

# Implementasi Algoritma Backtracking Pada Permainan Sudoku

1) Fauzan Azima Rahman

Universitas Budi Luhur, Jl. Ciledug Raya, Jakarta Selatan, Indonesia  
E-Mail: [Fauzan.arahman21@gmail.com](mailto:Fauzan.arahman21@gmail.com)

2) Dian Anubhakti

Universitas Budi Luhur, Jl. Ciledug Raya, Jakarta Selatan, Indonesia  
E-Mail: [Dian.anubhakti@budiluhur.ac.id](mailto:Dian.anubhakti@budiluhur.ac.id)

## ABSTRACT

Artificial intelligence is a field of computer science that is learning about computer devices that can be in the program using the machine in order to do commands from a programmer using certain instructions. One form of Artificial intelligence representation is a game or game. With the science of artificial intelligence then certain games will be able to be easily solved using computer assistance. One form of the game that can be inserted in an artificial intelligence game is Sudoku. The game is a form of numeric play where players are required to fill a blank box that is already available using a random number of numbers starting from 1 to 9. Many players are unable to complete this Sudoku game. This research uses Backtracking algorithms in solving sudoku puzzles. Research results in the form of applications capable of completing Sudoku puzzles.

**Keyword : Puzzle, Sudoku, Game.**

## PENDAHULUAN

Perkembangan teknologi komputer yang sangat pesat banyak digunakan dikalangan akademis, kesehatan, pendidikan, permainan dan sebagainya. salah satu cabang ilmu yang muncul dari komputer adalah Artificial Intelligence atau kecerdasan buatan. Artificial Intelligence merupakan bidang komputer yang mempelajari tentang perangkat komputer yang dapat diprogram menggunakan mesin dengan tujuan agar mampu melaksanakan keinginan programmer melalui perintah-perintah tertentu, sehingga perangkat komputer tersebut dapat berfikir layaknya manusia [1]. salah satu bentuk representasi dari kecerdasan buatan tersebut adalah sebuah permainan atau game. Sudoku merupakan sebuah permainan yang populer pada era tahun 70 an. Permainan ini adalah permainan angka yang mengharuskan pemain untuk mengisi kotak-kotak kosong yang telah disediakan menggunakan bilangan angka secara acak yaitu angka 1 sampai dengan 9 [2]. Aturan dari permainan Sudoku ini sebenarnya cukup sederhana yaitu pemain mengisi sebuah puzzle menggunakan bilangan acak. tetapi meskipun sederhana ternyata banyak pemain yang tidak sanggup menyelesaikannya. Dengan memanfaatkan algoritma Backtracking maka permasalahan puzzle pada permainan sudoku tersebut dapat terselesaikan dengan mudah dan dalam hitungan detik. Algoritma Backtracking adalah sebuah bentuk dari pemecahan masalah yang termasuk dalam strategi berbasis pada bentuk pencarian ruang solusi, namun tidak harus memeriksa seluruh kemungkinan yang ada, hanya yang mengarah kepada solusi saja yang akan di proses [3] Algoritma backtracking

sebenarnya juga merupakan algoritma yang bekerja secara rekursif, dimana proses pencariannya berbasis pada algoritma Depth-First Search (DFS) yaitu melakukan pencarian solusi persolan secara sistematis pada semua kemungkinan solusi yang ada dan pencarian solusi dilakukan dengan menelusuri suatu struktur berbentuk pohon berakar. Karena itu algoritma ini cukup bagus dan sangat baik untuk diterapkan dalam penyelesaian masalah dan untuk memberikan kecerdasan buatan pada permainan. Beberapa jenis permainan-permainan digital/game yang umumnya biasa dikenal oleh masyarakat seperti Catur, Labirin Matematika, Tic Tac Toe, hingga Sudoku dapat dicari solusinya dengan mengimplementasikan algoritma backtracking [4]. Penelitian serupa juga telah dilakukan oleh Febri Utama, Awang Harsa Kridalaksana dan Indah Fitri Astuti yang berjudul Implementasi Backtracking Algorithm untuk penyelesaian permainan sudoku pola 9x9. Tujuan dari penelitian tersebut adalah untuk membuat suatu aplikasi yang merupakan implementasi dari metode backtracking algorithm untuk menampilkan solusi Su Doku pola 9x9 yang unik dan membuktikan teori ketepatan metode backtracking algorithm dalam penggunaan pencarian solusi Su Doku pola 9x9. Implementasi dari metode backtracking algorithm dibuat dengan bahasa pemrograman Delphi. Hasil penelitian tersebut adalah sebuah aplikasi pencarian solusi Su Doku dengan pola 9x9, yang diberi nama "Program Aplikasi Sudoku Solver Backtracking Algorithm". Didalam aplikasi ini user menginputkan angka awal sesuai dengan level Su Doku, dimana hasil outputnya adalah tampilan dari solusi unik Su Doku [5]. Perbedaan pada penelitian ini adalah adanya fasilitas berupa

lama waktu yang diperlukan untuk menyelesaikan permainan tersebut. Pemanfaatan algoritma backtracking juga dilakukan oleh Abdul Kadir Hasani dan Hindayati Mustafidah yang berjudul Algoritma Backtracking Untuk Penyelesaian Puzzle Gambar Bendera. pada penelitian tersebut meneliti tentang bagaimana algoritma backtracking dapat menyelesaikan permainan puzzle gambar bendera negara-negara. terdapat 3 level kesulitan yaitu tingkat mudah dengan jumlah 10 kotak, tingkat sedang dengan jumlah 20 kotak dan tingkat sulit dengan jumlah 30 kotak. hasil penelitian tersebut membuktikan bahwa algoritma backtracking dapat diimplementasikan pada permainan tersebut. [6]. Pemanfaatan Algoritma Backtracking juga dilakukan oleh Akbar Serdano, Muhammad Zarlis dan Dedy Hartama yang berjudul Implementasi Algoritma Backtracking pada Knight's Tour Problem. pada penelitian tersebut berhasil memanfaatkan algoritma backtracking pada permainan catur. Namun hanya langkah kuda saja yang dititikberatkan pada permainan catur pada penelitiannya. penelitian tersebut memprediksi langkah-langkah kuda pada target sasaran yang diinginkan [7].

## BAHAN DAN METODE

Bahan dan metode dilakukan untuk melakukan penelitian agar berjalan dengan lancar dan tidak ada hambatan. Adapaun bahan dan metode yang digunakan adalah sebagai berikut.

### 1. Collecting Referensi

Penelitian ini dilakukan dengan langkah pertama adalah mengumpulkan data-data referensi terkait dengan algoritma backtracking.

### 2. Mengolah Data

Setelah proses pengumpulan data selesai maka tahapan berikutnya adalah mengolah data penelitian, dalam hal ini data berbentuk angka yang dimulai dari 1 sampai dengan 9, diluar angka itu maka data ditolak.

### 3. Mengembangkan Sistem

Pada tahapan ini akan membahas mengenai perancangan dari bentuk sistem yang terkait. Perancangan dilakukan untuk menggambarkan langkah demi langkah aplikasi yang hendak dibuat.

### 4. Implementasi Sistem

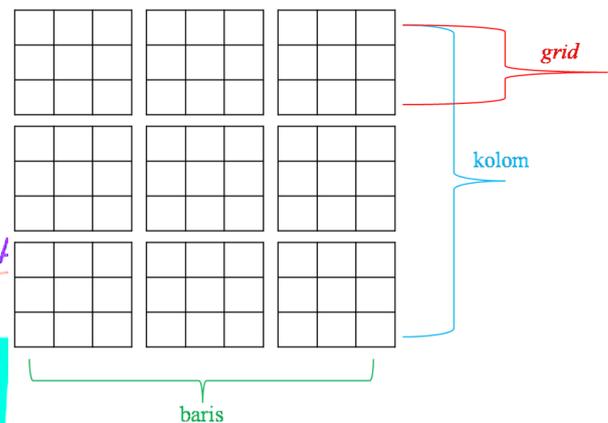
Setelah proses pengembangan sistem berhasil dilakukan maka langkah berikutnya adalah mengimplementasikan sistem yang telah berhasil dibuat.

### 5. Pengujian Sistem

Langkah terakhir adalah menguji sistem yang telah berhasil dibangun. Pada tahapan ini sistem akan menguji inputan data angka menggunakan algoritma backtracking. Angka-angka yang boleh diuji dan dimasukan adalah 1-9.

## SUDOKU

Sudoku adalah permainan digital yang termasuk kedalam genre puzzle game. Sudoku biasanya dikenal sebagai permainan teka-teki angka berbasis logika yang bertujuan untuk mengisi sebuah puzzle dengan angka acak. biasanya berukuran 3x3, 4x4, hingga 9x9 [8]. Pada permainan sudoku diperlukan kecermatan, kesabaran serta logika didalam menyelesaikan puzzle tersebut. bentuk umum pada sudoku terdiri atas matriks yang berjumlah sembilan baris dan sembilan buah kolom. dan pada matriks tersebut terdapat sebuah blok yang terdiri atas matriks 3x3. Seperti pada gambar 1.



Gambar 1. Elemen Matrik Sudoku

Biasanya saat awal permainan matrik akan diberikan angka awal terlebih dahulu sebagai angka pembantu atau biasanya dikenal dengan elemen awal puzzle. [9] yang akan ditunjukkan pada gambar 2.

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

Gambar 2. Elemen awal puzzle.

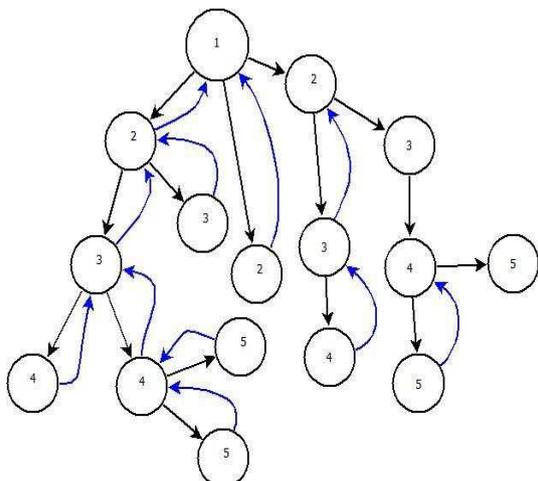
Pada gambar 2, Angka-angka pada beberapa baris dan kolom akan terisi secara otomatis pada awal permainan. Setiap permainan baru akan dimulai maka baris dan kolom pada posisi tertentu (acak) akan terisi dengan angka secara otomatis dan nilai dari angka tersebut akan random. Dan jika keseluruhan kotak tersebut telah terisi dengan angka secara benar permainan tersebut telah selesai. Berikut adalah contoh jika kotak pada puzzle sudoku telah terisi secara menyeluruh dan benar pada gambar 3.

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

Gambar 3. Puzzle yang telah terisi

**Backtracking**

Algoritma backtracking (runut balik) merupakan salah satu metode pemecahan masalah yang termasuk dalam strategi yang berbasis pencarian pada ruang solusi, namun tidak harus memeriksa seluruh kemungkinan yang ada, hanya yang mengarah kepada solusi saja yang akan di proses. Algoritma backtracking sebenarnya juga merupakan algoritma yang bekerja secara rekursif, dimana proses pencariannya berbasis pada algoritma Depth-First Search (DFS) yaitu melakukan pencarian solusi persoalan secara sistematis pada semua kemungkinan solusi yang ada dan pencarian solusi dilakukan dengan menelusuri suatu struktur berbentuk pohon berakar [10]. Berikut adalah contoh tree dari algoritma backtracking seperti pada gambar 4.



Gambar 4. Bactracking tree

Berdasarkan gambar 4, pada saat proses pencarian dilakukan dengan cara mengunjungi salah satu cabang terlebih dahulu sampai dengna simpul terakhir (1 sampai 4). Jika tujuan yang hendak dicari belum didapatkan maka proses berikutnya adalah dengan melanjutkan Implementasi Algoritma Backtracking Pada Permainan Sudoku. Oleh :Fauzan Azima Rahma, Dian Anubhakti

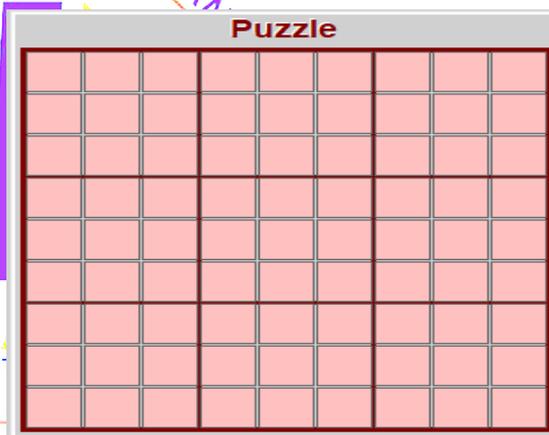
pencarian ke salah satu cabang sebelumnya(2 sampai 3), dan turun hingga ke bawah jika memang masih terdapat cabang lainnya. Begitu seterusnya hingga diperoleh tujuan akhirnya (1 sampai 5). Dengan metode backtracking ini, dalam proses pencariannya tidak mengarah kepada semua kemungkinan solusi yang ada, hanya mengarah kepada satu solusi saja. Akibatnya rentang waktu dalam pencarian solusi lebih sedikit sehingga waktu yang dibutuhkan menjadi efektif dan efesien

**HASIL DAN PEMBAHASAN**

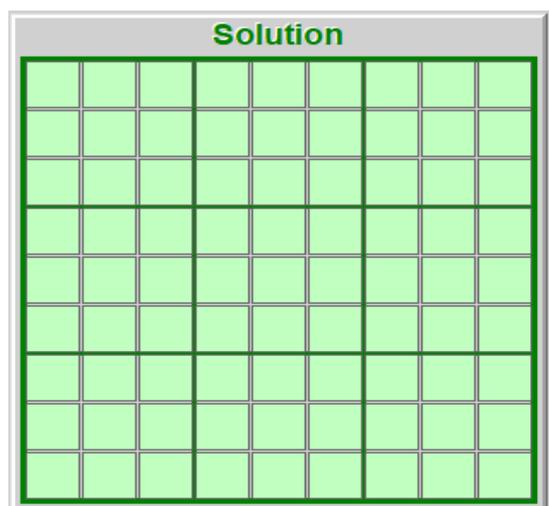
**Pembahasan**

**User Interface**

Pada user interface merupakan bentuk form dari program yang akan dijalankan. Terdapat 2 buah form yaitu form puzzle digunakan untuk memasukan angka dan form solusi digunakan untuk menampilkan angka-angka secara otomatis berdasarkan algoritma backtracking.



Gambar 5. Form Puzzle

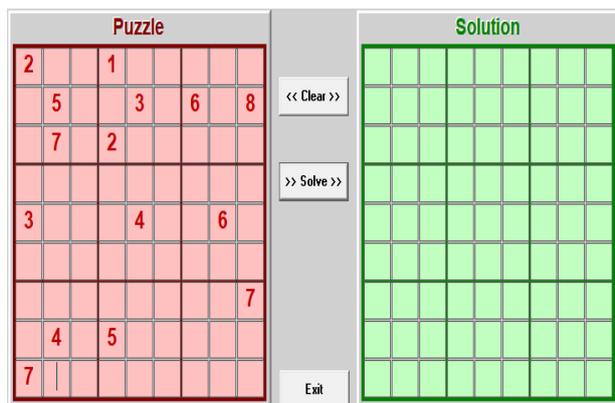


Gambar 6. Form Solution

**Uji coba Puzzle**

Untuk mencoba puzzle maka kita jalankan program dan nilai awal akan terisi secara

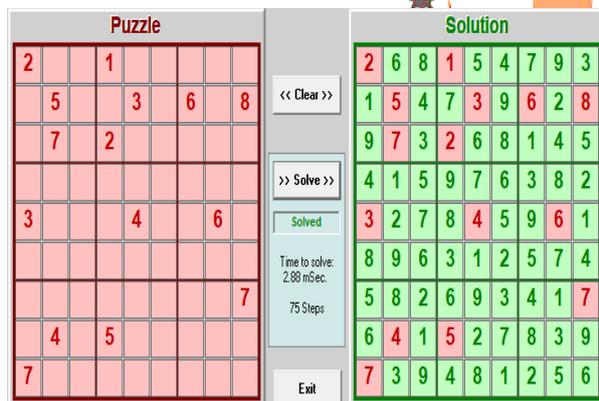
otomatis pada form seperti pada Gambar 7 berikut.



Gambar 7. Nilai awal pada puzzle.

Pada Gambar 7. Terdapat beberapa tombol yaitu Clear untuk membersihkan kotak-kotak angka pada form, tombol solve digunakan untuk mendapatkan angka secara otomatis menggunakan algoritma backtracking dan tombol Exit untuk keluar dari form.

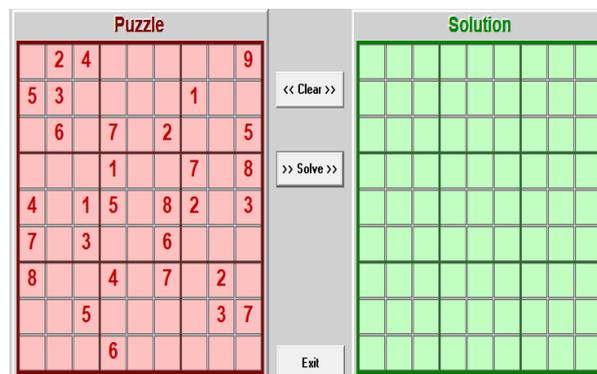
Kemudian untuk menampilkan solusi dari permasalahan puzzle tersebut maka kita tekan tombol solve. Maka akan tampil solusi seperti pada Gambar 8.



Gambar 8. Angka Solution yang telah terisi .

Pada form solusi pada Gambar 8. akan terisi secara otomatis nilai-nilai menggunakan algoritma backtracking.

Pada aplikasi juga terdapat menu save permainan, ketika kita sudah tidak dapat memikirkan angka mana saja yang cocok, maka dapat kita save dan kita lanjutkan nanti. Dan pada menu load permainan kita pilih permainan yang kita simpan. Berikut Gambar 9. Menu load permainan.



Gambar 9. Load Puzzle.

Dan jika kita sudah menyerah dan ingin mengakhiri permainan maka kita pilih tombol solve dan secara otomatis akan terisi kotak-kotak kosong tersebut, seperti pada Gambar 10.



Gambar 10. Solusi form pada load puzzle

Potongan-script algoritma Backtracking.

```

With Cells(LastHit)
    If .Value = 0 Then 'free cell
        Pattern =
        Groups(.ColumnNumber).PermitPattern And _
        Groups(.RowNumber).PermitPattern And _
        Groups(.BlockNumber).PermitPattern
        CurPermit = 0
        For k = 1 To 9
            If Pattern And Bits(k) Then
                CurPermit = CurPermit + 1
            End If
        Next k
        If CurPermit < MinPermit Then
            MinPermit = CurPermit
            PermissionBits = Pattern
        End If
    End If
End With
    
```

## KESIMPULAN

Setelah mengamati dan mempelajari penelitian tersebut maka dapat diambil kesimpulan sebagai berikut :

1. Algoritma Backtracking dapat diterapkan pada permainan Sudoku dengan baik.
2. Program yang dihasilkan mampu menampilkan waktu yang dimainkan dan berapa langkah yang diperlukan untuk menyelesaikan permainan tersebut.
3. Permainan dapat disimpan dan dimainkan kembali jika pemain merasa jenuh.

## DAFTAR PUSTAKA

- [1.] Sihombing DJC, Wirapraja A. Tren Penerapan Artificial Intelligence Pada Bidang Akuntansi, Energi Terbarukan Dan Proses Industri Manufaktur (Studi Literatur). *J Eksek.* 2018;15(2):302-315.
- [2.] Rahayu DS, Suryapratama A, Amongsaufa AZ, Koloay BIK. Evaluasi Algoritma Runut Balik Dan Simulated Annealing Pada Permainan Sudoku. *J Tek Inform dan Sist Inf.* 2017;3(1):169-178. doi:10.28932/jutisi.v3i1.592
- [3.] Rifqo MH, Apridiansyah Y. Implementasi Algoritma Backtracking Dalam Sistem Informasi Perpustakaan Untuk Pencarian
- [4.] Coastera FF, Nomansa A. Backtracking Pada Aplikasi Permainan Tradisional Dam-Daman.
- [5.] Utama F, Kridalaksana AH, Astuti IF. Implementasi Backtracking Algorithm Untuk Penyelesaian Permainan Su Doku Pola 9x9. *Inform Mulawarman J Ilm Ilmu Komput.* 2016;11(1):29. doi:10.30872/jim.v11i1.200
- [6.] Hasani AK, Mustafidah H. Algoritma Backtracking untuk Penyelesaian Puzzle Gambar Bendera ( Backtracking Algorithm for Completing Puzzle Flag ). *Juita.* 2016;IV(November):87-95.
- [7.] Serdano A, Zarlis M. Implementasi Algoritma Backtracking Pada Knight ' s Tour Problem. 2019;3.
- [8.] Wardani S. Jurnal Dinamika Informatika Volume 5, Nomor 1, November 2015. *J Din Inform.* 2015;5(November):1-13.
- [9.] Cowen R. A Beginner's Guide to Solving Sudoku Puzzles by Computer. *Math J.* 2018:368.
- [10.] Juliantus S, Sitorus L, Informatika T, St U, Setia J, No B. Implementasi Backtracking dan Shift Or pada Proses Pencarian Data Skripsi. 2019;04(479):39-45.

Judul Buku (Studi Kasus Unit Pelayanan Terpadu Perpustakaan Universitas Muhammadiyah Bengkulu). *Pseudocode.* 2017;4(1):90-96.

doi:10.33369/pseudocode.4.1.90-96

