

Aplikasi Brute Force dan Depth First Search dalam Menyelesaikan Puzzle pada Permainan OneShot

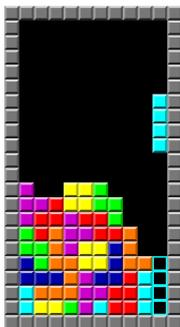
Andrianata Putra Tjandra 13519147
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): 13519147@std.stei.itb.ac.id

Abstract—OneShot adalah permainan berbasis puzzle dimana pemain harus menjelajahi suatu map untuk mendapat potongan puzzlenya. Untuk menyelesaikannya, pemain harus menggabungkan beberapa potongan dengan potongan lainnya dan berinteraksi dengan suatu tempat pada map tersebut. Pemain dapat memanfaatkan algoritma Brute Force dan DFS atau Depth First Search dalam menyelesaikan puzzle.

Keywords—OneShot, Puzzle, DFS, Brute Force(key words)

I. PENDAHULUAN

Permainan telah lama berada di dunia ini. Ada banyak jenis permainan seperti permainan papan seperti catur, permainan lapangan seperti sepak bola, dan salah satu yang paling sering dimainkan adalah permainan elektronik. Permainan ini dimainkan pada gawai seperti komputer, smartphone, atau gawai khusus permainan elektronik seperti Playstation. Permainan elektronik sering disebut sebagai game di kehidupan sehari-hari. Game memiliki genre layaknya musik dan salah satu genrenya adalah puzzle dimana pemain harus menyelesaikan sebuah masalah.



Gambar 1.1 Salah satu game puzzle, Tetris
Sumber: <https://id.wikipedia.org/wiki/Tetris>

OneShot adalah salah satu game pada komputer yang terkenal karena desain gamenya yang unik dan memiliki cerita yang menarik. Permainan ini banyak memanfaatkan *Fourth Wall Breaking*, yaitu kondisi media, baik film atau game, yang seharusnya memiliki dunianya sendiri, yaitu di dalam film atau game itu sendiri, melakukan sesuatu yang mengindikasikan mereka dalam sebuah simulasi dan ada dunia luar, yaitu dunia kita sendiri. Selain itu, inti permainan game ini adalah puzzle yang harus diselesaikan untuk dapat melanjutkan ceritanya.

Untuk dapat menyelesaikan puzzle ini, dapat dimanfaatkan dua buah algoritma, yaitu Brute Force dan Depth First Search atau DFS.

II. LANDASAN TEORI

A. Algoritma Brute Force

Brute Force adalah algoritma yang paling dasar dari semua strategi algoritma. Inti dari algoritma ini adalah mencoba semua kemungkinan yang ada dari sebuah masalah sampai ditemukan solusi. Algoritma ini pasti akan menemukan solusi dari masalah tersebut namun biasanya membutuhkan waktu yang sangat lama. Adapun cara untuk mengurangi waktu Brute Force dengan mengaplikasikan teknik Heuristik, yaitu teknik yang juga memperhitungkan intuisi, terkaan, atau akal sehat, dengan mengabaikan kemungkinan yang pasti bukan solusi.

Contoh penggunaan Brute Force adalah pada permainan Kubus ajaib. Permainan ini mengharuskan pemain untuk menyusun angka 1-9 dalam kotak 3x3 dan jumlah baris, kolom, dan diagonalnya sama dengan 15. Penggunaan Brute Force akan memeriksa setiap kemungkinan, yaitu 9! buah kemungkinan, apakah syarat dipenuhi.

8	1	6	6	1	8	4	3	8	2	7	6
3	5	7	7	5	3	9	5	1	9	5	1
4	9	2	2	9	4	2	7	6	4	3	8
(1)			(2)			(3)			(4)		
2	9	4	4	9	2	6	7	2	8	3	4
7	5	3	3	5	7	1	5	9	1	5	9
6	1	8	8	1	6	8	3	4	6	7	2
(5)			(6)			(7)			(8)		

Gambar 2.1 Kubus Ajaib

Sumber:

[https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Brute-Force-\(2021\)-Bag2.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Brute-Force-(2021)-Bag2.pdf)

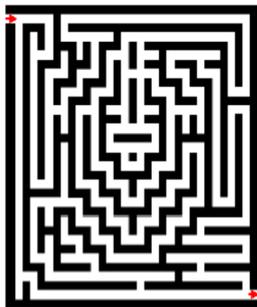
Penggunaan Brute Force pada Kubus ajaib tersebut dapat dibuat lebih sederhana menggunakan Heuristik. Salah satu Heuristik yang dapat digunakan adalah jika pada baris pertama

tidak sama dengan 15, maka sudah pasti bukan dan dapat lanjut ke pemeriksaan selanjutnya

B. Algoritma Depth First Search (DFS)

DFS merupakan salah satu dari algoritma traversal graf dengan algoritma lainnya adalah BFS, A*, DLS, IDS, dan UCS. DFS adalah algoritma pencarian pada sebuah graf terhubung yang cara traversalnya dilakukan secara mendalam. Pencarian akan dimulai pada simpul awal dan ada simpul lain yang ingin dicapai. Untuk setiap simpul tetangga, akan ditulis simpul tersebut dalam sebuah queue. DFS memanfaatkan queue LIFO atau stack, yaitu queue anggota yang masuk terakhir akan keluar paling pertama.

Contoh penggunaan DFS adalah pada penyelesaian labirin atau *Maze*. *Maze* mengharuskan pemain untuk mencapai jalan keluar dari titik awal namun akan ada banyak kemungkinan jalan dan beberapa darinya adalah jalan buntu.



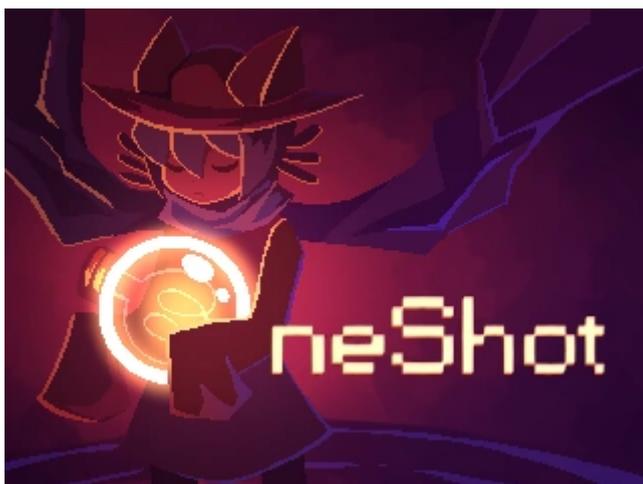
Gambar 2.2 Maze

Sumber:

[https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Brute-Force-\(2021\)-Bag2.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Brute-Force-(2021)-Bag2.pdf)

C. Game OneShot

OneShot adalah game puzzle dan adventure 2D yang dirilis pada tanggal 8 Desember 2016 di platform Windows. Game ini dibuat oleh Future Cat dan dipublikasi oleh Degica.



Gambar 2.3 OneShot

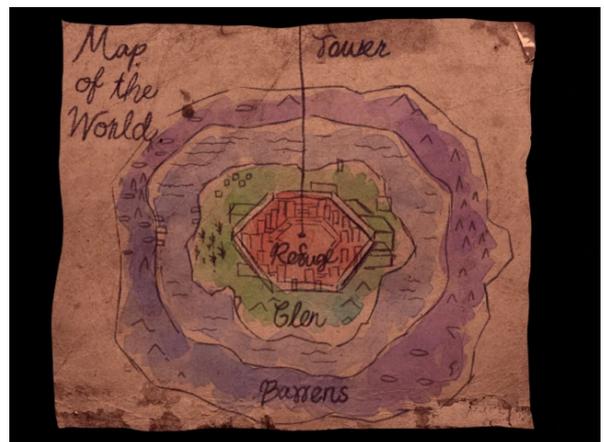
Sumber: <https://imgur.com/gallery/arBpM>

Di game ini, pemain akan mengontrol sebuah karakter bernama Niko di sebuah dunia lain dan Niko dengan bantuan pemain akan menyelamatkan dunia tersebut dari kegelapan dengan mengembalikan matahari di sebuah menara. Elemen utama dalam permainan ini adalah cerita bagaimana Niko dapat menyelesaikan misinya tersebut sambil pemain memandu apa yang harus dilakukannya dan interaksi pemain dengan Niko menggunakan *Fourth Wall Breaking*.

Dalam prosesnya, pemain harus menyelesaikan puzzle dengan mencari barang kunci. Beberapa barang kunci dapat digabungkan barang kunci lain menjadi sebuah barang kunci baru atau mengubah barang kunci yang ada. Barang kunci dapat digunakan untuk berinteraksi dengan suatu entitas pada area untuk membuka akses ke tempat lain pada area tersebut. Proses ini dilakukan berulang sampai Niko keluar dari area tersebut. Beberapa objek mungkin memiliki puzzlenya sendiri lagi dan pemain harus menyelesaikannya untuk mendapat akses atau mendapat barang darinya.

Beberapa puzzle membutuhkan pencarian ke luar game sebagai estetika dari *Fourth Wall Breaking* pada game ini. Luar game meliputi folder Documents pada komputer atau wallpaper desktop.

Game ini terbagi menjadi tiga area besar, yaitu Barren, Glen, dan Refuge. Tiap area ini terbagi menjadi beberapa area yang lebih kecil lagi seperti rumah-rumah atau daerah bawah tanahnya. Setiap area didesain sedemikian rupa sehingga tujuan pasti dapat dicapai dari titik awal dan area besar tidak terhubung dengan area besar lainnya.



Gambar 2.4 Peta pada OneShot

Sumber:

https://oneshot.fandom.com/wiki/The_World?file=Cg_map.png

Game ini memiliki fitur inventory, yaitu Niko memiliki sejumlah barang yang diambil dari lingkungannya dan dapat memegang salah satu barang dari yang dimilikinya. Niko dapat memegang satu benda dengan cara membuka menu dan mengkonfirmasi sebuah benda. Beberapa barang juga dapat digabungkan dengan barang lainnya dan menghasilkan barang baru. Dengan cara ini, Niko dapat menggunakan suatu benda

pada suatu entitas tertentu yang membutuhkan sebuah barang spesifik.



Gambar 2.5 Penggabungan Dua Barang
Sumber: dokumentasi penulis

OneShot versi ini merupakan ekspansi dari versi gratis yang dirilis pada tahun 2014. Pada versi 2014, pemain hanya dapat memainkan game ini satu kali saja seperti yang judulnya gambarkan, yaitu OneShot yang berarti satu kali coba. Selain itu, akhir dari versi gratis ini berbeda dari versi yang baru dan pada versi baru ada kelanjutan juga setelah akhir dari yang seharusnya.

III. PENERAPAN ALGORITMA PADA ONESHOT

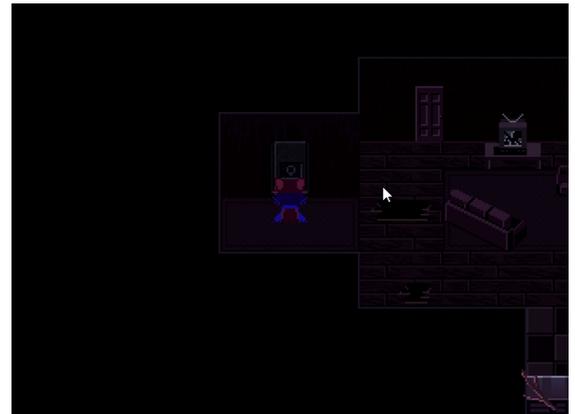
Objektif setiap area adalah untuk mencapai area selanjutnya. Area akan dibagi menjadi beberapa bagian area dengan entitas masing-masing. Entitas mungkin berupa NPC (Non-Playable Character atau karakter komputer) atau benda mati dan beberapa yang menonjol dapat dilakukan interaksi. Area yang dikunjungi akan menjadi sebuah graf yang simpulnya merepresentasikan sebuah bagian area dan sisi menunjukkan bagian area tersebut terhubung ke bagian area lain.

Saat melakukan interaksi, mungkin untuk suatu entitas memerlukan sebuah barang kunci untuk membuka jalan baru menuju objektif. Pemain dapat membuat Niko memegang suatu barang kunci untuk dicoba pada entitas tersebut.

Untuk percobaan kali ini, akan diuji pada bagian paling awal permainan, yaitu saat Niko baru saja bangun dari tidurnya sampai Niko keluar rumah. Strategi yang akan digunakan adalah sebagai berikut:

- Selama masih berada di area, kunjungi seluruh bagian area yang dapat ditelusuri (mungkin ada area yang dikunci untuk saat ini). Jika ada lokasi yang membutuhkan sebuah barang kunci dan dari penelusuran sebelumnya didapat sebuah barang kunci baru, utamakan untuk ke lokasi tersebut menggunakan DFS. Jika sudah berada pada area lain, maka tujuan tercapai.
- Untuk setiap bagian area yang dikunjungi, interaksi dengan entitas yang menonjol. Entitas yang menonjol

adalah entitas yang lebih terlihat dari entitas yang lainnya.



Gambar 3.1 Salah satu objek menonjol
Sumber: dokumentasi penulis

- Jika ada interaksi yang membutuhkan sebuah barang kunci, lakukan Brute Force dengan mencoba semua barang kunci yang ada. Barang kunci yang sudah dicoba tidak perlu dicoba lagi. Jika Brute Force tidak membuahkan hasil, berarti barang kunci yang dibutuhkan tidak ada dan lokasinya dicatat untuk kemudian jika sudah mendapat barang kunci baru.
- Jika dari langkah di atas didapat barang kunci baru, jika jumlah barang kunci lebih dari 1, lakukan Brute Force dengan mencoba semua kombinasi barang kunci dengan harapan mendapat barang kunci yang baru. Kombinasi yang sudah dicoba tidak perlu dilakukan lagi.

A. Penelusuran Pertama

Berikut adalah kondisi awal:

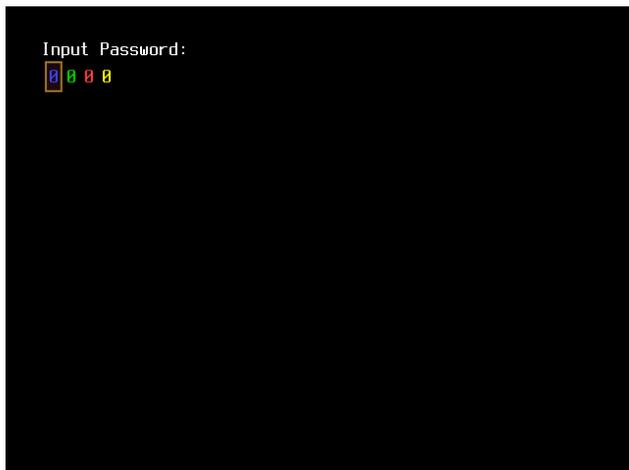


Gambar 3.2 Kondisi awal
Sumber: dokumentasi penulis

Pada kondisi awal, Niko tidak memiliki barang kunci apapun, belum menelusuri tempat apapun juga, dan berada di

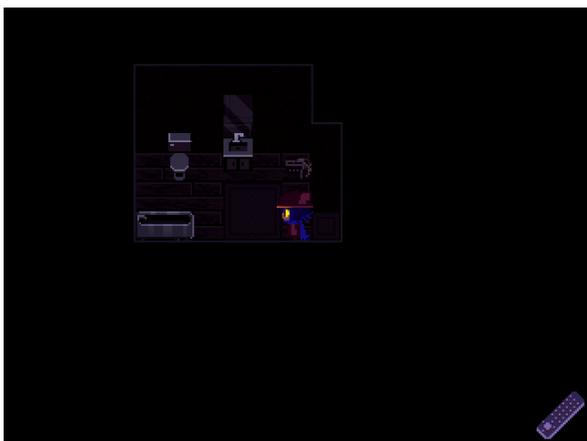
ruang awal. Ruang ini akan disebut Ruang 1. Di ruang ini, ada lima titik yang menonjol, yaitu tempat tidur, komputer, rak buku, jendela, dan remot. Selain itu, ada dua pintu di sebelah kiri yang akan disebut Ruang 2 dan sebelah bawah yang akan disebut Ruang 3 ruangan untuk dimasuki.

Langkah selanjutnya ada melakukan interaksi dengan kelima objek yang ada. Saat interaksi dilakukan pada komputer, didapat bahwa diperlukan sebuah kunci berupa empat buah angka dengan warna berbeda sehingga entitas ini perlu dicatat lokasinya. Interaksi pada remote akan membuat Niko mengambil remot tersebut dan mendapat sebuah barang kunci. Interaksi pada tempat tidur dan rak buku tidak menghasilkan apa-apa. Interaksi dengan jendela juga tidak menghasilkan apa-apa, namun membutuhkan sebuah barang kunci.



Gambar 3.3 Input Password
Sumber: dokumentasi penulis

Dengan ruang pertama selesai, selanjutnya adalah memasuki ruangan di sebelah kiri.



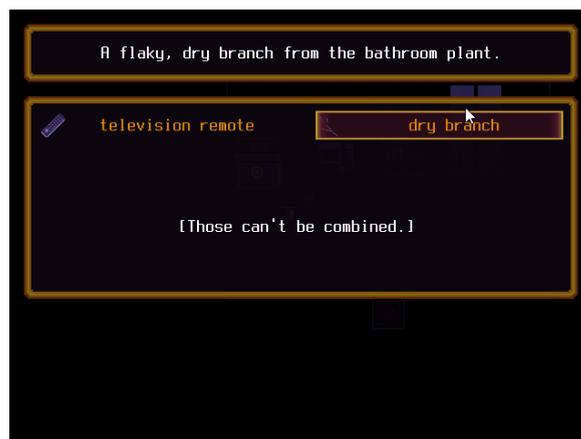
Gambar 3.4 Ruang 2
Sumber: dokumentasi penulis

Di ruang ini, ada empat entitas yang menonjol, yaitu bak mandi, kloset, cermin, dan tanaman yang layu.

Lakukan interaksi dengan keempat entitas. Dari keempat entitas, bak mandi, kloset, dan cermin tidak memberikan hasil apa-apa. Interaksi dengan tanaman layu memberikan Niko sebuah barang kunci baru, yaitu sebuah ranting.

Ruang yang tersisa adalah Ruang 3 dari ruang yang pertama, namun tidak bisa dimasuki karena dikunci. Jadi, penelusuran pertama sudah selesai.

Hasil yang didapat adalah dua buah barang kunci, yaitu ranting dan remot, dan sebuah komputer yang perlu sebuah komputer yang perlu empat buah angka. Ketika dua barang kunci dicoba untuk digabung, tidak menghasilkan sebuah barang baru, maka langsung lanjut ke penelusuran ke-dua.

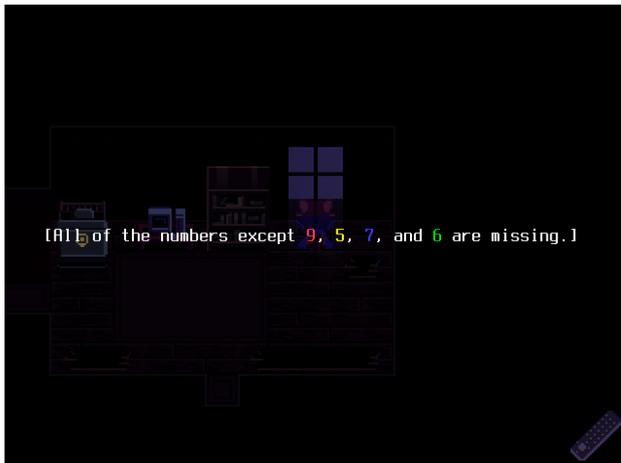


Gambar 3.5 Barang Tidak Dapat Digabung
Sumber: dokumentasi penulis

B. Penelusuran Kedua

Karena ada komputer yang memerlukan sebuah barang kunci, pertama pergi ke tempat komputer berada. Lalu, dengan semua barang kunci yang ada, coba digunakan pada komputer. Hasil yang diperoleh adalah komputer tidak berubah perilakunya, maka masih tetap dicatat lokasinya.

Lakukan interaksi yang sama pada semua entitas, namun sambil mencoba semua barang kunci yang ada pada entitas tersebut. Pada jendela, akan muncul hasil baru, yaitu empat buah angka dengan warna yang berbeda. Meskipun tidak masuk ke inventory, ini merupakan sebuah kunci.



Gambar 3.6 Hasil pada Remot
Sumber: dokumentasi penulis

Pada ruangan selanjutnya juga dilakukan hal yang sama. Hasilnya adalah tidak ada perubahan sama sekali. Ruang 3 juga masih terkunci

C. Penelusuran Ketiga

Kembali lagi ke komputer yang belum diselesaikan. Kali ini, Niko memiliki sebuah kunci, yaitu empat buah angka dengan warna yang berbeda, maka samakan angka pada komputer dengan yang didapat dari remot dan konfirmasi.

Sekarang, lakukan penelusuran yang sama dengan yang sebelumnya, sejauh ini tidak akan ada perubahan dari penelusuran sebelumnya, namun kali ini akan didapat pintu menuju Ruang 3 sudah terbuka, maka penelusuran akan dilanjutkan ke ruang setelahnya.



Gambar 3.7 Ruang 3
Sumber: dokumentasi penulis

Di ruangan ini, ada enam entitas yang menonjol, yaitu simbol di sebelah kiri, televisi rusak, perapian, wastafel, meja, dan kulkas. Ruangan ini juga terhubung dengan sebuah ruang lain di sebelah kanan yang akan disebut Ruang 4.

Lakukan interaksi semua entitas tersebut. Simbol, televisi, dan perapian akan menunjukkan bahwa diperlukan sebuah barang kunci, maka untuk entitas tersebut, lakukan Brute

Force barang kunci yang dimiliki. Hasil yang didapat adalah tidak ada perubahan, maka catat lokasi ketiga entitas tersebut. Wastafel dan meja tidak memberikan apa-apa. Kulkas akan memberikan Niko sebuah barang kunci baru, yaitu botol alkohol.

Saat hendak memasuki ruang sebelah kanan, akan didapat pintu tersebut dikunci dan perlu sebuah barang kunci. Karena itu, tidak ada lagi yang dapat ditelusuri untuk penelusuran ini.

Selanjutnya, karena Niko mendapat sebuah barang kunci baru, lakukan Brute Force barang kunci botol alkohol tersebut dengan barang kunci lainnya. Hasilnya adalah didapat sebuah barang kunci baru, yaitu ranting yang dibasahi alkohol dan botol alkohol menjadi kosong. Ini berarti Niko memiliki dua barang kunci baru.



Gambar 3.8 Kedua Barang Digabungkan
Sumber: dokumentasi penulis

D. Penelusuran Keempat

Untuk penelusuran keempat, masih ada empat entitas yang memerlukan sebuah barang kunci, yaitu simbol, televisi, perapian, dan pintu menuju Ruang 4. Maka lakukan searching entitas-entitas tersebut dan pergi ke lokasi mereka, yaitu Ruang 3. Untuk setiap entitas tersebut, lakukan Brute Force dengan semua barang kunci yang ada, yaitu ranting basah dan botol kosong.

Interaksi ranting basah dan botol kosong pada simbol tidak membuahkan hasil apa-apa, maka langsung dilanjutkan ke entitas selanjutnya.

Interaksi televisi dan ranting basah akan menyebabkan ranting menjadi terbakar. Ranting terbakar akan menggantikan tempat ranting basah sebelumnya. Interaksi dengan botol kosong tidak akan menghasilkan apa-apa sehingga barang kunci kini menjadi ranting terbakar dan botol kosong.

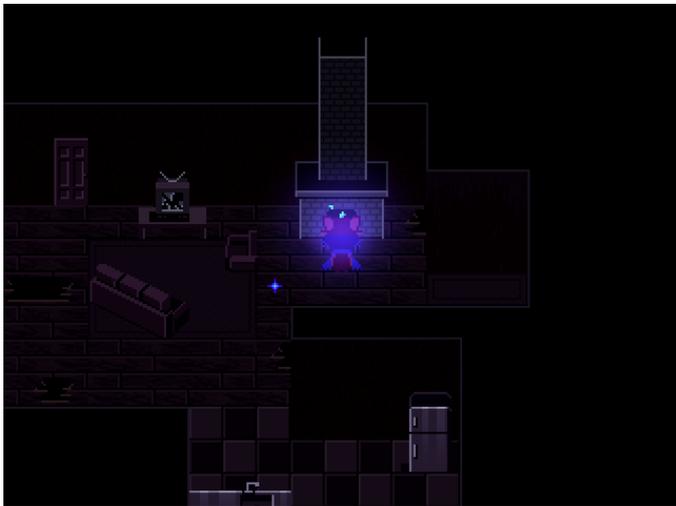


Gambar 3.9 Ranting Terbakar
Sumber: dokumentasi penulis

Interaksi perapian dengan ranting terbakar akan menyalakan perapian. Interaksi ini akan memunculkan sebuah entitas menonjol baru, yaitu kilauan di lantai.

Pintu menuju Ruang 4 juga masih terkunci

Karena semua entitas yang dicatat sudah dikunjungi, interaksi dengan semua entitas menonjol yang belum di interaksi lagi. Semua entitas pada Ruang 1 dan 2 tidak perlu diperiksa lagi karena sudah di interaksi. Ruang 3 hanya ada satu yang belum, yaitu kilauan pada lantai yang baru muncul, maka selanjutnya akan berinteraksi dengan entitas tersebut. Interaksi ini akan memberikan Niko sebuah barang kunci baru, yaitu *Basement Key*.



Gambar 3.10 Kilauan Lantai
Sumber: dokumentasi penulis

Karena tidak ada lagi yang bisa ditelusuri, maka sekarang untuk setiap barang kunci dicoba untuk digabungkan. Barang kunci yang ada hanya *Basement Key* dan botol kosong dan

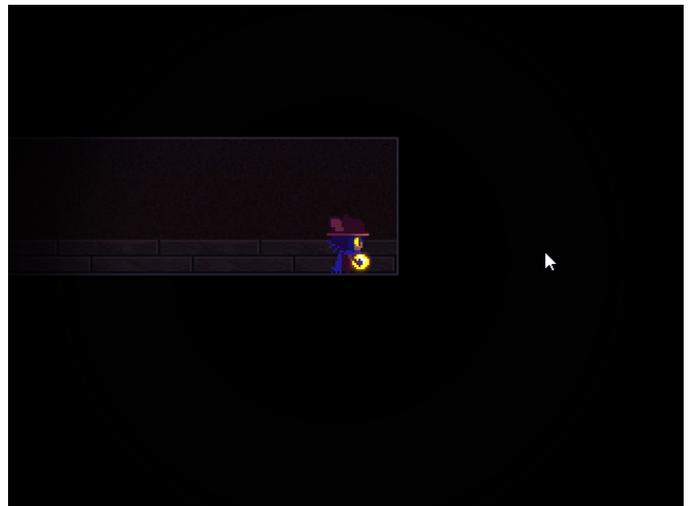
kombinasi kedua barang kunci tidak membuahkan apa-apa. Dengan ini, berakhir penelusuran keempat.

E. Penelusuran Kelima

Penelusuran akan dimulai dengan pergi ke lokasi simbol dan pintu menuju Ruang 4 karena merupakan entitas yang perlu sebuah barang kunci dan pada penelusuran sebelumnya didapat barang kunci baru, yaitu *Basement Key*. Interaksi dengan *Basement Key* dan simbol tersebut tidak memberikan hasil apa-apa. Interaksi dengan pintu akan membuka pintu tersebut dan sekarang terbuka sebuah jalan baru.

Lanjut dengan penelusuran seperti biasa. Semua entitas lain sudah selesai, sehingga akan langsung menuju Ruang 4. Di ruang ini, hanya ada satu entitas menonjol yang dapat berinteraksi, yaitu bohlam yang mati.

Interaksi akan memberikan Niko sebuah barang kunci baru, yaitu *Lightbulb*. Karena penelusuran sudah selesai, akan dicoba penggabungan *Lightbulb* dan botol kosong karena hanya ada kedua barang kunci tersebut. Hasilnya keduanya tidak dapat digabungkan, maka dengan ini berakhir penelusuran kelima.

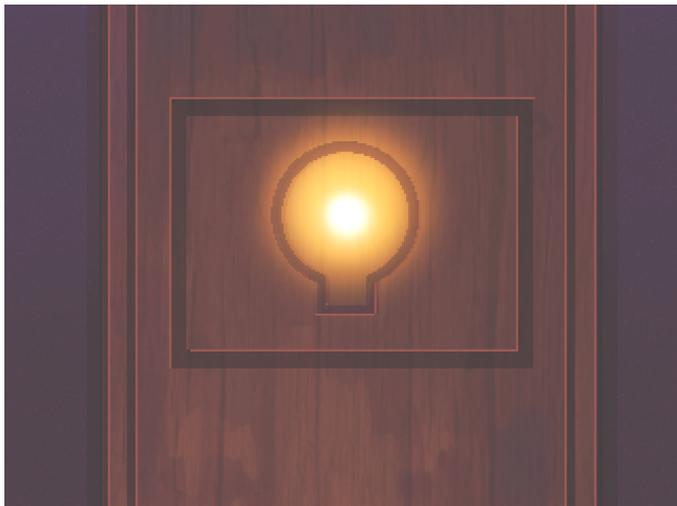


Gambar 3.11 Mendapat *Lightbulb*
Sumber: dokumentasi penulis

F. Penelusuran Keenam

Pada penelusuran ini, tidak banyak yang berubah kecuali bertambahnya barang kunci *Lightbulb* dan simbol tersisa yang membutuhkan sebuah barang kunci. Maka penelusuran dimulai dengan berinteraksi dengan simbol tersebut dengan *Lightbulb*.

Interaksi ini akan membawa Niko keluar dari rumah dan meninggalkan area pertama. Dengan ini, tujuan berhasil dicapai.



Gambar 3.2 Simbol Menyalakan
Sumber: dokumentasi penulis

IV. KESIMPULAN

Algoritma Brute Force merupakan salah satu algoritma dalam strategi algoritma yang dapat digunakan untuk menyelesaikan sebuah puzzle. Brute Force dapat mencari semua kemungkinan yang ada tanpa melewati apapun. Meskipun dalam skala besar akan semakin tidak efektif, Brute Force tetap merupakan salah satu solusi yang dapat digunakan.

Dalam memanfaatkan algoritma ini, Brute Force dapat dibuat lebih efisien dengan memanfaatkan Heuristik. Pada makalah, Heuristik dapat digunakan saat menggabungkan barang kunci berupa kecocokan. Contohnya pada Gambar 3.5, remot dan ranting tidak memiliki kecocokan sama sekali jadi tidak perlu dicoba.

Algoritma DFS dapat digunakan untuk path finding atau mencari jalan. Pada permainan ini, DFS dapat berguna untuk mencari jalan dari suatu ruang ke ruang lainnya.

V. SARAN

Selain pada game OneShot, algoritma serupa dapat diaplikasikan juga pada game yang serupa, yaitu game yang memerlukan pencarian barang kunci untuk membuka jalan untuk sampai ke sebuah tujuan, seperti Escape Room.

Algoritma DFS dapat dibuat lebih optimal. Pada skala besar, DFS mungkin akan menghasilkan rute yang lebih panjang padahal ada jalan yang lebih pendek. Untuk pemain biasa, DFS tidak menjadi masalah, namun untuk beberapa pemain yang mementingkan waktu, akan lebih baik mencari algoritma yang lebih efektif.

VIDEO LINK AT YOUTUBE

https://youtu.be/wCe_ouCyeMI

UCAPAN TERIMA KASIH

Saya berterima kasih kepada Tuhan Yang Maha Esa yang memampukan saya untuk menulis makalah ini dengan sebaik-baiknya mulai dari mencari ide sampai menyelesaikannya.

Saya juga berterima kasih kepada orang-orang yang telah mendukung saya selama proses pembuatan makalah. Saya berterima kasih kepada orang tua saya yang mendukung saya selama di rumah, berterima kasih kepada Bapak dan Ibu dosen khususnya Bapak Prof. Ir. Dwi Hendratmo Widyantoro, M.Sc., Ph.D. sebagai pengajar mata kuliah IF2211 Strategi Algoritma di kelas saya, dan Bapak Dr. Ir. Rinaldi Munir yang menyediakan akses ke websitenya.

Untuk yang terakhir, saya berterima kasih kepada pembuat game OneShot, Future Cat, yang telah membuat game yang sangat berkesan bagi saya dan banyak orang lainnya.

REFERENCES

- [1] [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Brute-Force-\(2021\)-Bag1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Brute-Force-(2021)-Bag1.pdf)
- [2] [https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Brute-Force-\(2021\)-Bag2.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Algoritma-Brute-Force-(2021)-Bag2.pdf)
- [3] <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/BFS-D FS-2021-Bag1.pdf>
- [4] <https://oneshot.fandom.com/>
- [5] <https://store.steampowered.com/app/420530/OneShot/>

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 26 April 2021

Andrianata Putra Tjandra 13519147