

**EVALUASI PENERAPAN ALGORITMA A* DAN
ALGORITMA JUMP POINT SEARCH PADA PERMAINAN
LODE-RUNNER, THE LEGEND RETURN**

Skripsi



oleh

ADITYA BIMANTARA

71110091

PROGRAM STUDI TEKNIK INFORMATIKA FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
2015

**EVALUASI PENERAPAN ALGORITMA A* DAN
ALGORITMA JUMP POINT SEARCH PADA PERMAINAN
LODE-RUNNER, THE LEGEND RETURN**

Skripsi



Diajukan kepada Program Studi Teknik Informatika Fakultas Teknologi Informasi
Universitas Kristen Duta Wacana
Sebagai Salah Satu Syarat dalam Memperoleh Gelar
Sarjana Komputer

Disusun oleh

ADITYA BIMANTARA
71110091

PROGRAM STUDI TEKNIK INFORMATIKA FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
2015

PERNYATAAN KEASLIAN SKRIPSI

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul:

EVALUASI PENERAPAN ALGORITMA A* DAN ALGORITMA JUMP POINT SEARCH PADA PERMAINAN LODE-RUNNER, THE LEGEND RETURN

yang saya kerjakan untuk melengkapi sebagian persyaratan menjadi Sarjana Komputer pada pendidikan Sarjana Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Kristen Duta Wacana, bukan merupakan tiruan atau duplikasi dari skripsi keserjanaan di lingkungan Universitas Kristen Duta Wacana maupun di Perguruan Tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Jika dikemudian hari didapati bahwa hasil skripsi ini adalah hasil plagiasi atau tiruan dari skripsi lain, saya bersedia dikenai sanksi yakni pencabutan gelar keserjanaan saya.

Yogyakarta, 15 Juni 2015



ADITYA BIMANTARA

71110091

HALAMAN PERSETUJUAN

Judul Skripsi : EVALUASI PENERAPAN ALGORITMA A* DAN
ALGORITMA JUMP POINT SEARCH PADA
PERMAINAN LODER-RUNNER, THE LEGEND
RETURN

Nama Mahasiswa : ADITYA BIMANTARA

N I M : 71110091

Matakuliah : Skripsi (Tugas Akhir)

Kode : TIW276

Semester : Genap

Tahun Akademik : 2014/2015

Telah diperiksa dan disetujui di
Yogyakarta,
Pada tanggal 6 Mei 2015

Dosen Pembimbing I

Rosa Delima, S.Kom., M.Kom.

Dosen Pembimbing II

Nugroho Agus Haryono, M.Si

HALAMAN PENGESAHAN

EVALUASI PENERAPAN ALGORITMA A* DAN ALGORITMA JUMP POINT SEARCH PADA PERMAINAN LODER-RUNNER, THE LEGEND RETURN

Oleh: ADITYA BIMANTARA / 71110091

Dipertahankan di depan Dewan Penguji Skripsi
Program Studi Teknik Informatika Fakultas Teknologi Informasi
Universitas Kristen Duta Wacana - Yogyakarta
Dan dinyatakan diterima untuk memenuhi salah satu syarat memperoleh gelar
Sarjana Komputer
pada tanggal 11 Juni 2015

Yogyakarta, 19 Juni 2015
Mengesahkan,

Dewan Penguji:

1. Rosa Delima, S.Kom., M.Kom.
2. Nugroho Agus Haryono, M.Si
3. Budi Susanto, SKom.,M.T.
4. Ignatia Dhian E K R, S.Kom, M.Eng

Dekan


(Budi Susanto, S.Kom., M.T.)

Ketua Program Studi


(Gloria Virginia, Ph.D.)

UCAPAN TERIMA KASIH

Puji syukur penulis panjatkan kehadiran Tuhan Yang Maha Esa karena atas berkat dan rahmat-Nya, skripsi yang berjudul “Evaluasi Penerapan Algoritma A* Dan Algoritma Jump Point Search Pada Game Lode-Runner, The Legend Return“ ini dapat terselesaikan.

Penulis menyusun skripsi ini dalam rangka memenuhi salah satu persyaratan untuk mencapai gelar sarjana (S1) pada Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Duta Wacana Yogyakarta.

Penulis menyadari bahwa terselesaikannya Skripsi ini tak lepas dari campur tangan berbagai pihak. Untuk itulah penulis ingin menyampaikan terima kasih kepada:

1. Ibu Rosa Delima, S.Kom., M.Kom., selaku dosen pembimbing I, yang telah banyak memberikan bimbingan selama penyusunan dan penulisan Skripsi ini.
2. Bapak Nugroho Agus Haryono, M.Si., selaku dosen pembimbing II yang juga telah banyak memberikan masukan dan arahan selama pembuatan skripsi.
3. Kepada keluarga terkasih, Bapak, Ibu, dan kakak saya yang memberikan dukungan, doa, nasehat, dan motivasi hingga sampai detik ini penulis menyelesaikan studi.
4. Kepada teman-teman jurusan Teknik Informatika 2011, yang senantiasa ada untuk memberikan dukungan, dan sama-sama berjuang untuk menyelesaikan tugas akhir.
5. Kepada keluarga besar yang senantiasa memotivasi serta selalu mendoakan kelancaran studi hingga Skripsi ini terselesaikan, dan semua keluarga yang tak bisa disebutkan satu per satu.

6. Kepada sahabat-sahabat Program Studi Teknik Informatika 2011, Krisna, Bono, Greg, Graha, Ucup, Verdy, Oscar, Andy, Lyvi, Deo, Dani, Roy, Hagios, Irawan, Okke, Tanu, Hanako dan Silek segala kebersamaan selama ini. Terimakasih atas rasa kekeluargaan yang selalu ada selama masa studi.
7. Terakhir, penulis hendak menyapa setiap nama yang tidak dapat penulis cantumkan satu per satu, terima kasih atas doa yang senantiasa mengalir tanpa sepengetahuan penulis.

Dalam penyusunan tugas akhir ini, tentunya penulis masih memiliki banyak kekurangan pada topik dalam Skripsi ini dan penulisannya yang masih banyak terdapat kekurangan.

Oleh karena itu, penulis sangat menghargai dan menerima jika ada berbagai masukan dari para pembaca baik berupa kritik maupun saran yang sifatnya membangun demi penyempurnaan penulisan-penulisan Skripsi di masa yang akan datang. Penulis meminta maaf bila ada kesalahan dalam penulisan Skripsi ini.

Terima Kasih

KATA PENGANTAR

Puji syukur Penulis Panjatkan ke Hadirat Tuhan Yang Maha Esa karena atas Rahmat dan Karunia-Nya, sehingga penulis dapat menyelesaikan tugas akhir ini.

Dengan selesainya tugas akhir ini tidak lepas dari bantuan banyak pihak yang telah memberikan masukan-masukan kepada penulis. Untuk itu penulis mengucapkan banyak terimakasih.

Penulis menyadari bahwa laporan tugas akhir ini masih jauh dari kesempurnaan baik dari bentuk penyusunan maupun materinya. Oleh karena itu segala kritikan dan saran yang membangun akan penulis terima dengan baik. Akhir kata semoga laporan tugas akhir ini dapat memberikan manfaat kepada kita sekalian.

Yogyakarta, Mei 2015

Penulis

INTISARI

EVALUASI PENERAPAN ALGORITMA A* DAN ALGORITMA JUMP POINT SEARCH PADA GAME LODE-RUNNER, THE LEGEND RETURN

Path finding merupakan sebuah usaha yang dilakukan oleh sebuah *object* untuk mencari jalan dari suatu tempat ke tempat lain. Ada beberapa algoritma yang sering digunakan dalam melakukan *path finding*, di antaranya adalah algoritma A* dan Jump Point Search (JPS). Dalam penelitian ini, penulis melakukan evaluasi terhadap performa algoritma A* dan algoritma JPS yang merupakan pengembangan dari algoritma A*. Evaluasi dilakukan melalui implementasi kedua algoritma kedalam permainan Lode-Runner: The Legend Return dengan mengukur variabel *running-time*, *visited node* dan *open node*. Berdasarkan hasil analisa dari penerapan algoritma terhadap permainan, didapatkan bahwa kedua algoritma memiliki *path* solusi yang sama namun algoritma JPS memiliki jumlah *visited node* dan *open node* yang lebih sedikit.

Kata Kunci: A*, Jump Point Search, Analisis Algoritma, Permainan Lode-Runner

DAFTAR ISI

HALAMAN JUDUL	ii
HALAMAN PERSETUJUAN	iv
HALAMAN PENGESAHAN	v
UCAPAN TERIMA KASIH	vi
KATA PENGANTAR	viii
INTISARI	ix
DAFTAR ISI	x
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xv
BAB I	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah	2
1.4. Tujuan Penelitian	2
1.5. Metodologi Penelitian	3
1.6. Sistematika Penulisan	6
BAB II	7
2.1. Tinjauan Pustaka	7
2.2. Landasan Teori	10
2.2.1. Algoritma A*	10
2.2.2. Algoritma Jump Point Search	12
2.2.3. Lode-Runner, The Legend Return	15
2.3. Contoh Penerapan Algoritma	15

2.3.1. Algoritma A*	16
2.3.2. Algoritma Jump Point Search	18
BAB III	20
3.1. Spesifikasi Perangkat.....	20
3.1.1. Kebutuhan Perangkat keras	20
3.1.2. Kebutuhan Perangkat Lunak.....	20
3.2. <i>Game Design</i>	21
3.2.1. <i>Goal</i>	21
3.2.2. <i>Rule</i>	21
3.2.3. <i>Action</i>	22
3.2.4. <i>Level</i> permainan.....	22
3.2.5. <i>Mapping input device</i>	23
3.2.6. <i>Flowchart</i> Permainan Lode-Runner: The Legend Return	23
3.3. Perancangan Proses Pengembangan Sistem	24
3.4. Algoritma Pencarian dari Agen Cerdas	27
3.4.1. Perancangan Implementasi Algoritma A*	27
3.4.2. Perancangan Implementasi Algoritma Jump Point Search.....	28
3.4.3. Perancangan Cara Kerja Agen Cerdas.....	29
3.5. Perancangan Antar Muka.....	30
3.5.1. <i>Title Screen Page</i>	30
3.5.2. <i>Help Page</i>	30
3.5.3. <i>Option Page</i>	31
3.5.4. <i>Credits Page</i>	32
3.5.5. <i>Game Maps</i>	32
3.6. Perancangan Pengujian Sistem	33

3.6.1. Pengujian Game	33
3.6.2. Pengujian Algoritma	34
BAB IV	36
4.1. Implementasi Sistem	36
4.1.1. Antarmuka Sistem	36
4.1.2. Sistem <i>Map Object</i>	38
4.1.3. Sistem <i>Player Object</i>	39
4.1.4. Sistem agen cerdas	40
4.1.5. Implementasi Algoritma	41
4.2. Analisis Sistem	45
4.2.1. Evaluasi Performa Agen Cerdas Dengan Menggunakan Algoritma A* dan Algoritma Jump Point Search	45
4.2.2. Perbandingan Gerakan Agen Cerdas Algoritma A* Dengan Agen Cerdas Algoritma Jump Point Search	55
BAB V	57
5.1. Simpulan	57
5.2. Saran	57
DAFTAR PUSTAKA	58

DAFTAR GAMBAR

Gambar 2.1 Pseudocode Algoritma A*	11
Gambar 2.2 Contoh pencarian agen cerdas dengan menggunakan algoritma A*. 12	
Gambar 2.3 Pseudocode <i>Identidy successors</i> algoritma JPS	13
Gambar 2.4 Pseudocode Jump function algoritma JPS	13
Gambar 2.5 contoh pencarian agen cerdas dengan menggunakan algoritma Jump Point Search	14
Gambar 2.6 Map permainan Lode-Runner, The Legend Return	15
Gambar 2.7 <i>grid map, start node dan goal node</i>	16
Gambar 2.8 hasil pencarian dengan algoritma A*	18
Gambar 2.9 hasil pencarian dengan algoritma JPS	18
Gambar 3.1 <i>Flowchart</i> alur permainan Lode-Runner: The Legend Return.....	24
Gambar 3.2 Skema proses pengembangan sistem	25
Gambar 3.3 <i>Flowchart</i> algoritma A*	27
Gambar 3.4 <i>Flowchart</i> algoritma Jump Point Search.....	28
Gambar 3.5 <i>Flowchart</i> cara kerja agen cerdas.....	29
Gambar 3.6 Rancangan Halaman Menu	30
Gambar 3.7 Rancangan Halaman bantuan <i>game</i>	31
Gambar 3.8 Rancangan halaman <i>option game</i>	31
Gambar 3.9 Rancangan Halaman <i>credits game</i>	32
Gambar 4.1 Implementasi Halaman Menu	36
Gambar 4.2 Implementasi Halaman bantuan <i>game</i>	37
Gambar 4.3 Implementasi halaman <i>option game</i>	37
Gambar 4.4 Implementasi Halaman <i>credits game</i>	37

Gambar 4.5 Gambar karakter <i>player</i>	46
Gambar 4.6 Gambar karakter agen cerdas	46
Gambar 4.7 Gambar permainan Lode-Runner: The Legend Return dan jalur agen cerdas pada map	46
Gambar 4.8 Gambar <i>map</i> 1 dan jalur agen cerdas pada <i>map</i> 1	47
Gambar 4.9 Gambar <i>map</i> 2 dan jalur agen cerdas pada <i>map</i> 2.....	48
Gambar 4.10 Gambar <i>map</i> 3 dan jalur agen cerdas pada <i>map</i> 3.....	49
Gambar 4.11 Gambar <i>map</i> 4 dan jalur agen cerdas pada <i>map</i> 4.....	50
Gambar 4.12 Gambar <i>map</i> 5 dan jalur agen cerdas pada <i>map</i> 5.....	51
Gambar 4.13 Grafik nilai rata-rata <i>visited node</i>	52
Gambar 4.14 Grafik nilai rata-rata <i>running-time</i>	52
Gambar 4.15 Grafik nilai rata-rata <i>open node</i>	53
Gambar 4.16 <i>Jumping Stright</i>	54
Gambar 4.17 Agen cerdas dengan algoritma A*	56
Gambar 4.18 Agen cerdas dengan algoritma JPS	56
Gambar 4.19 Dua buah agen cerdas yang ditempatkan pada posisi yang sama ...	56
Gambar 4.21 Gambar permainan pada map 3 (2)	56

DAFTAR TABEL

Tabel 2.1 Perbandingan beberapa algoritma <i>path finding</i> berdasarkan waktu eksekusi.....	8
Tabel 2.2 Perbandingan beberapa algoritma <i>path finding</i> berdasarkan <i>visited node</i>	8
Tabel 2.3 Hasil pencarian menggunakan Algoritma A*	17
Tabel 2.4 Hasil pencarian menggunakan Algoritma JPS.....	19
Tabel 3.1 <i>Input device</i> dan keterangan <i>action input device</i>	23
Tabel 4.1 Data penelitian <i>map 1</i>	47
Tabel 4.2 Data penelitian <i>map 2</i>	48
Tabel 4.3 Data penelitian <i>map 3</i>	49
Tabel 4.4 Data penelitian <i>map 4</i>	50
Tabel 4.5 Data penelitian <i>map 5</i>	51
Tabel 4.6 Nilai rata-rata variable pada Ttabel 4.1 hingga 4.5.....	52

INTISARI

EVALUASI PENERAPAN ALGORITMA A* DAN ALGORITMA JUMP POINT SEARCH PADA GAME LODE-RUNNER, THE LEGEND RETURN

Path finding merupakan sebuah usaha yang dilakukan oleh sebuah *object* untuk mencari jalan dari suatu tempat ke tempat lain. Ada beberapa algoritma yang sering digunakan dalam melakukan *path finding*, di antaranya adalah algoritma A* dan Jump Point Search (JPS). Dalam penelitian ini, penulis melakukan evaluasi terhadap performa algoritma A* dan algoritma JPS yang merupakan pengembangan dari algoritma A*. Evaluasi dilakukan melalui implementasi kedua algoritma kedalam permainan Lode-Runner: The Legend Return dengan mengukur variabel *running-time*, *visited node* dan *open node*. Berdasarkan hasil analisa dari penerapan algoritma terhadap permainan, didapatkan bahwa kedua algoritma memiliki *path* solusi yang sama namun algoritma JPS memiliki jumlah *visited node* dan *open node* yang lebih sedikit.

Kata Kunci: A*, Jump Point Search, Analisis Algoritma, Permainan Lode-Runner

BAB I

PENDAHULUAN

1.1. Latar Belakang

Agen cerdas adalah salah satu topik yang rumit dan menarik untuk diteliti. Hal ini terbukti dengan terus meningkatnya kompleksitas kebutuhan agen cerdas dari masa ke masa. Misalnya saja seperti alokasi memori yang terbatas pada CPU, penghitungan gerakan *path finding* secara *real time* yang mungkin terjadi untuk 1 agen atau lebih, tipe-tipe perkembangan *path finding* yang baru seperti *adversarial path finding*, agen cerdas untuk peta yang selalu berubah-ubah informasinya, agen cerdas untuk peta yang tidak semua informasi petanya diketahui dan masih banyak lagi. (Botea, Bouszy, Bruno, Bauckhage, & Nau, 2013)

Performa agen cerdas sangat berpengaruh bagi kepuasan pengguna suatu aplikasi permainan. Algoritma merupakan salah satu faktor yang signifikan untuk memaksimalkan performa dari sebuah agen cerdas. Dengan demikian untuk membuat permainan yang bagus dan memuaskan, pemilihan algoritma untuk agen cerdas haruslah tepat, efektif dan efisien.

Oleh karena itu, penulis ingin meneliti mengenai performa dua buah algoritma, algoritma A* dan algoritma Jump Point Search, yang dirasa mampu untuk memaksimalkan sebuah agen cerdas dalam melakukan *path finding* pada sebuah agen yang bergerak. Algoritma yang digunakan adalah algoritma A* yang sudah terbukti kemampuannya dalam melakukan *path finding* ke sebuah *object* dan algoritma Jump Point Search. Algoritma Jump Point Search adalah algoritma baru dan lebih cepat dibandingkan dengan algoritma A* ketika melakukan *path finding* ke sebuah *object*. (Harabor, Fast Pathfinding via Symmetry Breaking, 2011)

Pada penelitian ini, variabel yang dianalisa oleh penulis adalah performa *running-time*, *open node* dan *visited node* yang dilakukan oleh agen cerdas ketika melakukan *path finding*. Hal tersebut akan penulis implementasikan pada

permainan *Lode-Runner: the Legend Return*, dimana agen cerdas dalam permainan ini memiliki karakteristik yang sama dengan yang diteliti oleh penulis.

1.2. Rumusan Masalah

Rumusan Masalah yang dapat dibuat dalam membantu evaluasi algoritma adalah sebagai berikut:

1. Bagaimana hasil evaluasi dari *running-time*, *open node* dan *Visited Node* dari *path finding* agen cerdas algoritma A* dengan agen cerdas algoritma Jump Point Search ke sebuah *object* dalam permainan Lode-Runner: the Legend Return?
2. Bagaimana perbandingan gerakan yang dilakukan agen cerdas algoritma A* dengan agen cerdas algoritma Jump Point Search?

1.3. Batasan Masalah

Batasan-batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Satu buah Agen cerdas hanya menggunakan satu algoritma pencarian.
2. Percobaan dilakukan dalam kondisi yang sama.
3. *Running-time* diukur dengan menghitung lama waktu agen cerdas melakukan *path finding* yang dimulai ketika agen cerdas melakukan pencarian jalan hingga menemukan jalan pada *object* yang dituju.

1.4. Tujuan Penelitian

Tujuan penelitian ini adalah mengetahui evaluasi kinerja dari algoritma A* dan Jump Point Search dengan membandingkan hasil *path finding* berupa data *open node*, *visited node* dan *running-time* agen yang melakukan pencarian jalan menuju satu buah *object*.

1.5. Metodologi Penelitian

a. Studi literatur

Penulis memulai penelitian dengan mengumpulkan dan mempelajari bahan-bahan referensi tentang teori algoritma dari berbagai sumber, seperti buku, jurnal, maupun situs internet.

b. Membuat *game design*

Desain permainan yang dirancang oleh penulis adalah permainan *Lode-Runner, the Legend Return* yang nantinya digunakan sebagai demo implementasi algoritma.

c. Membuat Sistem

Penulis mulai membuat sistem dengan memulai dari mempersiapkan *sprite* gambar karakter, *background*, *map*, *tree node* dan *code engine* dari permainan tersebut.

d. Implementasi algoritma pada agen cerdas dan percobaan

Sistem yang sudah dibuat kemudian ditambahkan algoritma A* dan algoritma Jump Point Search ke dalam *object* agen cerdas.

e. Evaluasi Sistem

Evaluasi terhadap program dilakukan dengan **membandingkan *sample data* dari kedua algoritma** (algoritma A* dan algoritma Jump Point Search) yang diperoleh dari percobaan yang dilakukan.

- Data yang diteliti

Data *running-time* merupakan lama waktu agen cerdas melakukan pencarian jalan sampai menemukan jalan menuju *player*. Data tersebut didapatkan dengan mencatat selisih waktu agen cerdas sebelum melakukan pencarian dan waktu agen cerdas ketika mendapatkan jalan menuju *player* (menggunakan algoritma JPS dan A*).

Data *Open node* merupakan jumlah *node* yang dimasukkan ke dalam *open list* oleh agen cerdas ketika pertama kali melakukan pencarian jalan pada *player*. Data ini dihitung berdasarkan asumsi

bahwa posisi agen cerdas dan posisi *player* tetap/tidak bergerak. *Node* yang dimasukkan kedalam *open list* merupakan *start node*, hasil ekspansi dari *start node* hingga mencapai *player* (algoritma A*) dan hasil ekspansi yang memenuhi kondisi *forced neighbor* untuk menjadi *jump point* dari *start node* hingga mencapai *player* (Algoritma JPS).

Data *visited node* merupakan jumlah *node* yang diekspan oleh agen cerdas ketika pertama kali melakukan pencarian jalan menuju *player*. Data ini dihitung berdasarkan asumsi bahwa posisi agen cerdas dan posisi *player* tetap/tidak bergerak. *Node* yang diekspan merupakan *node* yang terdapat pada *open node* dan dapat diekspan dengan syarat *node* tersebut harus memiliki nilai $f(n)$ terendah pada *open node*, apabila *node* yang tersisa memiliki nilai yang sama, maka *node* yang diekspan merupakan *node* yang berada pada posisi paling atas pada *open List*. *Open List* menggunakan struktur data **binary heap** dimana setiap *node* akan diurutkan berdasarkan *node* yang memiliki nilai $f(n)$ paling rendah ke *node* yang memiliki $f(n)$ paling tinggi (menggunakan algoritma JPS dan A*)

- Percobaan untuk mendapatkan data

Percobaan dilakukan dengan membuat *variable counter* untuk mengukur jumlah *visited node* dan jumlah *open node*. Data *running-time* didapatkan dengan mencatat selisih waktu ketika agen cerdas akan melakukan pencarian dan ketika agen cerdas telah selesai melakukan pencarian.

Peneliti melakukan percobaan sebanyak 50 kali. 1 kali percobaan dilakukan dengan menaruh agen cerdas algoritma A* dan agen cerdas algoritma JPS pada posisi yang sama secara *random* lalu kedua algoritma tersebut melakukan pencarian jalan menuju *object* yang sama yaitu *player*. Percobaan dilakukan dalam kondisi *player* yang statis dan tidak bergerak, hal ini disebabkan karena ketika *player* bergerak, pohon pencarian akan mengalami perubahan ketika posisi *player/goal node* berubah.

- Validitas data

Data yang diambil oleh penulis akan divalidasi kembali. Proses validasi tersebut menghasilkan data yang valid apabila kondisi dari setiap percobaan (percobaan ketika menggunakan algoritma Jump Point Search dan algoritma A*) berada dalam kondisi yang sama. Yang dimaksud dengan kondisi yang sama adalah untuk setiap posisi *player*, posisi agen cerdas dan posisi *node* pada *graph* berada dalam posisi yang sama namun algoritma dari agen cerdas menggunakan salah satu dari algoritma Jump Point Search atau algoritma A*.

- Jumlah *sample* data

Jumlah *sample* data yang diambil dari percobaan adalah 50 data. Data tersebut mewakili algoritma Jump Point Search dan algoritma A* dimana data harus sudah valid. Data yang berjumlah 50 diambil dari 5 *map* yang berberda dimana 1 *map* akan diambil 10 data.

- Analisis dan Evaluasi

Setelah mendapatkan data valid dari percobaan yang dilakukan, data tersebut akan dianalisa dan dievaluasi. Analisa tersebut mengupas tentang rumusan masalah yang dihadapi oleh penulis. Misalnya saja seperti bagaimana hasil evaluasi *running-time*, *open node* dan *Visited Node* dari *path finding* agen cerdas algoritma A* dengan agen cerdas algoritma Jump Point Search ke sebuah *object* dalam permainan *Lode-Runner: the Legend Return* dan bagaimana perbandingan gerakan yang dilakukan agen cerdas algoritma A* dengan agen cerdas algoritma Jump Point Search.

1.6. Sistematika Penulisan

Sistematika penulisan laporan Tugas Akhir ini dikelompokkan menjadi 5 bab, yaitu:

Bab I Pendahuluan. Bab ini berisi gambaran tentang penelitian yang dilakukan. Bab ini terbagi menjadi beberapa bagian, yaitu latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, metode penelitian dan sistematika penulisan.

Bab II Tinjauan Pustaka. Bab ini berisi tentang teori yang menjadi landasan dalam pembuatan penelitian. Teori-teori tersebut diambil dari jurnal penelitian, maupun sumber-sumber lain yang dapat mendukung penelitian. Dasar teori dan tinjauan pustaka ini menjadi acuan dalam pembuatan analisis data dan pengambilan kesimpulan penelitian.

Bab III Analisis dan Perancangan Sistem. Bab ini menjelaskan mengenai tahap perancangan program aplikasi permainan *Lode Runner, The Legend Return* serta penerapan algoritma A* dan algoritma Jump Point Search pada agen cerdas permainan. Bab ini juga menjelaskan mengenai desain permainan yang berlaku dalam permainan *Lode Runner, The Legend Return*.

Bab IV Implementasi dan Analisis Sistem. Bab ini berisi tentang penjelasan hasil penelitian dan analisa berkaitan dengan masalah yang ada. Penjelasan disajikan dalam bentuk deskripsi, tabel, grafik, dan gambar pendukung.

Bab V Kesimpulan dan Saran. Bab ini berisi kesimpulan dari penelitian yang dilakukan beserta saran pengembangan sistem untuk penelitian yang memiliki topik yang sama dengan topik penelitian ini.

BAB V

KESIMPULAN DAN SARAN

5.1. Simpulan

Berdasarkan hasil implementasi dan analisis sistem, maka dapat disimpulkan bahwa:

- a. Performa Algoritma A* dan Algoritma Jump Point Search dalam melakukan pencarian jalan pada satu buah *object* memiliki performa yang sama optimalnya apabila dilihat dari segi hasil panjang *path* hasil pencarian jalan.
- b. Algoritma Jump Point Search melakukan *pruning* terhadap *node* yang terletak diantara *current node* dan persimpangan *node*. Hal ini mengakibatkan jumlah *open node* dari Algoritma Jump Point Search 3 kali lebih sedikit dari algoritma A* dan jumlah *visited node* algoritma Jump Point Search yang 4 kali lebih sedikit dari algoritma A*.
- c. Algoritma A* dan algoritma Jump Point Search memiliki hasil *path* yang sama ketika melakukan pencarian jalan, meski begitu, agen cerdas yang menggunakan algoritma tersebut tetap akan mengalami perbedaan gerakan yang disebabkan arah dari hasil pencarian jalan algoritma tersebut tidak sama.

5.2. Saran

Melalui penelitian yang dilakukan oleh penulis, penulis memberikan saran untuk pengembangan selanjutnya, diantaranya adalah:

- a. Pembuatan program dilakukan dengan *platform mobile*, hal ini disarankan untuk mengetahui *running-time* yang lebih akurat apabila program dijalankan dengan *memory smartphone* yang terbatas.
- b. Agen cerdas dibuat menjadi lebih pintar dengan bisa mengetahui *environment* disekitar agen. Salah satu contohnya adalah agen dapat menghindari rintangan apabila agen mendapat prediksi resiko yang merugikan ketika melakukan pencarian jalan.

DAFTAR PUSTAKA

- Botea, A., Bouszy, B., Bruno, M., Bauckhage, C., & Nau, D. (2013). Pathfinding in Games. *Artificial and Computational Intelligence in Games*, 1-3.
- Cui, X., & Shi, H. (2011). A*-based Pathfinding in Modern Computer Games. *IJCSNS International Journal of Computer Science and Network Security*, 1-6.
- Dolinskaya, I. S. (2012). OptimalPath NRL Dolinskaya. *Optimal Path Finding in Direction, Location and Time Dependent*, 17.
- Harabor, D. (2011). Fast Pathfinding via Symmetry Breaking. *Fast Pathfinding via Symmetry Breaking*, 1-9.
- Harabor, D., & Grastien, A. (2013). Online Graph Pruning for Pathfinding on Grid Maps. 1-6.
- Kilincerlar, M. (2007). Implementation Of A Path Finding Algorithm For The Navigation Of Visually Impaired People. 62.
- Krishnaswamy, N. (2009). A Comparison of Efficiency in Pathfinding Algorithms in Game Development. 1-25.
- Miroslav, K. (2012). Any-angle path-planning algorithms. 12.
- Mott, T. (2010). *1001 Video Games: You Must Play Before You Die*. London: Quintessence.
- Nosrati, M., Karimi, R., & Hasanvand, H. A. (2011). Investigation of the * (Star) Search Algorithms: Characteristics, Methods and Approaches. *World Applied Programming*, Vol (2), No (4), April 2012. 251-256, 1-3.
- Zarembo, I., & Kodors, S. (2013). Pathfinding Algorithm Efficiency Analysis in. *Proceedings of the 9th International Scientific and Practical Conference*, Volume 11, 1-5.