - game.c, 33
- draw.c, 25
 - drawBackground, 25
 - drawBullets, 25
 - drawEffects, 26
 - drawEnemies, 26
 - drawPlayers, 26
 - drawPowerUps, 26
 - drawText, 26
- draw.h, 26
 - backgroundRect, 27
 - bgSound, 27
 - bottomStripRect, 28
 - drawBackground, 27
 - drawBullets, 27
 - drawEffects, 27
 - drawEnemies, 27
 - drawPlayers, 27
 - drawPowerUps, 27
 - drawText, 27
 - explosionSound, 28
 - fire, 28
 - font, 28
 - game, 28
 - lastCampaignLevel, 28
 - renderer, 28
 - topStripRect, 28
 - window, 28
- drawBackground
 - draw.c, 25
 - draw.h, 27
- drawBullets
 - draw.c, 25
 - draw.h, 27
- drawEffects
 - draw.c, 26
 - draw.h, 27
- drawEnemies
 - draw.c, 26
 - draw.h, 27
- drawPlayers
 - draw.c, 26
 - draw.h, 27

- drawPowerUps
 - draw.c, 26
 - draw.h, 27
- drawText
 - draw.c, 26
 - draw.h, 27
- editText
 - constructors.c, 51
 - instances.h, 56
- effect
 - effectListNodeStruct, 7
- effectIterationHandlerAirExplosion
 - handlers.h, 39
 - iterationHandlers.c, 45
- effectIterationHandlerExhaust
 - handlers.h, 39
 - iterationHandlers.c, 45
- effectIterationHandlerFallingPlane
 - handlers.h, 39
 - iterationHandlers.c, 45
- effectIterationHandlerHole
 - handlers.h, 41
 - iterationHandlers.c, 46
- effectIterationHandlerLandExplosion
 - handlers.h, 41
 - iterationHandlers.c, 46
- effectIterationHandlerPowerUp
 - handlers.h, 41
 - iterationHandlers.c, 46
- effectListNodeStruct, 7
 - effect, 7
 - next, 7
 - prev, 7
- effectNum
 - saveHeaderStruct, 22
- effectStruct, 8
 - counter, 8
 - handleIteration, 8
 - id, 8
 - position, 8
 - x_pos, 8
 - y_pos, 8
- effects
 - gameStruct, 14
- effectsCount
 - gameStruct, 14
- enemies
 - gameStruct, 14
- enemiesCount
 - gameStruct, 14
- enemy
 - enemyListNodeStruct, 9
- enemy_sprites
 - gameStruct, 14
- enemyIterationHandlerBoss
 - handlers.h, 41
 - iterationHandlers.c, 46
- enemyIterationHandlerDefault
 - handlers.h, 41
 - iterationHandlers.c, 46
- enemyIterationHandlerFollower
 - handlers.h, 42
 - iterationHandlers.c, 47
- enemyIterationHandlerGroundTarget
 - handlers.h, 42
 - iterationHandlers.c, 47
- enemyIterationHandlerKamikaze
 - handlers.h, 42
 - iterationHandlers.c, 47
- enemyListNodeStruct, 9
 - enemy, 9
 - next, 9
 - prev, 9
- enemyNum
 - saveHeaderStruct, 22
- enemySpawnCounter
 - gameStruct, 14
- enemyStruct, 9
 - coolDown, 10
 - counter, 10
 - handleIteration, 10
 - health, 10
 - id, 10
 - path_id, 11
 - position, 11
 - score, 11
 - texture_slot, 11
 - texturePosition, 11
 - x_pos, 11
 - y_pos, 11
- events
 - game.h, 36
 - menu.h, 66
- explosion_texture
 - gameStruct, 14
- explosionSound
 - collisions.h, 32
 - draw.h, 28
 - game.h, 36
 - handlers.h, 43
 - instances.h, 60
 - main.h, 62
 - menu.h, 66
 - savingGame.h, 50
- final_boss_active
 - gameStruct, 15
- fire
 - collisions.h, 32
 - draw.h, 28
 - game.h, 36
 - handlers.h, 43
 - instances.h, 60
 - main.h, 62
 - menu.h, 66
 - savingGame.h, 50
- font

- collisions.h, 32
- draw.h, 28
- game.h, 37
- handlers.h, 43
- instances.h, 60
- main.h, 62
- menu.h, 66
- savingGame.h, 50
- textStruct, 23
- frames
 - gameStruct, 15
- framesFromCont
 - gameStruct, 15
- freeBulletNode
 - destructors.c, 54
 - instances.h, 56
- freeEffectNode
 - destructors.c, 54
 - instances.h, 56
- freeEnemyNode
 - destructors.c, 54
 - instances.h, 57
- freeGame
 - destructors.c, 54
 - instances.h, 57
- freePlayerNode
 - destructors.c, 54
 - instances.h, 57
- freePowerUpNode
 - destructors.c, 55
 - instances.h, 57
- freeText
 - destructors.c, 55
 - instances.h, 57
- fumes_texture
 - gameStruct, 15
- game
 - collisions.h, 32
 - draw.h, 28
 - game.h, 37
 - handlers.h, 44
 - instances.h, 60
 - main.h, 62
 - menu.h, 66
 - savingGame.h, 50
- game.c
 - background_campaign, 33
 - background_load_row, 33
 - background_random, 33
 - background_swap, 33
 - doBackgroundJob, 33
 - renderGame, 33
 - runGame, 33
 - swap_tile_rows, 34
 - updateEffects, 34
- game.h
 - backgroundRect, 36
 - backgroundTexture, 36
 - bgSound, 36
 - bottomStripRect, 36
 - events, 36
 - explosionSound, 36
 - fire, 36
 - font, 37
 - game, 37
 - gameOverRect, 37
 - gameOverTexture, 37
 - HSBackgroundRect, 37
 - HSBackgroundTexture, 37
 - HSFileMemberPtr, 35
 - lastCampaignLevel, 37
 - levelClearTexture, 37
 - maxLevelSavePtr, 35
 - PLAYERS_NUM, 35
 - players, 37
 - renderGame, 36
 - renderer, 37
 - runGame, 36
 - TEXTS_NUM, 35
 - texts, 37
 - topStripRect, 37
 - window, 37
- game/campaign.c, 28
- game/campaign.h, 29
- game/collisions.c, 29
- game/collisions.h, 30
- game/game.c, 33
- game/game.h, 34
- game/handlers.h, 38
- game/iterationHandlers.c, 44
- game/savingGame.c, 48
- game/savingGame.h, 49
- game_mode
 - gameStruct, 15
- gameIterationHandlerDefault
 - handlers.h, 42
 - iterationHandlers.c, 47
- gameOverRect
 - game.h, 37
 - handlers.h, 44
- gameOverTexture
 - game.h, 37
- gameStruct, 11
 - background_offset, 13
 - backgroundTiles, 13
 - bomb_texture, 13
 - boss_active, 13
 - bullet_texture, 13
 - bullets, 14
 - bulletsCount, 14
 - campaign_level_ind, 14
 - campaign_row, 14
 - difficulty, 14
 - effects, 14
 - effectsCount, 14
 - enemies, 14

- enemiesCount, 14
- enemy_sprites, 14
- enemySpawnCounter, 14
- explosion_texture, 14
- final_boss_active, 15
- frames, 15
- framesFromCont, 15
- fumes_texture, 15
- game_mode, 15
- HSFileName, 15
- HSNames, 15
- HSScores, 15
- handleIteration, 15
- holeTexture, 15
- level_default, 15
- level_file, 15
- level_matrix, 16
- pauseSurface, 16
- pauseTexture, 16
- playerTexture, 16
- playerTextureLeft, 16
- playerTextureRight, 16
- players, 16
- playersCount, 16
- powerUpTexture, 16
- powerUps, 16
- score, 16
- shieldTexture, 16
- speedTexture, 17
- getLevelFileName
 - menu.c, 63
- getSavedGameName
 - menu.c, 63
- HSBackgroundRect
 - game.h, 37
 - handlers.h, 44
- HSBackgroundTexture
 - game.h, 37
- HSFileMemberPtr
 - game.h, 35
- HSFileMemberStruct, 17
 - name, 17
 - score, 17
- HSFileName
 - gameStruct, 15
- HSNames
 - gameStruct, 15
- HSScores
 - gameStruct, 15
- handleIteration
 - bulletStruct, 6
 - effectStruct, 8
 - enemyStruct, 10
 - gameStruct, 15
 - playerStruct, 19
 - powerUpStruct, 22
- handlers.h
 - backgroundRect, 43
 - bgSound, 43
 - bottomStripRect, 43
 - bulletIterationHandlerBomb, 39
 - bulletIterationHandlerBullet, 39
 - effectIterationHandlerAirExplosion, 39
 - effectIterationHandlerExhaust, 39
 - effectIterationHandlerFallingPlane, 39
 - effectIterationHandlerHole, 41
 - effectIterationHandlerLandExplosion, 41
 - effectIterationHandlerPowerUp, 41
 - enemyIterationHandlerBoss, 41
 - enemyIterationHandlerDefault, 41
 - enemyIterationHandlerFollower, 42
 - enemyIterationHandlerGroundTarget, 42
 - enemyIterationHandlerKamikaze, 42
 - explosionSound, 43
 - fire, 43
 - font, 43
 - game, 44
 - gameIterationHandlerDefault, 42
 - gameOverRect, 44
 - HSBackgroundRect, 44
 - lastCampaignLevel, 44
 - playerIterationHandlerDefault, 42
 - playerIterationHandlerDemo, 43
 - powerUpIterationHandler, 43
 - renderer, 44
 - topStripRect, 44
 - window, 44
- health
 - enemyStruct, 10
 - playerStruct, 20
- helpRect
 - menu.h, 66
- helpTexture
 - menu.h, 66
- helpTipRect
 - menu.h, 66
- holeTexture
 - gameStruct, 15
- id
 - bulletStruct, 6
 - effectStruct, 8
 - enemyStruct, 10
 - powerUpStruct, 22
- initBullet
 - constructors.c, 51
 - instances.h, 57
- initEffect
 - constructors.c, 52
 - instances.h, 58
- initEnemy
 - constructors.c, 52
 - instances.h, 58
- initGame
 - constructors.c, 52
 - instances.h, 58
- initLevelMatrix

- constructors.c, 52
- instances.h, 58
- initLoad_curr_lvl
 - instances.h, 58
- initPlayer
 - constructors.c, 52
 - instances.h, 59
- initPowerUp
 - constructors.c, 53
 - instances.h, 59
- initProgram
 - main.c, 61
- initText
 - constructors.c, 53
 - instances.h, 59
- initTextures
 - constructors.c, 53
 - instances.h, 59
- instances.h
 - backgroundRect, 59
 - bgSound, 59
 - bottomStripRect, 60
 - editText, 56
 - explosionSound, 60
 - fire, 60
 - font, 60
 - freeBulletNode, 56
 - freeEffectNode, 56
 - freeEnemyNode, 57
 - freeGame, 57
 - freePlayerNode, 57
 - freePowerUpNode, 57
 - freeText, 57
 - game, 60
 - initBullet, 57
 - initEffect, 58
 - initEnemy, 58
 - initGame, 58
 - initLevelMatrix, 58
 - initLoad_curr_lvl, 58
 - initPlayer, 59
 - initPowerUp, 59
 - initText, 59
 - initTextures, 59
 - lastCampaignLevel, 60
 - renderer, 60
 - saveHighScore, 59
 - topStripRect, 60
 - window, 60
- instances/constructors.c, 51
- instances/destructors.c, 53
- instances/instances.h, 55
- isPlayerOwned
 - bulletStruct, 6
- iterationHandlers.c
 - bulletIterationHandlerBomb, 45
 - bulletIterationHandlerBullet, 45
 - effectIterationHandlerAirExplosion, 45
 - effectIterationHandlerExhaust, 45
 - effectIterationHandlerFallingPlane, 45
 - effectIterationHandlerHole, 46
 - effectIterationHandlerLandExplosion, 46
 - effectIterationHandlerPowerUp, 46
 - enemyIterationHandlerBoss, 46
 - enemyIterationHandlerDefault, 46
 - enemyIterationHandlerFollower, 47
 - enemyIterationHandlerGroundTarget, 47
 - enemyIterationHandlerKamikaze, 47
 - gameIterationHandlerDefault, 47
 - playerIterationHandlerDefault, 47
 - playerIterationHandlerDemo, 48
 - powerUpIterationHandler, 48
- killGame
 - main.c, 61
- lastCampaignLevel
 - collisions.h, 32
 - draw.h, 28
 - game.h, 37
 - handlers.h, 44
 - instances.h, 60
 - main.h, 62
 - menu.h, 66
 - savingGame.h, 50
- level
 - maxLevelSaveStruct, 18
- level_default
 - gameStruct, 15
- level_file
 - gameStruct, 15
- level_matrix
 - gameStruct, 16
- levelClearTexture
 - game.h, 37
- loadGame
 - savingGame.c, 48
 - savingGame.h, 50
- logoRect
 - menu.h, 66
- logoTexture
 - menu.h, 66
- MIN
 - collisions.c, 29
- main
 - main.c, 61
- main.c, 60
 - initProgram, 61
 - killGame, 61
 - main, 61
- main.h, 61
 - backgroundRect, 62
 - bgSound, 62
 - bottomStripRect, 62
 - explosionSound, 62
 - fire, 62

- font, 62
- game, 62
- lastCampaignLevel, 62
- renderer, 62
- topStripRect, 62
- window, 62
- mainMetalFrameRect
 - menu.h, 66
- maxLevelSavePtr
 - game.h, 35
- maxLevelSaveStruct, 17
 - level, 18
- menu.c
 - getLevelFileName, 63
 - getSavedGameName, 63
 - selectCampaignLevel, 63
 - showMenu, 63
- menu.h
 - backgroundOverlay, 65
 - backgroundRect, 65
 - backgroundTexture, 65
 - bgSound, 66
 - bottomStripRect, 66
 - difficulty, 66
 - events, 66
 - explosionSound, 66
 - fire, 66
 - font, 66
 - game, 66
 - helpRect, 66
 - helpTexture, 66
 - helpTipRect, 66
 - lastCampaignLevel, 66
 - logoRect, 66
 - logoTexture, 66
 - mainMetalFrameRect, 66
 - menuMusic, 67
 - metalFrameRect, 67
 - metalFrameTexture, 67
 - renderer, 67
 - selector2Pos, 67
 - selector2Rect, 67
 - selectorPos, 67
 - selectorRect, 67
 - selectorTexture, 67
 - showMenu, 65
 - TEXT_TEXTURE_NUM, 65
 - textTextures, 67
 - topStripRect, 67
 - window, 67
- menu/menu.c, 63
- menu/menu.h, 64
- menuMusic
 - menu.h, 67
- metalFrameRect
 - menu.h, 67
- metalFrameTexture
 - menu.h, 67
- movement
 - playerStruct, 20
- name
 - HSFileMemberStruct, 17
- next
 - bulletListNodeStruct, 5
 - effectListNodeStruct, 7
 - enemyListNodeStruct, 9
 - playerListNodeStruct, 18
 - powerUpListNodeStruct, 21
- PLAYERS_NUM
 - game.h, 35
- path_id
 - enemyStruct, 11
- pauseSurface
 - gameStruct, 16
- pauseTexture
 - gameStruct, 16
- player
 - playerListNodeStruct, 18
- playerIterationHandlerDefault
 - handlers.h, 42
 - iterationHandlers.c, 47
- playerIterationHandlerDemo
 - handlers.h, 43
 - iterationHandlers.c, 48
- playerListNodeStruct, 18
 - next, 18
 - player, 18
 - prev, 18
- playerStruct, 19
 - bombCoolDown, 19
 - bulletCoolDown, 19
 - counter, 19
 - handleIteration, 19
 - health, 20
 - movement, 20
 - position, 20
 - shieldActive, 20
 - speedActive, 20
 - texturePosition, 20
 - x_pos, 20
 - y_pos, 20
- playerTexture
 - gameStruct, 16
- playerTextureLeft
 - gameStruct, 16
- playerTextureRight
 - gameStruct, 16
- players
 - game.h, 37
 - gameStruct, 16
- playersCount
 - gameStruct, 16
- position
 - bulletStruct, 6
 - effectStruct, 8

- enemyStruct, 11
 - playerStruct, 20
 - powerUpStruct, 22
- powerUp
 - powerUpListNodeStruct, 21
- powerUpIterationHandler
 - handlers.h, 43
 - iterationHandlers.c, 48
- powerUpListNodeStruct, 20
 - next, 21
 - powerUp, 21
 - prev, 21
- powerUpNum
 - saveHeaderStruct, 22
- powerUpStruct, 21
 - counter, 22
 - handleIteration, 22
 - id, 22
 - position, 22
 - x_pos, 22
 - y_pos, 22
- powerUpTexture
 - gameStruct, 16
- powerUps
 - gameStruct, 16
- prev
 - bulletListNodeStruct, 5
 - effectListNodeStruct, 7
 - enemyListNodeStruct, 9
 - playerListNodeStruct, 18
 - powerUpListNodeStruct, 21
- rect
 - textStruct, 23
- renderGame
 - game.c, 33
 - game.h, 36
- renderer
 - collisions.h, 32
 - draw.h, 28
 - game.h, 37
 - handlers.h, 44
 - instances.h, 60
 - main.h, 62
 - menu.h, 67
 - savingGame.h, 51
- runGame
 - game.c, 33
 - game.h, 36
- saveGame
 - savingGame.c, 48
 - savingGame.h, 50
- saveHeaderPtr
 - savingGame.h, 50
- saveHeaderStruct, 22
 - bulletNum, 22
 - effectNum, 22
 - enemyNum, 22
 - powerUpNum, 22
 - saveHighScore
 - constructors.c, 53
 - instances.h, 59
 - savingGame.c
 - loadGame, 48
 - saveGame, 48
 - savingGame.h
 - backgroundRect, 50
 - bgSound, 50
 - bottomStripRect, 50
 - explosionSound, 50
 - fire, 50
 - font, 50
 - game, 50
 - lastCampaignLevel, 50
 - loadGame, 50
 - renderer, 51
 - saveGame, 50
 - saveHeaderPtr, 50
 - topStripRect, 51
 - window, 51
 - score
 - enemyStruct, 11
 - gameStruct, 16
 - HSFileMemberStruct, 17
 - selectCampaignLevel
 - menu.c, 63
 - selector2Pos
 - menu.h, 67
 - selector2Rect
 - menu.h, 67
 - selectorPos
 - menu.h, 67
 - selectorRect
 - menu.h, 67
 - selectorTexture
 - menu.h, 67
 - shieldActive
 - playerStruct, 20
 - shieldTexture
 - gameStruct, 16
 - show
 - textStruct, 23
 - showMenu
 - menu.c, 63
 - menu.h, 65
 - speedActive
 - playerStruct, 20
 - speedTexture
 - gameStruct, 17
 - str
 - textStruct, 23
 - swap_tile_rows
 - game.c, 34
 - TEXT_TEXTURE_NUM
 - menu.h, 65
 - TEXTS_NUM

- game.h, 35
- textStruct, 23
 - color, 23
 - font, 23
 - rect, 23
 - show, 23
 - str, 23
 - texture, 23
- textTextures
 - menu.h, 67
- texts
 - game.h, 37
- texture
 - textStruct, 23
- texture_slot
 - enemyStruct, 11
- texturePosition
 - bulletStruct, 6
 - enemyStruct, 11
 - playerStruct, 20
- topStripRect
 - collisions.h, 32
 - draw.h, 28
 - game.h, 37
 - handlers.h, 44
 - instances.h, 60
 - main.h, 62
 - menu.h, 67
 - savingGame.h, 51
- updateEffects
 - game.c, 34
- window
 - collisions.h, 32
 - draw.h, 28
 - game.h, 37
 - handlers.h, 44
 - instances.h, 60
 - main.h, 62
 - menu.h, 67
 - savingGame.h, 51
- x_pos
 - bulletStruct, 7
 - effectStruct, 8
 - enemyStruct, 11
 - playerStruct, 20
 - powerUpStruct, 22
- y_pos
 - bulletStruct, 7
 - effectStruct, 8
 - enemyStruct, 11
 - playerStruct, 20
 - powerUpStruct, 22