

Penerapan Algoritma Backtracking dalam Permainan Labirin

Louis Riemenn 13519016
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
13519016@std.stei.itb.ac.id

Abstract—Permasalahan atau teka teki labirin adalah salah satu permainan yang sering kita jumpai semasa muda kita. Biasanya permainan ini kita temui pada masa sekolah sebagai salah satu contoh persoalan ataupun sebagai permainan untuk mengisi waktu luang. Dalam literatur ini, penulis akan mencoba menyelesaikan masalah labirin ini dengan metode backtracking dan diharapkan hasil karya ini dapat membantu kita untuk lebih bisa memahami metode backtracking dan penerapannya secara langsung di kehidupan sehari-hari.

Keywords—backtracking, labirin, teka-teki

I. PENDAHULUAN

Permainan Labirin merupakan permainan yang kebanyakan dari kita pasti pernah melihat atau ketahui. Permainan ini biasanya atau kebanyakan kita temui dalam soal yang diberikan ketika kita berada di bangku sekolah ataupun ketika kita sedang bermain seperti biasa dengan sahabat kita. Permainan Labirin biasanya identik dengan sebuah daerah yang terisolasi dengan satu jalan masuk dan satu jalan keluar ataupun dua pintu yang menghubungkan daerah terisolasi tersebut dengan daerah luar yang mana ke dua titik tersebut pastilah terhubung dan tersedia jalur yang menghubungkan kedua titik tersebut.

Permainan labirin merupakan permainan dimana pemain akan memulai berjalan dari suatu titik asal dan berusaha menemukan jalan yang benar yang nantinya jika jalan itu dilalui dari titik asal, maka akan menuntun pemain tersebut ke jalan keluar atau titik akhir dari labirin tersebut. Umumnya labirin dibuat dalam bentuk kotak, namun banyak juga yang membuatnya dalam bentuk lingkaran atau yang lainnya.

Pada masanya, permainan labirin merupakan salah satu permainan yang digemari oleh anak-anak bahkan sampai orang dewasa. Hal ini disebabkan permainan ini memaksa otak kita untuk tetap focus dan berpikir untuk mencari jalan yang benar agar kita dapat mencapai tujuan kita. Tak hanya itu, karena populernya konsep labirin, bahkan konsep ini juga banyak ditiru dan sedikit dimodifikasi dan diterapkan dalam berbagai macam game seperti misalnya game ‘Pac-Man’ yang sedikit meniru konsep labirin untuk membuat gamenya menarik.

II. LANDASAN TEORI

A. Labirin

Secara historis, perlu diketahui bahwa labirin sudah ada sejak zaman dulu. Bahkan dalam mitologi Yunani sudah ada menyebutkan labirin. Dalam mitologi Yunani, dipercaya bahwa labirin dibuat oleh Daedalus untuk menjebak sebuah monster bernama Minotaur. Minotaur sendiri merupakan sebuah monester dalam mitologi Yunani yang bentuk tubuhnya seperti bentuk tubuh manusia, namun monster ini memiliki kepala yang mirip dengan kepala banteng. Selain daripada teori tersebut, ada juga yang percaya bahwa pada awalnya, labirin dibuat bukan untuk membingungkan manusia melainkan untuk menuntun perjalanan spiritual mereka.

Pada awalnya, labirin hanyalah satu buah jalan yang membingungkan. Namun lama kelamaan konsep labirin digunakan dan dimodifikasi untuk digunakan dalam berbagai macam hal, hingga akhirnya muncullah konsep labirin dengan banyak jalan dimana kita harus menemukan jalan yang benar untuk menuju titik akhir atau lokasi tujuan.

Beberapa tahun kemudian labirin pun mulai dipergunakan untuk berbagai macam keperluan. Seperti misalnya saja pada tahun 1882, labirin mulai digunakan untuk melakukan penelitian psikologi hewan. Tidak hanya itu, muncul pula sebuah pertunjukan yang bertemakan labirin yang mana dalam pertunjukan itu, jika seseorang berhasil memecahkan labirin tersebut, maka orang tersebut akan menerima hadiah berupa sejumlah uang. Sampai pada saat ini, banyak yang menggunakan konsep labirin untuk dijadikan sebuah teka-teki untuk anak-anak, untuk mengasah kemampuan anak.

Dalam artikel kali ini, penulis akan membahas labirin yang menggunakan konsep ruangan terisolasi dengan banyak jalan yang mana pemain akan memulai perjalanan dari suatu titik dan akan mencari jalan yang benar agar pemain tersebut dapat sampai ke titik akhir dari labirin tersebut.

B. Backtracking

Backtracking merupakan sebuah metode atau algoritma pemecahan yang mangkus, terstruktur, dan sistematis, baik untuk persoalan optimasi maupun non-optimasi.

Seperti algoritma pencarian lain, backtracking juga merupakan sebuah algoritma yang merupakan hasil dari

perbaiki algoritma exhaustive search. Exhaustive search sendiri merupakan sebuah metode atau algoritma yang mana algoritma tersebut akan mencari semua kemungkinan solusi yang mungkin lalu dievaluasi satu per satu. Dalam metode exhaustive search sendiri akan menjamin untuk menemukan sebuah solusi yang optimal. Meskipun metode ini menjamin untuk menemukan hasil yang optimal namun metode ini memakan banyak waktu sehingga muncullah metode-metode yang lain yang mungkin tidak selalu menghasilkan hasil yang optimal namun dapat memangkas banyak waktu yang digunakan dalam pencarian.

Backtracking sendiri merupakan algoritma yang hanya memilih pilihan yang mengarah ke solusi yang akan dieksplorasi. Jika pilihan tersebut sudah tidak mengarah ke solusi maka pilihan itu akan dipangkas dan tidak dipertimbangkan lagi. Algoritma backtracking atau dikenal juga dengan algoritma runut balik sendiri pertama kali diperkenalkan oleh D.H. Lehmer tahun 1950. Kemudian R.J Walker, Golomb, dan Baumert menyajikan uraian umum tentang algoritma runut balik.

Algoritma backtracking memiliki beberapa property umum:

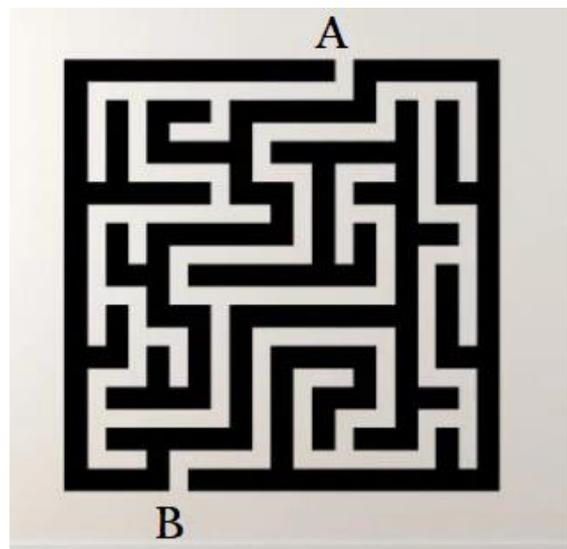
1. Solusi Persoalan
2. Fungsi Pembangkit
3. Fungsi Pembatas

Solusi persoalan akan berisi solusi untuk persoalan yang sedang dikerjakan. Sedangkan untuk fungsi pembangkit merupakan fungsi yang digunakan untuk membangkitkan simpul (menjadi simpul hidup) dari simpul ekspansi yang sedang diperiksa. Yang terakhir, fungsi pembatas merupakan fungsi yang digunakan untuk memeriksa simpul ekspansi yang sedang diperiksa apakah melanggar Batasan (tidak menuju solusi) atau tidak (masih menuju solusi).

III. PENERAPAN ALGORITMA BACKTRACKING PADA PERMAINAN LABIRIN

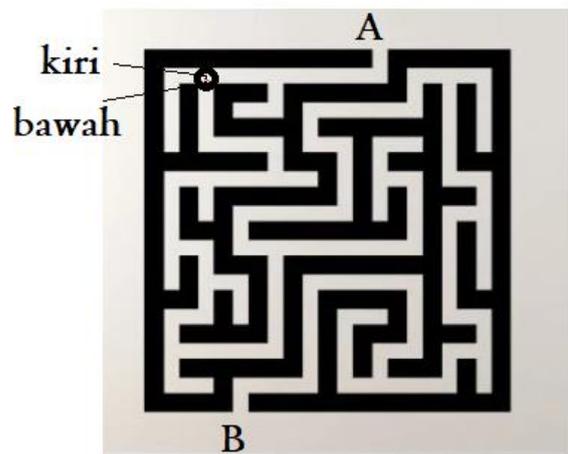
Pada bab ini akan dijelaskan penerapan algoritma backtracking atau runut balik dalam permainan labirin. Seperti yang telah dijelaskan pada bab sebelumnya, maka kita akan mencari fungsi untuk algoritma backtracking terlebih dahulu.

Sebelum itu, kita perlu contoh labirin terlebih dahulu sebagai contoh kasus. Untuk contoh kasus, penulis akan menggunakan labirin di bawah ini.



Gambar 3.1 Contoh Kasus labirin

Pada kasus ini, kita akan mencari solusi berupa jalur yang benar agar kita bisa berjalan dari A sampai ke B. Lalu akan kita cari fungsi pembangkit dan juga fungsi pembatas untuk kasus labirin tersebut. Untuk fungsi pembatas, Karena ini merupakan labirin bukan graf, maka akan dibuat pengandaian, setiap ada persimpangan, anggap saja bahwa saat itu ada beberapa simpul yang bisa diekspan sesuai arah jalurnya. Untuk simpul yang diekspan, anggap saja diekspan berdasarkan arahnya, dengan urutan prioritas sebagai berikut dari yang terbesar ke yang terkecil : kiri – bawah – kanan – atas. Untuk lebih jelasnya, coba perhatikan contoh dibawah di saat simpul asal atau root (dari posisi A) akan diekspan.



Gambar 3.2 Contoh Pertemuan Simpang

Pada gambar di atas, saat bertemu dengan sebuah persimpangan, maka dapat diartikan sebagai ada dua jalur atau dua simpul yang perlu dipertimbangkan dalam pemilihannya. Saat kita memilih sebuah simpul, maka kita akan menamainya dengan : <arah>-<n> dengan arah adalah salah satu dari [kiri , bawah , kanan , atas] dan n merupakan urutan simpul ke berapa jika dihitung setelah simpul asal. Missal untuk contoh kasus tadi kita memilih simpul kiri, maka akan dinamai KIRI-1.

Selanjutnya kita perlu mencari fungsi pembangkit dan pembatas untuk permasalahan labirin tersebut. Untuk fungsi pembangkit, perlu kita ingat Kembali kita telah memiliki prioritas pembangkit di paragraph sebelumnya yaitu urutan prioritas sebagai berikut dari yang terbesar ke yang terkecil : kiri – bawah – kanan – atas. Urutan ini akan digunakan sebagai fungsi pembangkit saat menemukan sebuah persimpangan.

Adapaun untuk fungsi pembatas, dikarenakan ini merupakan permasalahan labirin, yang mana satu-satunya cara menentukan sebuah jalur tidak lagi menuju solusi adalah saat kita menemukan jalan buntu di jalur tersebut, maka fungsi pembatasnya akan menjadi apakah simpul tersebut menghasilkan jalan buntu atau tidak.

Strategi algoritma backtracking akan diaplikasikan sebagai berikut.

Iterasi 1 :

Simpul Ekspan : R (anggap saja sebuah nama untuk simpul awal)

Simpul Hidup : KIRI-1 , BAWAH-1 (dapat dilihat dari gambar 3.2)

Solusi saat ini : [R]

Sesuai urutan pembangkitan, maka selanjutnya kita akan melakukan ekspansi pada simpul KIRI-1.

Iterasi 2 :

Simpul Ekspan : KIRI-1

Simpul Hidup : -

Solusi saat ini : [R , KIRI-1]

Jika kita lihat, saat kita memilih simpul KIRI-1, kitapun menemukan jalan buntu, oleh karena itu, maka simpul KIRI-1 akan kita pangkas dan Kembali lagi ke simpul sebelumnya yaitu R.

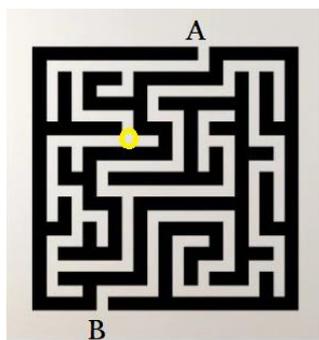
Iterasi 3 :

Simpul Ekspan : R

Simpul Hidup : BAWAH-1

Solusi saat ini : [R]

Selanjutnya kita akan mengekspan ke simpul hidup yang telah dibangkitkan yang pada kasus ini merupaka BAWAH-1.



Gambar 3.4 Jalur BAWAH-1

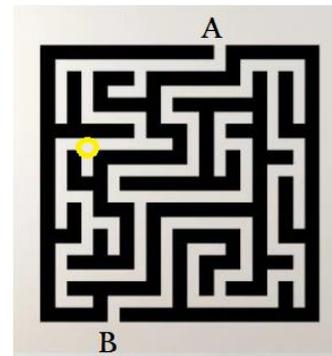
Iterasi 4 :

Simpul Ekspan : BAWAH-1

Simpul Hidup : KIRI-2 , KANAN-2 (dapat dilihat pada gambar 3.4)

Solusi saat ini : [R , BAWAH-1]

Selanjutnya kita akan mengekspan sesuai urutan kebangkitan yang dalam kasus ini, maka kita akan mengekspan ke simpul KIRI-2.



Gambar 3.5 Jalur KIRI-2

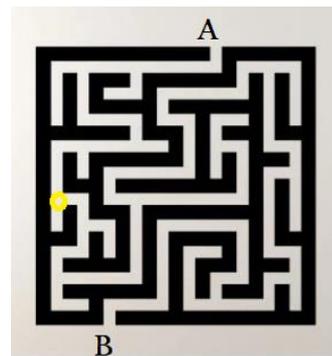
Iterasi 5 :

Simpul Ekspan : KIRI-2

Simpul Hidup : KIRI-3 , BAWAH-3 (dapat dilihat pada gambar 3.5)

Solusi saat ini : [R, BAWAH-1, KIRI-2]

Selanjutnya kita akan melakukan ekspansi ke simpul KIRI-3.



Gambar 3.6 Jalur KIRI-3

Iterasi 6 :

Simpul Ekspan : KIRI-3

Simpul Hidup : BAWAH-4 , KANAN-4 (dapat dilihat pada gambar 3.6)

Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3]

Selanjutnya kita akan melakukan ekspansi ke simpul BAWAH-4.

Iterasi 7 :

Simpul Ekspan : BAWAH-4

Simpul Hidup : -

Solusi Saat ini : [R, BAWAH-1, KIRI-2, KIRI-3, BAWAH-4]

Bisa dilihat dari gambar 3.6 bahwa saat kita melakukan ekspansi ke simpul BAWAH-4, maka kita akan menemukan jalan buntu. Oleh sebab itu, maka kita akan memangkas simpul BAWAH-4 dan kembali ke simpul sebelumnya.

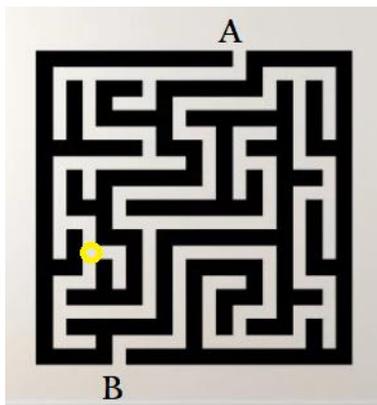
Iterasi 8 :

Simpul Ekspan : KIRI-3

Simpul Hidup : KANAN-4

Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3]

Selanjutnya kita akan melakukan ekspansi ke simpul yang telah dibangkitkan yaitu simpul KANAN-4.



Gambar 3.7 Jalur KANAN-4

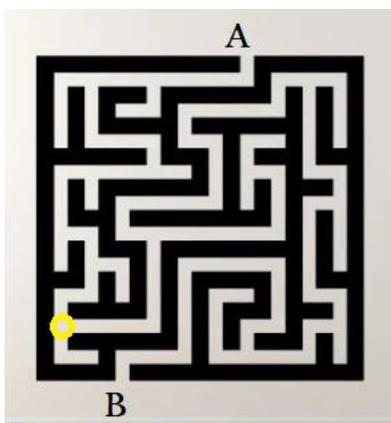
Iterasi 9 :

Simpul Ekspan : KANAN-4

Simpul Hidup : BAWAH-5 , KANAN-5 (dapat dilihat pada gambar 3.7)

Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3, KANAN-4]

Selanjutnya akan dilakukan ekspansi terhadap simpul yang telah dibangkitkan yaitu simpul BAWAH-5.



Gambar 3.8 Jalur BAWAH-5

Iterasi 10 :

Simpul Ekspan : BAWAH-5

Simpul Hidup : BAWAH-6 , KANAN-6 (dapat dilihat pada gambar 3.8)

Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3, KANAN-4, BAWAH-5]

Selanjutnya kita akan melakukan ekspansi terhadap simpul yang telah dibangkitkan yaitu simpul BAWAH-6.

Iterasi 11 :

Simpul Ekspan : BAWAH-6

Simpul Hidup : -

Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3, KANAN-4, BAWAH-5, BAWAH-6]

Dapat kita lihat pada gambar 3.8, saat kita melakukan ekspansi ke simpul BAWAH-6, ternyata kita menemukan sebuah jalan buntu. Oleh sebab itu, simpul BAWAH-6 akan dipangkas dan kita akan Kembali ke simpul sebelumnya.

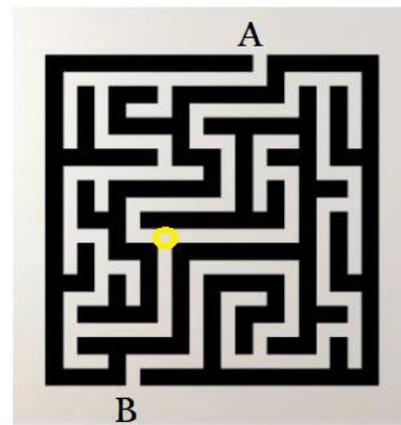
Iterasi 12 :

Simpul Ekspan : BAWAH-5

Simpul Hidup : KANAN-6

Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3, KANAN-4, BAWAH-5]

Selanjutnya kita akan melakukan ekspansi terhadap simpul yang telah dibangkitkan yaitu simpul KANAN-6.



Gambar 3.9 Jalur KANAN-6

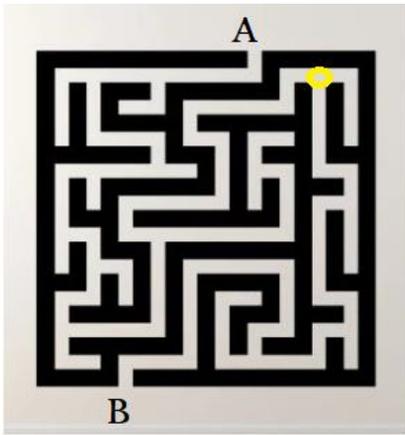
Iterasi 13 :

Simpul Ekspan : KANAN-6

Simpul Hidup : KIRI-7 , KANAN-7 (dapat dilihat pada gambar 3.9)

Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3, KANAN-4, BAWAH-5, KANAN-6]

Selanjutnya dilakukan ekspansi terhadap simpul yang telah dibangkitkan yaitu simpul KIRI-7.



Gambar 3.10 Jalur KIRI-7

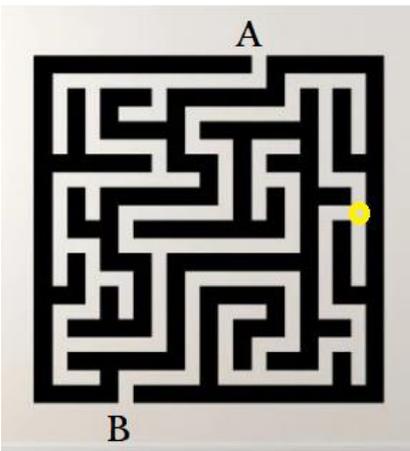
Iterasi 14 :

Simpul Ekspan : KIRI-7

Simpul Hidup : BAWAH-8 , KANAN-8 (dapat dilihat pada gambar 3.10)

Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3, KANAN-4, BAWAH-5, KANAN-6, KIRI-7]

Selanjutnya akan dilakukan ekspansi terhadap simpul yang telah dibangkitkan yaitu simpul BAWAH-8.



Gambar 3.11 Jalur BAWAH-8

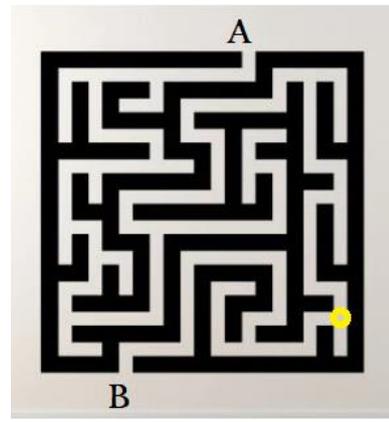
Iterasi 15 :

Simpul Ekspan : BAWAH-8

Simpul Hidup : KIRI-9 , BAWAH-9 (dapat dilihat pada gambar 3.11)

Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3, KANAN-4, BAWAH-5, KANAN-6, KIRI-7, BAWAH-8]

Selanjutnya akan dilakukan ekspansi terhadap simpul yang telah dibangkitkan yaitu simpul KIRI-9.



Gambar 3.12 Jalur KIRI-9

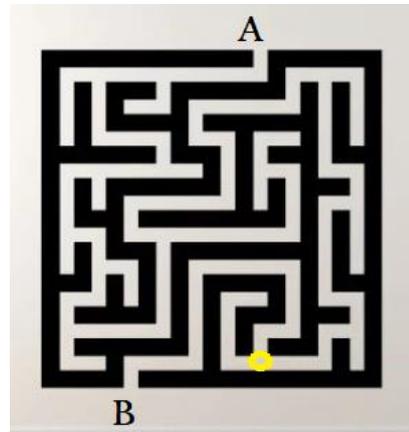
Iterasi 16 :

Simpul Ekspan : KIRI-9

Simpul Hidup : KIRI-10 , BAWAH-10 (dapat dilihat pada gambar 3.12)

Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3, KANAN-4, BAWAH-5, KANAN-6, KIRI-7, BAWAH-8, KIRI-9]

Selanjutnya akan dilakukan ekspansi terhadap simpul yang telah dibangkitkan yaitu simpul KIRI-10.



Gambar 3.13 Jalur KIRI-10

Iterasi 17 :

Simpul Ekspan : KIRI-10

Simpul Hidup : KIRI-11 , ATAS-11 (dapat dilihat pada gambar 3.13)

Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3, KANAN-4, BAWAH-5, KANAN-6, KIRI-7, BAWAH-8, KIRI-9, KIRI-10]

Selanjutnya akan dilakukan ekspansi terhadap simpul yang telah dibangkitkan yaitu simpul KIRI-11.

Iterasi 18 :

Simpul Ekspan : KIRI-11

Simpul Hidup : -

Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3, KANAN-4, BAWAH-5, KANAN-6, KIRI-7, BAWAH-8, KIRI-9, KIRI-10, KIRI-11]

Dari gambar 3.13 bisa kita ketahui bahwa saat kita melakukan ekspan pada simpul KIRI-11, maka kita akan menemukan jalan buntu. Oleh sebab itu, simpul KIRI-11 dipangkas dan akan kembali ke simpul sebelumnya.

Iterasi 19 :

Simpul Ekspan : KIRI-10

Simpul Hidup : ATAS-11

Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3, KANAN-4, BAWAH-5, KANAN-6, KIRI-7, BAWAH-8, KIRI-9, KIRI-10]

Selanjutnya akan dilakukan ekspansi terhadap simpul yang telah dibangkitkan yaitu simpul ATAS-11.

Iterasi 20 :

Simpul Ekspan : ATAS-11

Simpul Hidup : G (simpul tujuan, dapat dilihat dari table 3.13)

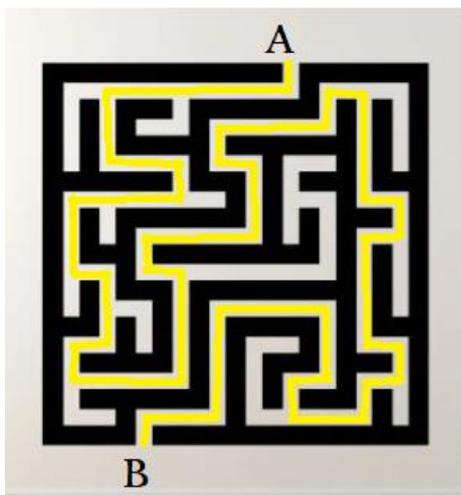
Solusi saat ini : [R, BAWAH-1, KIRI-2, KIRI-3, KANAN-4, BAWAH-5, KANAN-6, KIRI-7, BAWAH-8, KIRI-9, KIRI-10, ATAS-11]

Selanjutnya akan dilakukan ekspan terhadap simpul yang telah dibangkitkan yaitu simpul G.

Iterasi 21 :

Simpul Ekspan : G

Karena simpul G telah dicapai maka pencarian pun dihentikan dengan solusi berupa [R, BAWAH-1, KIRI-2, KIRI-3, KANAN-4, BAWAH-5, KANAN-6, KIRI-7, BAWAH-8, KIRI-9, KIRI-10, ATAS-11, G]



Gambar 3.14 Solusi

IV. ANALISIS HASIL PENERAPAN

Algoritma Backtracking digunakan untuk mencari jalur yang menuju ke tempat tujuan yang diinginkan (dalam kasus ini ingin dituju posisi B). Dengan menggunakan algoritma backtracking, maka akan didapatkan hasil berupa jalur yang benar yang menuju ke posisi akhir dengan strategi mengeliminasi simpul yang tidak menuju ke simpul tujuan.

Berdasarkan percobaan, telah didapatkan jalur yang tepat yang menuju posisi akhir dari posisi awal seperti tertera pada gambar 3.14 di atas.

V. KESIMPULAN

Dari percobaan yang telah dilakukan, dapat disimpulkan bahwa algoritma backtracking sangat cocok diterapkan dalam permasalahan labirin, dikarenakan permasalahan labirin hanya memiliki satu solusi sehingga algoritma backtracking akan selalu menghasilkan solusi optimal.

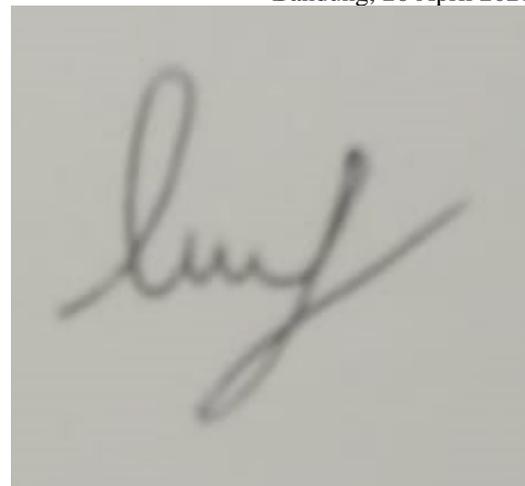
REFERENSI

- [1] <https://id.aliexpress.com/item/32692993076.html> diakses pada tanggal 11 mei 2021 pukul 19.16
- [2] Munir, Rinaldi. 2009. Diktat Kuliah IF2211 Strategi Algoritma. Departemen Teknik Informatika IT

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



Louis Riemenn 13519016