|  |
| --- |
| TUGAS AKHIR  PROCEDURAL MAP GENERATION UNTUK  GAME “SPLATTED” |
| LOGO ISTTS TRANS PUTIH |
|  |
| Oleh:  Lukky Hariyanto  219116856 |
| PROGRAM SARJANA  PROGRAM STUDI INFORMATIKA  FAKULTAS SAINS DAN TEKNOLOGI  INSTITUT SAINS DAN TEKNOLOGI TERPADU SURABAYA  SURABAYA  2023 |

TUGAS AKHIR

PROCEDURAL MAP GENERATION UNTUK

GAME “SPLATTED”

Diajukan Guna Memenuhi Sebagian Persyaratan Untuk Memperoleh Gelar

Sarjana Komputer

Pada

Institut Sains dan Teknologi Terpadu Surabaya

Disetujui oleh Tim Penguji Tugas Akhir:

1. Hendrawan Armanto, S.Kom., M.Kom. (Pembimbing)
2. Dr. Ir. Hj. Endang Setyati, M.T. (Penguji I)
3. Eka Rahayu Setyaningsih, S.Kom., M.Kom. (Penguji II)
4. Dr. Yosi Kristian, S.Kom. M.Kom. (Penguji III)

SURABAYA

OKTOBER 2023

SURAT PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama : Lukky Hariyanto

Fakultas/ Prodi : Sains dan Teknologi/ Informatika

NRP : 219116856

dengan ini menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir/Tesis dengan judul:

PROCEDURAL MAP GENERATION UNTUK

GAME “SPLATTED”

adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, 01-10-2023

Yang Membuat Pernyataan,

Lukky Hariyanto

219116856

ABSTRAK

Game merupakan aspek hiburan yang sekarang ini merajalela di seluruh dunia, dimana video game telah menjadi salah satu industri hiburan terbesar melebihi buku, film dan music. Tetapi itu disebabkan oleh jumlah talenta yang diperlukan untuk membuat sebuah game, diisi oleh beberapa desainer / direktur yang memimpin pekerjaan yang diperlukan seperti direktur seni, direktur lagu, direktur efek suara, desainer mekanik game dan masih banyak lagi, dengan salah satu aspek yang tidak kalah penting tapi tidak terlalu terkenal adalah desain Map.

Dalam tugas akhir ini, akan dibuat sebuah game yang mencoba memakai Map Generation. Game ini adalah Splatted, sebuah game 5v5 dimana setiap orang akan bertarung dengan mengambil salju di tanah, membuat nya menjadi sebuah bola salju, lalu melempar bola ke tim musuh sekaligus menangkap atau menghindari bola – bola yang datang ke tim nya. Di dalam game ini juga terdapat beberapa bola spesial yang bisa membuat permainan lebih menarik dengan menambah jumlah bola yang bisa dilempar oleh setiap orang. Lalu untuk map generation yang dipakai akan menggunakan algoritma genetik, sebuah algoritma yang didasarkan pada teori evolusi Darwin, dimana hewan – hewan yang memiliki fitur tertentu akan memiliki keunggulan dalam bertahan hidup dan beranak. Menggunakan itu akan dibuat 2 jenis pembuatan Map, yaitu Tile Generation yang membuat Map berdasarkan isi dari setiap tile di Map atau Template Generation dimana setiap tile 5x5 akan direpresentasikan menggunakan sebuah template yang telah dibuat secara manual. Dengan ini akan diberi 3 opsi ukuran Map kepada player, yaitu kecil (20 x 15 tile), sedang (30 x 20) tile dan besar (40 x 25 tile).

Dari game yang telah dibuat ini disimpulkan bila Procedural Map Generation bisa dipakai di kasus – kasus tertentu dalam sebuah game dan bekerja dengan baik, tapi salah satu saran yang penting untuk diingat setiap pemakai dari Procedural Map Generation adalah untuk memasukkan unsur – unsur buatan secara manual kedalam Map–Map yang dibuat karena bila tidak maka Map akan bersifat membosankan, dan dikarenakan memasukkan unsur – unsur manusiawi akan membuat Map jauh lebih unik dan menarik untuk dimainkan.

ABSTRACT

Games are an uprising aspect in the entertainment industry, surpassing books, films, and music. But that is due to the amount of talent required to make a game, filled with several designers/directors who lead the necessary work, such as art director, music director, SFX director, game mechanic designer, and many more, with one aspect that is just as important but not very well known are the Map designers.

In this thesis, a game will be created that applies Procedural Map Generation. This game, henceforth named Splatted, is a 5v5 game where each person will fight by picking up snow on the ground, turning it into a snowball, and then throwing the ball at the enemy team while simultaneously catching or avoiding the balls that come towards their team. Several unique balls can make the game more interesting in this game by increasing the variety of balls each person can throw. Then, the Map Generation will use a Genetic Algorithm, an algorithm based on Darwin's theory of evolution, where animals that have specific features will have an advantage in surviving and procreating. Using it, two types of Map creation will be created: Tile Generation, which creates Maps based on the contents of each tile in the Map, or Template Generation, where each 5x5 tile will be represented using a template made by hand. With this, players will be given 3 Map size options, namely small (20 x 15 tiles), medium (30 x 20) tiles, and large (40 x 25 tiles).

From the game that has been finished, it can be concluded that Procedural Map Generation can be used in some instances in a fun and works very well. Still, one of the critical pieces of advice for every user of Procedural Map Generation to remember is to include handmade elements in the maps created. Otherwise, the maps would be boring, and including human-made details would make the maps much more unique and exciting to play.

KATA PENGANTAR

Game merupakan bagian besar dari hidup saya dimulai dari masa SD saya. Dan mungkin ini berlaku sama bagi banyak orang lain, oleh karena itu, saat saya besar saya memiliki mimpi untuk menjadi sebuah game developer. Tetapi mimpi tersebut tidak diwujudkan hingga saat saya memulai belajar di iSTTS, dimana saya menyadari bila kita menginginkan sesuatu, kita tidak bisa menunggu kesempatan itu datang, tetapi kita harus membuat kesempatan itu sendiri. Oleh karena itu saya belajar di iSTTS selama 4 tahun, berkat bimbingan dosen – dosen yang sabar, perhatian dan teman – teman seperjuangan, saya bisa mencapai titik ini. Sebagai salah satu syarat kelulusan dari iSTTS, kita diwajibkan untuk membuat sebuah tugas akhir, entah proyek atau studi belajar, dan mengingat mimpi saya, untuk tugas akhir ini saya membuat sebuah game yang diharapkan bisa dinikmati teman – teman saya, dengan harapan saya bisa membuat game yang lebih bagus di masa depan. Saya ucapkan terima kasih banyak terhadap dosen – dosen yang membimbing saya di iSTTS beserta karyawan – karyawannya, orang tua saya yang mendorong saya untuk menjadi lebih baik, teman – teman saya yang membantu saya selama ini dan terakhir adalah Pak Hendrawan yang membimbing saya seperjalanan membuat tugas akhir ini.

Surabaya, Oktober 2023

Penulis

DAFTAR ISI

Halaman

HALAMAN JUDUL i

HALAMAN PENGESAHAN ii

SURAT PERNYATAAN KEASLIAN iii

ABSTRAK iv

ABSTRACT v

KATA PENGANTAR vi

DAFTAR ISI vii

DAFTAR GAMBAR xiv

DAFTAR ALGORITMA xx

DAFTAR SEGMEN PROGRAM xxi

DAFTAR RUMUS xxiii

BAB I PENDAHULUAN 1

1.1 Latar Belakang 1

1.2 Tujuan 2

1.3 Teori Penunjang 2

1.3.1 OOP 3

1.3.2 Algoritma A\* 4

1.3.2 Finite State Machine 5

1.3 Ruang Lingkup 6

1.4.1 Fitur Game 6

1.4.2 Batasan Masalah 7

1.4.3 Tahap penyelesaian 7

1.4.4 Uji Coba 8

1.5 Sistematika Pembahasan 8

BAB II GENETIC ALGORITHM 10

2.1 Terminologi 10

2.1.1 Gene & Chromosome (Kromosom) 10

2.1.2 Populasi / Generasi 10

2.1.3 Fitness 11

2.1.4 Parent Child Chromosome 11

2.2 Metodologi 12

2.3 Langkah – Langkah Algoritma Genetik 13

2.3.1 Inisialisasi Populasi 13

2.3.2 Seleksi 14

2.3.3 Reproduksi 14

2.3.4 Mutation (Mutasi) 15

2.3.5 Elitism 16

2.3.6 Repetisi dan Pemberhentian 16

2.4 Library GeneticSharp 17

2.4.1 Permasalahan 17 2.4.2 Representasi 17 2.4.3 Fitness Function 18 2.4.4 Metode – Metode Pemrosesan Data 19 2.4.5 Pseudocode dan Penjelasan 20

2.4.6 Tracing 23

BAB III UNITY 28

3.1 UI & Scene 28

3.2 Game Object & Prefab 30

3.3 Script 36

3.4 Tilemap & Rule Tile 37

3.5 State Machine 38

3.6 Shader Graph 39

BAB IV ANALISA GAME INSPIRASI 40

4.1 CrossCode 41 4.1.1 Story 42 4.1.2 Gameplay 45 4.1.3 Stats & Elements 47 4.1.4 Skill Tree 48 4.1.5 Combat Arts 51 4.1.6 Equipment 52 4.1.7 Consumables 53 4.1.8 Status Conditions 54 4.1.9 Party 55 4.1.10 UI & UX 57 4.1.11 Enemies & Bosses 65 4.1.12 PVP 67 4.1.13 Level Design 69 4.1.14 Aspek Inspirasi 70

4.2 Knockout City 71

4.2.1 Gameplay 71 4.2.2 UI 74

4.2.3 Special Ball 75

4.2.4 Level Design 76

4.2.5 Aspek Inspirasi 77

4.3 Spelunky 2 77

4.3.1 Story 78

4.3.2 Gameplay 79

4.3.3 Environment 82

4.3.4 Items 83

4.3.5 Shops 84

4.3.6 Map Generation 85

4.3.7 Aspek Inspirasi 85

BAB V DESAIN GAME 86

5.1 Flow Aplikasi & UX 86

5.2 Gameplay 87

5.3 Main Level 88

5.4 Bola Spesial 88

5.4.1 Ice Piercer 89 5.4.1 Snow-a-rang 89

5.4.1 Explod-o-Ball 89

5.4.1 Freezing Winter 90

5.4.1 Stone Auger 90

5.5 User Interface 91

5.6 Desain Aksi Player / Bot 92

5.6.1 Movement / Gerakan 92

5.6.2 Mengambil Bola 93

5.6.3 Menangkap Bola 93

5.6.4 Melempar Bola 94

5.6.5 Kena Lempar Bola 95

5.6.6 Fakeout (Khusus Player) 95

5.7 Tutorial 95

5.8 Map Generation 96

5.8.1 Jenis Generasi Map & Representasi 97

5.8.1.1 Tile Generation 97

5.8.1.2 Template Generation 98

5.8.2 Fitness Level 99

5.8.2.1 Jumlah batu dalam Map (Tile Generation) 99

5.8.2.2 Ukuran kelompok batu (Tile Generation) 101

5.8.2.3 Luas Area yang bisa diakses 102

5.8.2.4 Jumlah bola spesial yang bisa diambil 102

5.8.2.5 Rasio antara bola spesial dan luas area level 103

5.8.2.6 Variasi template dalam level (Template) 104

5.8.7 Selection 105

5.8.8 Crossover 106

5.8.9 Mutation 107

5.8.10 Stop Condition 107

5.8.10 Variabel-Variabel pelengkap algoritma genetik 108

5.9 Bot State Machine 108

5.9.1 Start State 109

5.9.2 Mencari Bola 109

5.9.3 Ambil Bola 109

5.9.4 Mencari Musuh 110

5.9.5 Bidik Musuh 110

5.9.6 Kejar Musuh 110

5.9.7 Coba Tangkap Bola 110

BAB VI IMPLEMENTASI MAP GENERATION

MENGGUNAKANF GENETIC ALGORITHM 111

6.1 Chromosome / Representasi 111

6.1.1 Tile Chromosome 111

6.1.2 Templated Map Chromosome 113

6.2 Main Code 116 6.3 Function FitnessFunction 119 6.4 Fitness Individual 121

6.4.1 Rock Amount Fitness 123

6.4.2 Rock Groups Size Fitness 126

6.4.3 Area Fitness 128

6.4.4 Power Up Access Fitness 129

6.4.5 Power Up Ratio Fitness 131

6.4.6 Template Variety Fitness 132

BAB VII IMPLEMENTASI GAME 136

7.1 Mengolah Hasil Genetic Algorithm Menjadi Level 136

7.1.1 SetObjects 136

7.1.2 MakeFloor 139

7.1.3 PlayersManager 140

7.2 Bola Salju 142

7.2.1 Snowball Manager 143

7.2.2 Ball Movement 146

7.2.3 Bola Spesial 147

7.2.4 Power Up 151

7.3 Karakter Player 155

7.3.1 SnowBrawler 155

7.3.2 Movement / Gerakan 158

7.3.3 Shoot Mechanic 160

7.3.4 Catch Ball 162

7.4 Karakter Bot 163

7.4.1 Bot Actions 163

7.4.2 Metode Melempar Bot 167

7.4.3 Coordinate Movement 168

7.4.4 A\* Algorithm 173

7.4.5 State Machine 178

7.4.5.1 Random Walking 179

7.4.5.2 Random Walking Menuju Ambil Bola 179

7.4.5.3 Ambil Bola 180

7.4.5.4 Ambil Bola Menuju Random Walking 180

7.4.5.5 Random Walking Menuju Aim & Throw 180

7.4.5.6 Aim & Throw 181

7.4.5.7 Aim & Throw Menuju Random Walking 181

7.4.5.8 Aim & Throw Menuju Follow Target 182

7.4.5.9 Follow Target 182

7.4.5.10 Follow Target menuju Random Walking 182

7.4.5.11 Follow Target menuju Aim & Throw 183

7.4.5.12 Any State menuju Catch Ball 183

7.4.5.13 Catch Ball dan transisinya 184

7.5 Sistem Timer & Skor 185

BAB VIII UJI COBA 188

8.1 Tes AI Dari Bot 188

8.1.1 Bisa Melihat Dan Mengambil Salju 189

8.1.2 Mencari Lawan Dan Membidiknya 189

8.1.3 Melempar Bola Dengan Prediksi 190

8.1.4 Mencari Salju Baru Setelah Melempar Bola 191

8.1.5 Menangkap Bola 193

8.2 Evaluasi Level Yang Dibuat Genetic Algorithm 192

8.2.1 Level Dari Template Generation 192

8.2.2 Level Dari Tile Generation 196

8.3 Kusisioner 200

8.3.1 Generasi Level 201

8.3.2 Kualitas Bot 203

8.3.3 Keseruan Dari Game 203

BAB IX PENUTUP 205

9.1 Kesimpulan 205

9.2 Saran 205

DAFTAR PUSTAKA 207

RIWAYAT HIDUP 208

LAMPIRAN A KUESIONER A-1

LAMPIRAN B TAMPILAN PROGRAM B-1

LAMPIRAN C TEMPLATE YANG TERSEDIA DALAM SPLATTED C-1

LAMPIRAN B GAMBAR-GAMBAR STATE MACHINE D-1

DAFTAR GAMBAR

Gambar Halaman

1.1 Nilai Dari Setiap Industri Hiburan Menurut Statista 1

1.2 Contoh Object Oriented Programming Pada Mobil 3

1.3 Visualisasi Sederhana Algoritma A\* 4

1.4 Contoh Perhitungan Algoritma A\* 4

1.5 Contoh Finite State Machine 5

2.1 Visualisasi Gene, Kromosom dan Populasi 11

2.2 Visualisasi Garis besar cara kerja Algoritma Genetik 12

2.3 Visualisasi One Point Crossover 14

2.4 Visualisasi Random Mutation 15

2.5 Visualisasi konsep Elitism 16

2.6 Sebuah Kubus dan 2 garis diagonal 17

2.7 Rumus menghitung diagonal dari kubus 18

2.8 Rumus menghitung jarak antara 2 titik di dalam kubus 18

2.9 Generasi Pertama 23

2.10 Persentase seluruh Populasi dan Visualisasi roda Roulette 24

2.11 Persentase dan Visualisasi Fitness kuadrat 25

2.12 Hasil Memutar roda Roulette 26

2.13 Crossover pasangan kromosom pertama dan hasil keseluruhan 26

2.14 Mutasi kromosom pertama dan hasil keseluruhan 27

3.1 Logo Unity sebelum 2021 dan dari 2021 hingga sekarang 28

3.2 UI Unity 29

3.3 Tampilan Transform 30

3.4 Tampilan Sprite Renderer 31

3.5 Tampilan Animator 32

3.6 Isi dari sebuah Animation Controller 32

3.7 Isi dari sebuah Transisi 33

3.8 Rigidbody 2D 34

3.9 Komponen Box Collider 2D dan tampilan di dalam Gameobject 35

3.10 Komponen Script Bot Action dan isi Variabel 36

3.11 Contoh Tilemap 37

3.12 Rule Tile 37

3.13 State Machine 38

3.14 Shader Graph 39

4.1 Cuplikan Layar CrossCode 41

4.2 Dari Kiri; Sergei Asimov, Carla Salvator dan Lea berbincang 43

4.3 Gerald “Jet” McFly didatangi oleh seorang Avatar biru 43

4.4 Lea dan Carla dihadang oleh monster milik avatar biru 44

4.5 Lea menyerang dengan VPI dan VRP 45

4.6 Lea membidik Charged Shot untuk menyelesaikan puzzle 46

4.7 5 Elemen dalam Crosscode dan Lea mengalami elemental overload 48

4.8 Tampilan Skill Tree secara keseluruhan 49

4.9 Tampilan Skill Tree elemen netral 49

4.10 Tampilan Skill Tree elemen Es 50

4.11 Special Point Bar yang ditampilkan 51

4.12 Combat Art jarak jauh Es level 3 “Gatling Arctillery” 52

4.13 Menu Equipment Lea 52

4.14 Lea Memakan Sandwich 54

4.15 Tampilan keempat status condition 54

4.16 Lea bertarung bersama dengan Emilie 55

4.17 Tampilan Menu mengajak anggota party 56

4.18 Menu untuk mengatur perilaku dari para party member 57

4.19 UI Crosscode secara normal dan saat shift ditahan 58

4.20 UI menu saat tab ditekan 59

4.21 Equipment lain dihover untuk melihat perbedaan stat 61

4.22 Dipencet help dan mouse melayang di modifier Bullseye 62

4.23 Menu Inventory Consumables 63

4.24 Menu Inventory Arms 64

4.25 Menu Inventory Trade 65

4.26 Lea melakukan Break pada Rockin’ Hillkat 66

4.27 PVP antara Lea dan Apollo di Bergen Trail 67

4.28 Apollo dan Lea sebelum duel dimulai 68

4.29 Sebuah silver chest di sebelah kiri layar 69

4.30 Screenshot game Knockout City 71

4.31 Aksi yang tersedia : Lempar, Fake out, Tangkap, dan Dash 72

4.32 Player menjadi bola dan player dilempar 73

4.33 UI Knockout City secara normal dan saat dilempar bola 74

4.34 Seluruh bola yang terdapat di dalam Knockout City 75

4.35 Gambar Jukebox Station dari wiki 76

4.36 Screenshot game Spelunky 2 77

4.37 Halaman terakhir jurnal Guy Spelunky yang ditinggal di bulan 77

4.38 Player membawa mayat musuh untuk menangkis panah perangkap 79

4.39 Player melempar bom dan bom meledak 80

4.40 Player melempar tali dan memanjat tali yang dilempar 81

4.41 Player mati kena perangkap tinju 82

4.42 Player bersama dengan kucingnya Percy 82

4.43 Player membawa sebuah teleporter 83

4.44 Player melakukan stun kepada penjaga toko untuk mencuri barang 84

5.1 Flow dari Splatted 86

5.2 Ice Piercer 89

5.3 Snow-a-rang 89

5.4 Explod-o-Ball 89

5.5 Freezing Winter 90

5.6 Stone Auger 90

5.7 Design UI untuk permainan Splatted 91

5.8 Dodo 92

5.9 Ilustrasi cara kerja A\* untuk mencapai tujuan 93

5.10 Ilustrasi Dodo membidik seseorang 94

5.11 Contoh Representasi Tile berbentuk 1 Dimensi dan 2 Dimensi 97

5.12 Contoh Representasi Template 1 Dimensi, 2 Dimensi dan hasil akhir 98

5.13 Hasil nilai m fungsi 5.1 100

5.14 Grafik hasil fitness contoh yang diberi sebelum dikuadrat 101

5.15 Hasil perhitungan template contoh 105

5.16 Hasil perhitungan template contoh dengan toleransi 105

5.17 Visualisasi Roulette Wheel Selection 106

5.18 Visualisasi Uniform Crossover 106

5.19 Visualisasi Partial Shuffle Mutation 107

5.20 State Machine Diagram Bot 108

6.1 Isi Variabel pengaturan dari algoritma genetik 119

6.2 Isi Variabel dari Rock Amount Fitness 125

6.3 Isi Variabel dari Rock Groups Size Fitness 127

6.4 Isi Variabel dari Power Up Ratio Fitness 132

6.5 Isi Variabel dari Template Variety Fitness 134

7.1 Variabel Set Objects di dalam Level utama 138

7.2 Tile batu dan Tampilan level dengan setobject dan Player Manager 139

7.3 Tile tanah salju dan Tampilan level hanya dengan makeFloor 140

7.4 Isi variabel player manager 142

7.5 Gumpalan salju yang di-spawn secara berkala 143

7.6 Isi Variabel dari script SnowBallManager 145

7.7 Bola – bola salju dengan power up id 0 – 5 148

7.8 Isi Variabel Ball Power Up 151

7.9 Tampilan Spawner Bola Spesial di dalam game 152

7.10 Variabel yang dipakai Script power up 154

7.11 Tampilan Karakter membidik sebuah bola 158

7.12 Variabel yang digunakan pada shoot Mechanic 161

7.13 Tampilan karakter player membidik bola 162

7.14 Tampilan karakter player mencoba menangkap bola 163

7.15 Variabel – variabel yang digunakan oleh Bot Actions 166

7.16 Bot menggunakan Linecast untuk “melihat” 167

7.17 Hukum Cosine 168

7.18 Hukum Cosine untuk keperluan prediksi lemparan bola 168

7.19 Rumus kuadrat ABC 171

7.20 State Machine dari bot 178

7.21 Isi state Follow Target 181

7.22 Isi state Catchball, transisi ke jalan dan ke Aim & Throw 184

7.23 Tampilan layar Splatted 185

7.24 Isi Variabel Bar Score Manager 186

8.1 Level yang akan digunakan untuk mencoba AI dari bot 188

8.2 Bot melihat bola dan mengambilnya 189

8.3 Bot melihat lawan dan membidik player 189

8.4 Bot memprediksi melempar boomerang ke kanan atas 190

8.5 Bot melihat bola dan berhasil mengambilnya 191

8.6 Bentuk lokasi uji coba tangkap bola bot 191

8.7 Bot menangkap bola di depannya 192

8.8 Level Pertama,fitnessnya dan array id templatenya 193

8.9 Level Kedua,fitnessnya dan array id templatenya 194

8.10 Level ketiga,fitnessnya dan array id templatenya 194

8.11 Level keempat,fitnessnya dan array id templatenya 195

8.12 Level kelima,fitnessnya dan array id templatenya 196

8.13 Level kedua dan fitnessnya 197

8.14 Level kedua dan fitnessnya 198

8.15 Level ketiga dan fitnessnya 198

8.16 Level keempat dan fitnessnya 199

8.17 Level kelima dan fitnessnya 200

8.18 Pendapat responden mengenai kualitas level yang dibuat 201

8.19 Jumlah metode Generation tertentu yang dipakai responden 202

8.20 Pendapat responden berdasarkan Map Generation yang dipakai 202

8.21 Pendapat responden mengenai kualitas bot yang dibuat 203

8.22 Pendapat responden mengenai keseruan game 203

C.1 Isi Template One Way C-1

C.2 Isi Template Two Way C-1

C.3 Isi Template Four Way C-1

D.1 Isi state Random Walking D-1

D.2 Isi transisi Jalan menuju ambil bola D-2

D.3 Isi state Ambil Bola D-3

D.4 Transisi Ambil bola ke berjalan D-4

D.5 Bagian kiri dan kanan dari transisi jalan ke bidik bola D-6

D.6 Isi state Aim & Throw D-7

D.7 Transisi aim & throw ke jalan bagian kiri dan kanan D-9

D.8 Transisi Aim & Throw menuju Follow Target D-10

D.9 Isi transisi Follow Target menuju jalan bagian kiri dan kanan D-12

D.10 Isi transisi Follow Target menuju Aim & Throw D-13

D.11 Isi transisi menangkap bola D-13

DAFTAR ALGORITMA

Algoritma Halaman

2.1 Pseudocode Library GeneticSharp 20

DAFTAR SEGMEN PROGRAM

Segmen Program Halaman

6.1 Class TileChromosome 111

6.2 Templated Map Chromosomes 115

6.3 Fungsi FitnessFunction 120

6.4 Class Abstrak FitnessBase 121

6.5 Class Abstrak InLoopFitnessBase 123

6.6 Class Rock Amount Fitness 123

6.7 Power Up Access Fitness 129

6.8 Template Variety Fitness 132

7.1 Class Make Floor 139

7.2 Class PowerUp 152

7.3 Class Player Movement 158

7.4 Class CatchBall 162

7.5 Class Coordinate 173

7.6 Class A Star Node 174

A.1 Class PossibleTemplates A-1

A.2 Genetic Algorithm A-2

A.3 Rock Groups Size Fitness A-3

A.4 Area Fitness A-5

A.5 Power Up Ratio Fitness A-6

B.1 Class Set Objects B-1

B.2 Class Players Manager B-3

B.3 Class Snow Ball Manager B-6

B.4 Class BallMovement B-9

B.5 Class Ball Power Up B-11

B.6 Class SnowBrawler B-13

B.7 Class Shoot Mechanic B-18

B.8 Class BotActions B-20

B.9 Fungsi GetAngle B-24

B.10 Fungsi Coordinate Movement B-26

B.11 A Star Algorithm B-27

B.12 Bar Score Manager B-30

DAFTAR RUMUS

Rumus Halaman

2.1 Perhitungan Euclidian Distance Kubus 24

5.1 Rumus Nilai Minus Jumlah Batu 99

5.2 Rumus Nilai Minus Maksimal Jumlah Batu 99

5.3 Rumus Fitness Jumlah Batu 99

5.4 Rumus Nilai Minus Kelompok Batu 101

5.5 Rumus Nilai Minus Maksimal Kelompok Batu 101

5.6 Rumus Fitness Kelompok Batu 101

5.7 Rumus Fitness Luas Area 102

5.8 Rumus Fitness Bola Spesial 102

5.9 Rumus Nilai Minus Rasio Bola Spesial 103

5.10 Rumus Nilai Minus Maksimal Rasio Bola Spesial 103

5.11 Rumus Fitness Rasio Bola Spesial 103

5.12 Rumus X Variasi Template 104

5.13 Rumus Fitness Variasi Template 104

7.1 Nilai dc Hukum Cosine 169

7.2 Nilai db Hukum Cosine 169

7.3 Nilai t Hukum Cosine 170

7.4 Nilai db Baru Hukum Cosine Menggunakan Rumus 7.2 170

7.5 Nilai r Hukum Cosine 170

7.6 Nilai db Baru Hukum Cosine Dari Rumus 7.4 170

7.7 Persamaan Menemukan dc Hukum Cosine 170

7.8 Penyederhanaan Persamaan dc Hukum Cosine 170

7.9 Rumus Kuadrat 170

7.10 Nilai a dalam Rumus Kuadrat 170

7.11 Nilai b dalam Rumus Kuadrat 171

7.12 Nilai c dalam Rumus Kuadrat 171

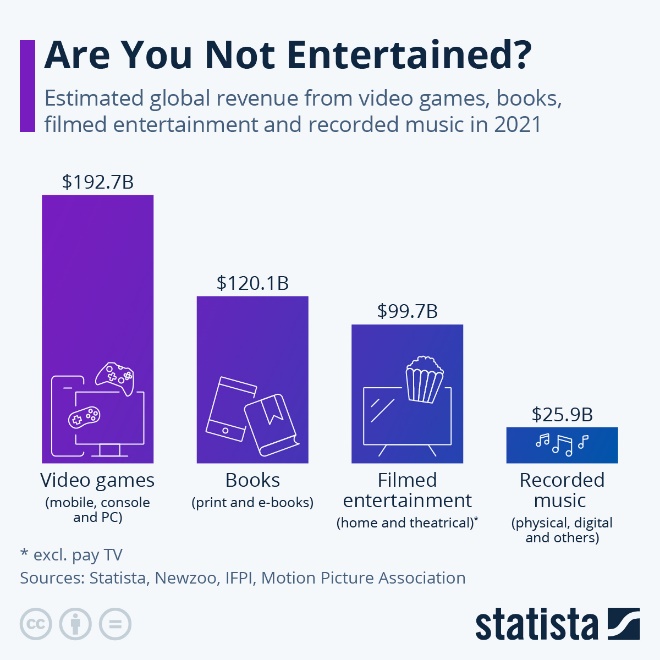
BAB I

PENDAHULUAN

Dalam bab ini akan dijelaskan mengenai garis besar buku tugas akhir yang telah dibuat, seperti latar belakang, metode uji coba hingga Sistematika pembahasan dari buku ini. Berikut adalah subyek – subyek yang akan dijelaskan :

1. Latar Belakang

Beberapa tahun terakhir ini game merupakan sebuah hiburan yang banyak digunakan, dimana hampir setiap orang yang memiliki komputer atau smartphone setidaknya pernah memainkan 1 game. Terdapat game untuk setiap orang, dimana bila orang tersebut sering sibuk bisa memainkan game untuk bersantai, atau ada juga beberapa yang bermain game untuk bersosialisasi, dan masih masih banyak lagi Oleh sebab itu, game menjadi salah satu industri hiburan terbesar sekarang ini melebihi industry film dan buku.



Gambar 1.1

Nilai dari setiap industri hiburan menurut statista

Setiap game yang dibuat memerlukan beberapa aspek untuk berjalan dengan baik agar game bisa berjalan dengan lancar, seperti desain Gameplay, arah Game, desain suara, desain UI, dan banyak lagi. Tapi ada satu lagi yang tersedia dalam hampir seluruh game, termasuk seri Dark Souls, Valorant dll. yang sangat membantu dalam membuat game-game tersebut menjadi bagus, yaitu map design.

1. Tujuan

Untuk setiap Map di Valorant, setiap boss Arena di Dark Souls, lokasi – lokasi bersembunyi di Sekiro, lantai-lantai dalam Rainbow Six Siege, terdapat seorang/tim desainer Map yang membuat Map tersebut agar bisa dinikmati banyak pemain. Diperlukan dan waktu yang lama untuk mendesain sebuah map yang bagus dan memiliki sebuah arah, dimana mungkin ada bos yang menghadiahkan player untuk mencari rute lain dengan memberi tempat menyerang yang menguntungkan, atau area dimana 2 tim berkonflik dengan beberapa tempat sembunyi strategis yang bisa mengubah arus pertandingan, semua itu tidak mudah dilakukan.

Tidak hanya itu, Map hasil karya desainer tidak memiliki banyak variasi setelah permainan pertama dikarenakan map yang dibuat telah didesain dengan sebuah tujuan tertentu, dan mengganti sedikit saja bagian dari Map bisa mengubah pengalaman yang diberikan oleh level yang menggunakan Map Tersebut. Ini menyebabkan map yang dibuat bersifat statis dan karena itu, beberapa game yang memerlukan replayability bisa kesulitan membuat map yang banyak bila seluruh level didesain secara manual seperti pada kasus game Roguelike.

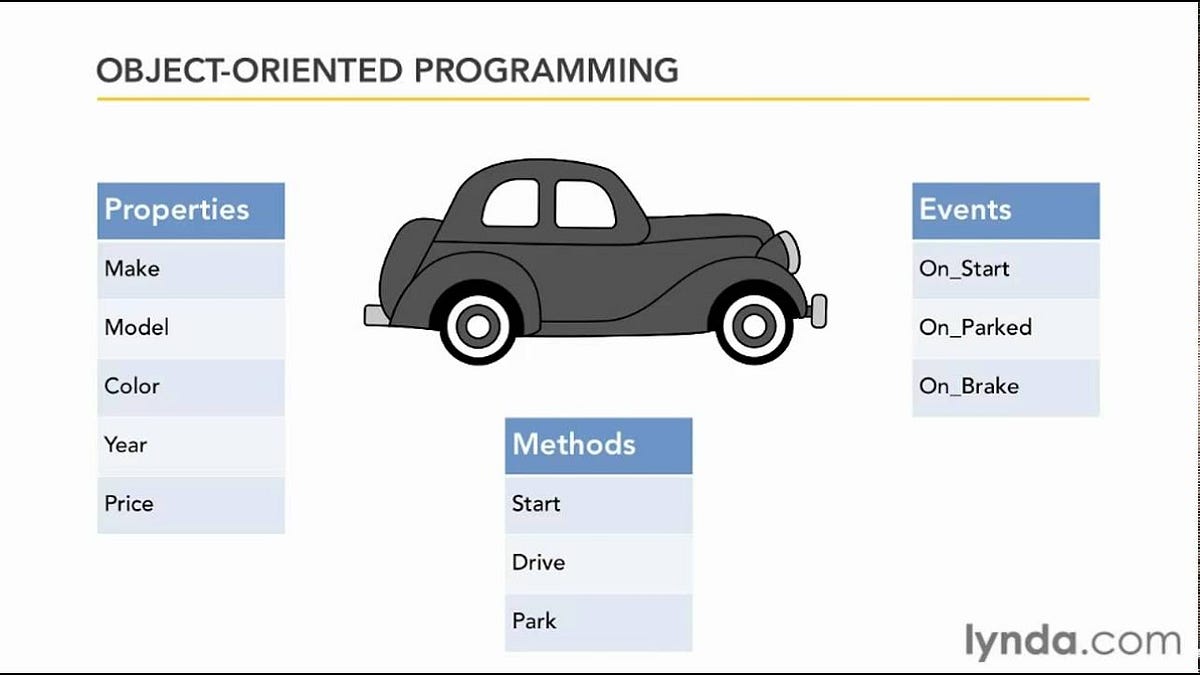
Solusi (atau kompromi) terhadap kasus dia atas adalah untuk menggunakan algoritma yang membuat level untuk dimainkan player. Metode yang dinamakan Procedural Map Generation ini telah dipakai dalam beberapa game, seperti Minecraft, Deep Rock Galactic, No Man Sky dan beberapa game lain. Dengan mengorbankan arah dan keunikan dari sebuah map, bisa dibuat lebih banyak map yang bisa dimainkan tanpa banyak usaha, meningkatkan replayibility game dari aspek level. Dan itulah tujuan dari tugas akhir ini, yaktu untuk membuat sebuah game yang bisa membuat levelnya sendiri dengan Procedural Map Generation menggunakan algoritma genetik.

1. Teori Penunjang

Dalam subbab ini akan dijelaskan beberapa konsep yang perlu diketahui sebelum melanjutkan ke bab–bab berikutnya, agar saat aspek tersebut disinggung pembaca dapat mengetahui apa aspek yang disinggung atau dapat membaca bagian ini apabila perlu menyegarkan ingatan. Berikut adalah aspek yang akan dijelaskan:

1. OOP

OOP atau Object Oriented Programming adalah sebuah model pemrograman komputer yang mengimajinasikan sebuah class sebagai sebuah objek yang memiliki atribut dan fungsi, dimana class yang berperilaku sebagai objek tersebut dapat dijalankan selayaknya objek di dunia nyata. Berikut adalah sebuah contoh :



Gambar 1.2

Contoh Object Oriented Programming pada mobil

Di gambar diatas, sebuah “Class” mobil memiliki beberapa properti, seperti merek(anggap saja Make itu Maker/pembuat), model, warna, tahun dibuat, dan harga mobil, dimana semua variabel tersebut bisa diubah – ubah. Berikutnya adalah Method, yang merupakan fungsi dari mobil tersebut dimana mobil punya 3, yaitu memulai mesin, mengendarai mobil dan parkir. Events adalah apa yang terjadi saat sesuatu dilakukan tapi bagian tersebut tidak perlu dibahas karena itu merupakan konsep lanjutan yang bersifat opsional.

1. Algoritma A\*

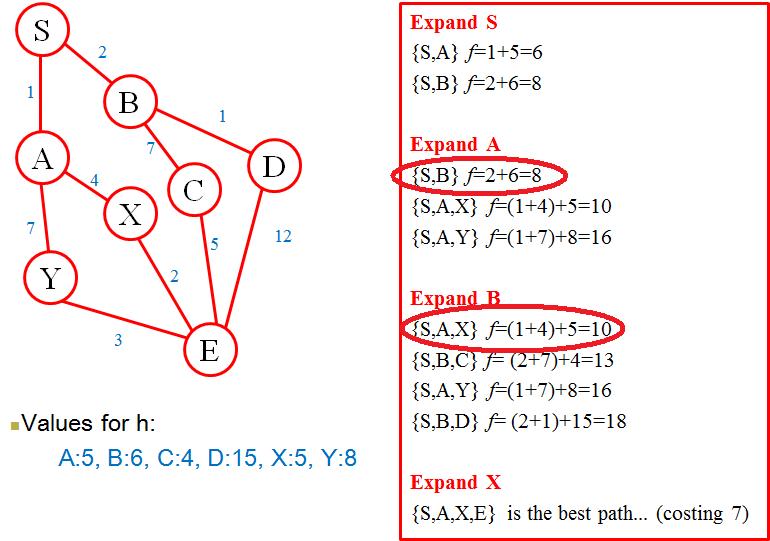
Algoritma A\* adalah sebuah algoritma path-finding (penemu jalan) yang populer sesuai namanya dipakai untuk menemukan jalan terbaik dalam mencapai sebuah tujuan. Berikut adalah sebuah visualisasi sederhana dari sebuah algoritma A\* :



Gambar 1.3

Visualisasi sederhana algoritma A\*

Dalam algoritma A\*, terdapat 3 variabel yang ditetapkan dalam setiap tile, yaitu f, g dan h. h adalah heuristic dari sebuah tile, menandakan seberapa jauh tile tersebut dari tujuan, kedua adalah g, yang berisi jarak yang ditempuh dari awal hingga tile tersebut, dan terakhir adalah f, yang merupakan hasil penjumlahan dari g dan h. Tujuan utama dari setiap tile yang ditempuh adalah untuk mencari tile dengan f yang terkecil.



Gambar 1.4

Contoh menghitung algoritma A\*

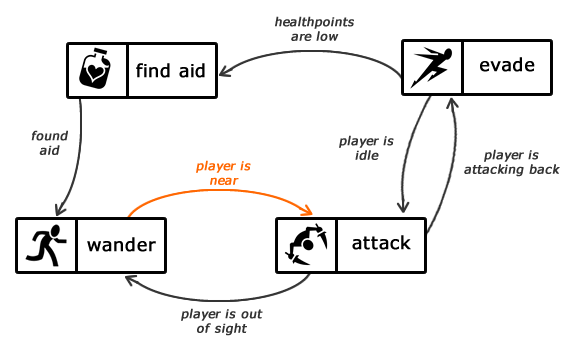
Dengan contoh gambar diatas, Kita memulai dari S, lalu lihat f dari A dan B. Dilihat diantara kedua itu node mana yang lebih kecil f-nya. Dilihat bila SA lebih kecil f nya, jadi buka semua posibilitas dari SA, tetapi jangan buang opsi yang tidak dipilih, dilihat dari bagian Expand A di kanan, dimana masih ada SB, SAX dan SAY.

Tahap selanjutnya adalah memilih rute yang memiliki f paling kecil dari ketiga rute yang tersedia, dalam kasus ini adalah SB. Dikarenakan SB memiliki f paling kecil, maka buka semua posibilitas yang dimiliki oleh B, dan hasilnya ada di segmen Expand B.

Lalu cari dari semua rute f yang paling kecil, dimana f paling kecil terdapat di rute SAX, dan dari situ, buka rute X, dimana pilihan X hanyalah SAXE, sehingga bandingkan SAXE dengan rute yang lain. Dikarenakan SAXE memiliki f paling kecil dan SAXE mencapai tujuan, maka SAXE ditentukan sebagai rute terbaik.

1. Finite State Machine

State machine atau juga dipanggil sebagai Finite State Machine adalah sebuah konsep dimana setiap aksi yang dilakukan oleh bot adalah sebuah state, dan setiap kali sebuah kriteria terpenuhi, maka state yang dijalankan oleh bot dapat berubah. Oleh karena itu, bot bisa terlihat seakan – akan paham dengan kondisinya saat ini, dan bisa beraksi sesuai dengan kondisi yang dialami, gambar 5.9 dibawah adalah contoh sempurna atas sebuah State Machine sebuah bot.



Gambar 1.5

Contoh Finite State Machine

Dimulai dari state wander / berjalan, bot akan berjalan – jalan ke posisi yang diinginkan. Bila bot menemukan sebuah player, maka bot akan berpindah ke state attack dimana bot akan mencoba menyerang player. Bila player kabur maka bot kembali jalan – jalan, tapi bila player menyerang balik maka bot akan mencoba menghindari serangan player.Bila bot kesakitan maka bot akan mencari pertolongan medis, dan setelah menemukan seperti ramuan minyak angin cap lebah, maka bot akan kembali berjalan – jalan sesuka hati.

1. Ruang Lingkup

Dalam subbab ini akan dijelaskan mengenai batasan–batasan dari game Splatted agar tidak terdapat Feature Creep (kemauan untuk menambah fitur baru dalam pengembangan aplikasi) dalam game ini. Berikut adalah ruang lingkup yang akan dibahas:

1. Fitur Game

Fitur yang dimiliki oleh game ini adalah :

* Map Generation

Map akan dibuat oleh algoritma, sehingga seluruh level akan berbeda diantara satu sama lain

* Tutorial Level

Terdapat Tutorial Level untuk mengajari dasar-dasar dari game Splatted agar player tidak kebingungan saat memainkan game

* Bola Spesial

Terdapat bola–bola spesial terpencar di dalam level yang bisa membuat permainan lebih menarik

* Basic Acrions

Karakter player bisa melakukan beberapa aksi, seperti berjalan, mengambil salju di tanah, menangkap bola, melempar bola dan melakukan fake-out melempar

* AI driven bots

Sebagai pengganti dari player karena game ini tidak memiliki fitur online, terdapat bot – bot yang menggunakan State Machine sebagai pengganti player.

1. Batasan Masalah

Dengan fitur yang diterapkan diatas, terdapat ada beberapa masalah yang dihadapi oleh game Splatted, yaitu :

* Hanya bisa dimainkan 1 player
* Tidak ada cerita
* Tidak ada perbedaan antara seluruh karakter kecuali outline tim
* Permainan berasa singkat

1. Tahap Penyelesaian

Dalam pembuatan game Splatted, berikut adalah tahap – tahap yang dilakukan :

* 1. Mempelajari Unity
  2. Mencari placeholder asset sebagai pengganti sementara asset game
  3. Membuat Level placeholder dan UI sederhana
  4. Implementasi Pergerakan Player
  5. Implementasi Sistem Bola salju
  6. Implamentasi pengambilan bola
  7. Buat Map Generation
  8. Implementasi bola spesial
  9. Implementasi Dummy / Bot sederhana
  10. Buat Shader untuk membedakan antara tim player dan tim musuh
  11. Implementasi State Machine dan fungsi – fungsi yang berhubungan
  12. Implemenysi menangkap bola untuk bot dan player
  13. Implementasi fakeout untuk player
  14. Kembangkan UI permainan
  15. Buat scene non – level seperti main menu, setting dkk
  16. Buat transisi antar scene
  17. Buat tutorial
  18. Implementasikan skor dan kondisi menang kalah
  19. Buat / cari asset game

1. Uji Coba

Untuk uji coba, akan digunakan 3 metode untuk mengukur kualitas game, yaitu :

* Uji coba AI bot
* Evaluasi level yang dibuat algoritma genetik
* Quisioner Kualitas Game

1. Sistematika Pembahasan

Dalam subbab ini akan dijelaskan bab – bab yang ada di dalam buku ini dan kegunaan dari masing – masing bab agar memudahkan pencarian subjek yang berhubungan. Bab – bab yang ada di buku ini adalah sebagai berikut :

* BAB I : PENDAHULUAN

Pada bab ini akan dijelaskan mengenai latar belakang beserta dengan proses pembuatan game Splatted, berserta dengan batasan – batasannya.

* BAB II : GENETIC ALGORITHM

Bab ini akan menjelaskan mengenai algoritma genetik, sebuah algoritma yang dipakai sebagai tulang punggung dari tugas akhir ini dalam sebuah Procedural Map Generation

* BAB III : UNITY

Bab ini akan menjelaskan Unity dan cara kerja dari beberapa bagian Unity yang memiliki peran penting dalam pembuatan game ini.

* BAB IV : ANALISA GAME

Bab ini akan digunakan untuk melakukan analisa terhadap beberapa game lain dengan harapan bisa mendapat inspirasi cara kerja dari game tersebut.

* BAB V : DESAIN GAME

Bab ini akan menjelaskan desain dari game Splatted dan alasan – alasan mengapa game didesain seperti itu, bersama dengan desain dari algoritma genetik Map game.

* BAB VI : MAP GEN. MENGGUNAKAN GENETIC ALGORITHM

Bab ini akan melihat kode – kode yang dipakai untuk membuat algoritma genetik map bersama dengan penjelasan dari masing – masing kode dan mengapa kode tersebut dipakai.

* BAB VII : IMPLEMENTASI GAME

Bab ini akan melihat kode – kode yang dipakai untuk membuat seluruh aspek game yang berhubungan dengan game ini sendiri selain Map Generation, seperti kode gerakan player dan State Machine bot.

* BAB VIII : UJI COBA

Bab ini akan melihat kualitas dari game menggunakan metode – metode yang telah dicantumkan dalam subbab 1.4.4 diatas.

* BAB IX : PENUTUP

Berisi konklusi yang didapat saat membuat game ini dan saran bagi para pembaca yang mungkin bisa membantu.