

PROTOTIPE GAME MAZE CHASER DENGAN ALGORITMA A*

Aditya Haryanov¹⁾

¹⁾Teknik Informatika STMIK AMIKOM Yogyakarta
Jl Ring road Utara, Condongcatur, Sleman, Yogyakarta 55281
Email : adityaharyanov@gmail.com¹⁾

Abstrak

Perkembangan dunia game semakin pesat. Hal ini diikuti dengan semakin bertambahnya jenis game. Salah satunya Casual Game / Game Casual.

Game Casual adalah game yang sedang populer di kalangan remaja sekarang. Semenjak boomingnya beberapa jenis game casual yang akhirnya mulai banyak dilirik game player terutama player dari Indonesia. Sudah banyak konsep konsep non teknologi yang telah diimplementasikan ke dalam sebuah game. Salah satunya adalah konsep kecerdasan buatan atau A.I. Kecerdasan Buatan ialah membuat mesin bekerja seperti manusia. Sekarang sudah ada berbagai macam algoritma kecerdasan buatan yang bisa digunakan tergantung dari tujuan dan fungsinya. Salah satunya Algoritma A. Algoritma A* adalah algoritma kecerdasan buatan untuk mencari jalur terpendek dari suatu titik ke titik lainnya.*

Berdasarkan masalah diatas, disini penulis mencoba membuat sebuah game casual yang mungkin masih jarang jenisnya di pasaran yaitu jenis maze runner yaitu player menghindari musuh didalam sebuah labirin. Di game ini penulis akan menerapkan konsep Kecerdasan Buatan dengan Algoritma A yang diharapkan membuat game ini menjadi lebih interaktif dan dinamis dengan musuh yang bisa berfikir / mengambil keputusan sendiri.*

Kata kunci: *Casual Games, Running, Maze, A.I*

1. Pendahuluan

Perkembangan dunia game pada saat ini sudah semakin pesat. Hal ini diikuti oleh semakin banyaknya jenis dari tiap game yang dipublikasikan. Berawal dari sebatas tampilan 16 bit, dengan sistem permainan yang sederhana. Kini kita sudah dapat menikmati game dengan tampilan poligon dan dengan tipe permainan sesuai keinginan kita, seperti shooter, RPG, action, simulated strategy, dan masih banyak yang lain.

Dari sekian banyak jenis game yang beredar di pasaran saat ini, game casual adalah genre game yang bisa dibiliang sedang trend saat ini dengan meningkatnya produktivitas game casual entah dalam mobile / desktop.

Dan dalam casual game itu sendiri ada banyak jenisnya. Ada jenis shooter, side-scrolling, running, dll. Dan ada juga jenis maze runner. Yaitu player menghindari / lari dari musuh yang bertempat di sebuah labirin. Bisa dilihat jika player harus menghindari musuh berarti musuh harus bisa berfikir sendiri atau memiliki A.I / Kecerdasan Buatan.

Kecerdasan buatan sendiri ada banyak jenis algoritmanya dengan berbagai fungsinya. Salah satunya adalah Algoritma A* yaitu sebuah algoritma kecerdasan buatan yang berfungsi untuk pathfinding / mencari jalur tercepat dari start pointnya ke target.

Dari penjelasan diatas penulis mencoba menghubungkan Algoritma A* dengan membuat sebuah casual game berjenis maze runner. Yang nantinya diharapkan bisa membuat game menjadi lebih dinamis dan membuat player menjadi lebih tertarik lagi.

2. Pembahasan

2.1 Kecerdasan Buatan

Kecerdasan buatan (*Artificial Intelligence*) ialah membuat mesin bekerja seperti manusia. Beberapa ahli mengemukakan definisi kecerdasan buatan (*Artificial Intelligence*), yaitu sebagai berikut:

John McCarthy [1956]: Kecerdasan buatan (*Artificial Intelligence*) ialah Memodelkan proses berpikir manusia dan mendesain mesin agar menirukan perilaku manusia.

H. A. Simon [1987]“ Kecerdasan buatan (*Artificial Intelligence*) ialah sebuah tempat sebuah penelitian, aplikasi dan instruksi yang terkait dengan pemrograman komputer untuk melakukan sesuatu hal yang -dalam pandangan manusia adalah- cerdas”

Rich and Knight [1991]: “Kecerdasan Buatan (*Artificial Intelligence*) ialah sebuah studi tentang bagaimana membuat komputer melakukan hal-hal yang pada saat ini dapat dilakukan lebih baik oleh manusia.”

Encyclopedia Britannica: “Kecerdasan Buatan (Artificial Intelligence) ialah cabang dari ilmu komputer yang dalam merepresentasi pengetahuan lebih banyak menggunakan bentuk simbol-simbol daripada bilangan, dan memproses informasi berdasarkan metode heuristic atau dengan berdasarkan sejumlah aturan”.

Dari sekian banyak jenis algoritma kecerdasan buatan. Setidaknya ada lima jenis kecerdasan buatan yang sering kita temui, yaitu :

1. **Jaringan Syaraf Buatan (Artificial Neural Networks),**
 Merupakan sekelompok jaringan saraf (*neuron*) buatan yang menggunakan model matematis atau komputasi untuk pemrosesan informasi berdasarkan pendekatan terhubung pada komputasi. Pada kebanyakan kasus, JST merupakan sistem adaptif yang merubah strukturnya berdasarkan informasi eksternal maupun internal yang mengalir melalui jaringan tersebut.
2. **Logika Fuzzy (Fuzzy Logics)**
3. **Algoritma Genetik (Genetic Algorithms),**
 Algoritma Genetik biasanya digunakan dibidang kedokteran, misal untuk menganalisis DNA.
4. **Robotika (Robotics),**
 AI ini banyak digunakan di pabrik. Biasanya dibuat untuk melakukan kegiatan otomatisasi, misal dalam PLC (Programmable Logic Contro
5. **Permainan Komputer (Games),**
 AI jenis ini yang paling disukai oleh anak-anak, misal untuk memainkan game Age of Mythology atau Counter Strike. Salah satu Algoritma yang bisa dipakai di Games adalah Algoritma A*.

2.2 Algoritma A*

Algoritma A*, dapat juga disebut sebagai Algoritma A Star, merupakan salah satu contoh algoritma pencarian yang cukup populer di dunia. Jika kita mengetikkan Algoritma A* pada sebuah mesin pencari, seperti google.com, maka akan kita temukan lebih dari sepuluh ribu literatur mengenai algoritma A*.

Beberapa terminologi dasar yang terdapat pada algoritma ini adalah starting point, simpul (nodes), A, open list, closed list, harga (cost), halangan (unwalkable).

Starting point adalah sebuah terminologi untuk posisi awal sebuah benda.

A adalah simpul yang sedang dijalankan dalam algoritma pencarian jalan terpendek.

Node adalah petak-petak kecil sebagai representasi dari area pathfinding. Bentuknya dapat berupa persegi, lingkaran, maupun segitiga.

Open list adalah tempat menyimpan data simpul yang mungkin diakses dari starting point maupun simpul yang sedang dijalankan.

Closed list adalah tempat menyimpan data simpul sebelum A yang juga merupakan bagian dari jalur terpendek yang telah berhasil didapatkan.

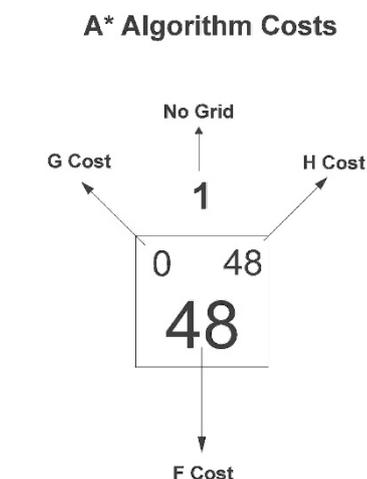
Harga / Cost adalah suatu nilai yang akan diolah dalam algoritma A*. Ada 3 jenis Cost dalam algoritma A* yaitu :

1. **G_Cost**
 adalah biaya yang sudah dikeluarkan dari keadaan awal sampai keadaan n.
2. **H_Cost**
 adalah estimasi biaya untuk sampai pada tujuan mulai dari n.
3. **F_Cost**
 adalah nilai hasil penjumlahan dari G_Cost dan H_Cost

Target Node yaitu node yang dituju.

Halangan adalah sebuah atribut yang menyatakan bahwa sebuah simpul tidak dapat dilalui oleh A.

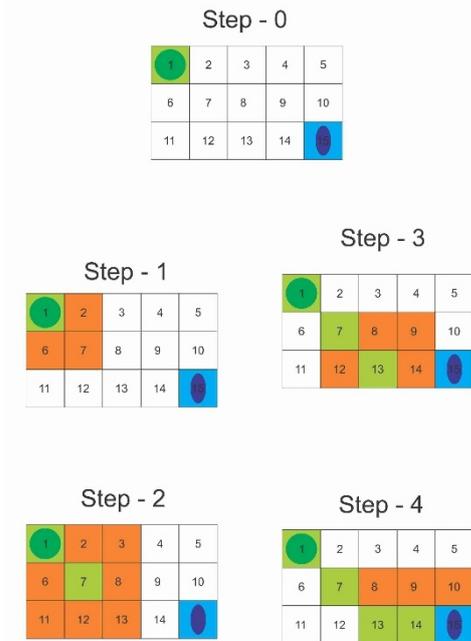
Prinsip algoritma ini adalah mencari jalur terpendek dari sebuah simpul awal (starting point) menuju simpul tujuan dengan memperhatikan F_Cost terkecil dan jika ada lebih dari 1 F_Cost yang memiliki nilai terkecil maka akan dilanjut memerhatikan nilai H_Cost yang terkecilnya^[1]



Gambar 1. A* Algorithm Basic

Gambar diatas adalah penjelasan dasar mengenai Cost dalam Algoritma A*. Di kiri atas adalah nilai dari G_Cost, kanan atas adalah nilai dari H_Cost, dan yang di tengah adalah nilai dari F_Cost. Angka di atas kotak adalah letak lokasi grid dari node tersebut.

Dibawah ini saya menyiapkan sebuah mini contoh kasus dari prototype game "MAZE CHASER".

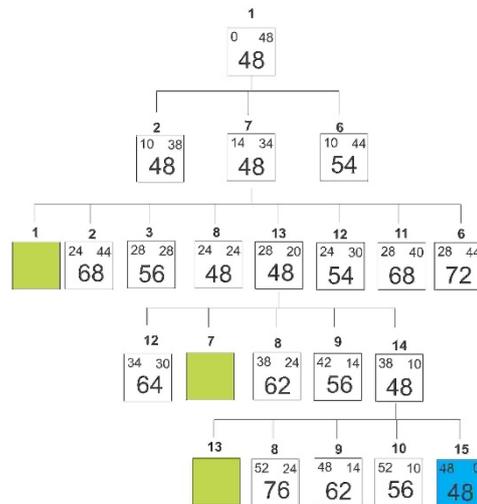


Gambar 2 Contoh Kasus

Di gambar diatas dapat dilihat step dalam pengambilan keputusan jalur untuk bisa mencapai target.

Dan dibawah ini adalah decision tree untuk kasus diatas

A* Algorithm Decision Tree



Gambar 3. A* Algorithm Decision Tree

Dari gambar diatas dapat kita lihat setiap pengambilan keputusan dilihat dari nilai F_Cost terkecil. Kecuali pada Step 1. Karena ada 2 buah grid yang memiliki nilai F_Cost terkecil yaitu grid 2 dan 7. Maka keutusan diambil dengan melihat nilai H_Cost yang terkecil.

Di bawah adalah pseudo-code program Algoritma A* yang digunakan.

PSEUDO-CODE

```
Membuat OPEN List
Membuat CLOSED List

begin

memasukkan start node ke OPEN List

loop
current = node di OPEN List dgn nilai f_cost terkecil
hapus current dari OPEN List
menambahkan current ke CLOSED List

if current adalah target node
return
end if

foreach neighbour dari current node
if neighbour ada di CLOSED List
skip ke neighbour selanjutnya
end if

if path baru ke neighbour lebih pendek or neighbour tidak berada di OPEN List
set f_cost dari neighbour
set parent dari neighbour menjadi current

if neighbour tidak berada di OPEN List
tambahkan neighbour ke OPEN List
end if
end foreach
end loop

end
```

Gambar 4. Pseudo-code Algoritma A*

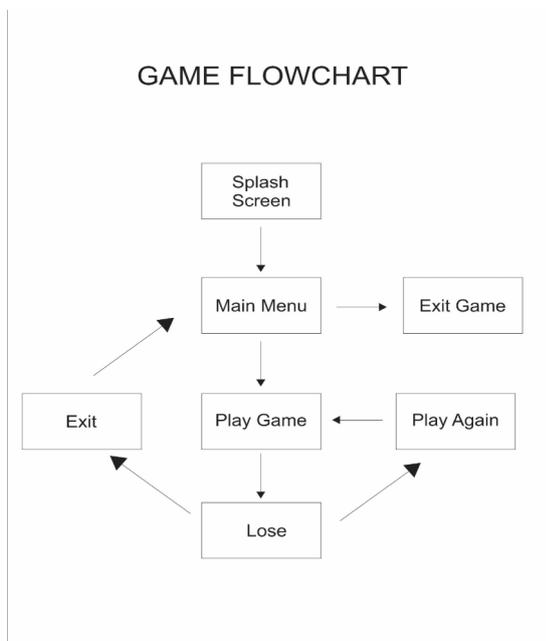
2.2 Casual Game

Menurut pengertian Fullerton (2008,p420) *Casual Game* adalah sebuah *game* yang dimaksudkan untuk dinikmati oleh setiap orang, baik pria dan wanita,maupun tua dan muda. Dengan ini *casual game* dapat diartikan sebuah *game* yang jauh dengan permainan hentakan, kekerasan, dan kompleksitas permainan dengan bertujuan untuk menarik pemain seluas mungkin dari berbagai kalangan.^[2]

2.3 Tahap Perancangan

a. Alur / Flowchart

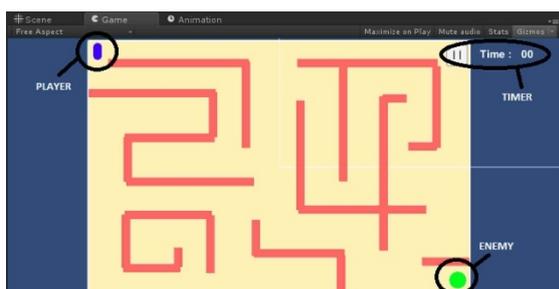
Tahap ini bertujuan untuk merancang konsep dan alur game.



Gambar 5. Flowchart Game

b. Perancangan Assets

Tahap ini adalah membuat dummy asset untuk me representasikan player, enemy, dinding labirin, dan ground. Dummy Asset di buat dengan software CorelDraw X6 dan di olah di Unity3D.



Gambar 6. Dummy Asset Game View



Gambar 7. Dummy Asset Main Menu View

2.4 Tahap Scripting

Tahap ini adalah tahap menyusun script untuk controlling player dan memasukkannya ke objek player, dan menyusun algoritma A* dan memasukkannya ke objek enemy. Dan menyusun script untuk time dan score.

2.5 Tahap pengujian

Tahap ini adalah tahap untuk menguji apakah semua komponen game sudah berjalan dengan baik tanpa ada error atau kesalahan. Hasil pengujian bisa dilihat di tabel berikut.

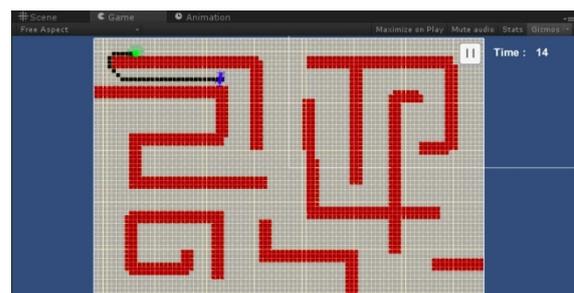
Tabel 1 Hasil Pengujian Game MAZE CHASER

Assets	OK
Player Control	OK
Enemy AI	OK
Implementing Obstacle	OK
Pause	OK
Lose Condition	OK
Main Menu Start	OK
Main Menu Exit	OK

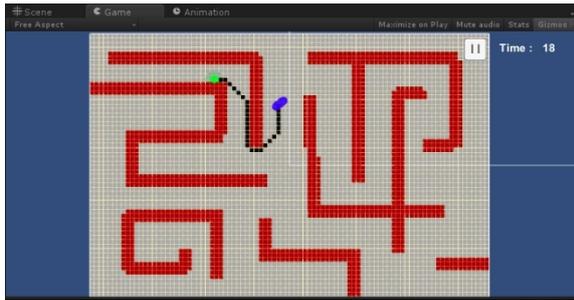
3. Kesimpulan

Game berjalan dengan baik. Musuh menjadi lebih interaktif dan dinamis karena dengan Algoritma A* dia bisa menentukan keputusan dari aturan-aturan yang saya buat dan game pun terasa semakin menyenangkan dan menarik.

Beberapa Sample Screenshot saat game berlangsung :



Gambar 8. Sample 1



Gambar 9. Sample 2

Bisa dilihat bahwa Musuh bisa mencari jalur menuju player secara dinamis walaupun player terus bergerak.

Daftar Pustaka

- [1] Victor, Hamidi Ivan, Aldillah Diko. "Algoritma A* (A Star) Sebagai Salah Satu Contoh Metode Pemrograman Branch and Bound". <http://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/Makalah/MakalahStmik24.pdf>, 26 November 2015
- [2] Fullerton, T. 2008. Game Design Workshop, Second Edition: A Playcentric Approach to Creating Innovative Games (Gama Network Series) Morgan Kaufmann
- [3] Anonim, 2015. Pengertian Kecerdasan Buatan Menurut Para Ahli. <http://www.pendidikanmu.com/2015/04/pengertian-kecerdasan-buatan-menurut-para-ahli.html>

Biodata Penulis

Aditya Haryanov, saat ini masih berstatus mahasiswa aktif STMIK AMIKOM YOGYAKARTA. Belum mendapat gelar apapun

