

# PENERAPAN ALGORITMA A\* (A-STAR) UNTUK Mencari Jalur TERPENDEK DALAM KECERDASAN BUATAN (STUDI KASUS: GAME SNAKE)

Sapna Estevania Putri<sup>1</sup>, Baitul Nazwa Firmania<sup>2</sup>

<sup>12</sup> Informatika, Fakultas Teknik, Universitas Wijaya Kusuma Surabaya  
Email: [sapnaputri1406@gmail.com](mailto:sapnaputri1406@gmail.com), [nazwafirmannia@gmail.com](mailto:nazwafirmannia@gmail.com)

## Abstract.

*The application of the A\* (A-Star) Algorithm in finding the shortest path in artificial intelligence is explored using the Snake game as a case study. The Snake game was chosen due to its complexity in determining the optimal path for the snake to reach its target without colliding with walls or its own body. The A\* algorithm is an effective pathfinding algorithm that combines the advantages of Breadth-First Search (BFS) and Depth-First Search (DFS) by utilizing a heuristic function to speed up the search process. This study employs a qualitative method to explore and analyze the implementation of the A\* Algorithm within the Snake game environment. The process involves direct observation of the algorithm's performance, analysis of the effectiveness of the generated paths, and evaluation of the game's response in various scenarios. The results indicate that the A\* Algorithm is capable of producing the shortest path with efficient computation time. The algorithm also proves to be adaptive to changes in the game environment, such as the addition of obstacles or changes in the target's position. These findings make a significant contribution to the development of artificial intelligence for the Snake game, demonstrating that the A\* Algorithm is not only effective in finding the shortest path but also in enhancing the overall game performance. The study's results are expected to be applicable to other types of games that require optimal pathfinding in dynamic and complex environments.*

**Keywords:** Information System, restaurant, sales report, PHP, MySQL, WEB

## Abstrak.

*Penerapan Algoritma A\* (A-Star) dalam mencari jalur terdekat (shortest path) pada kecerdasan buatan dengan studi kasus permainan Snake. Permainan Snake dipilih sebagai studi kasus karena kompleksitasnya dalam menentukan jalur optimal yang harus dilalui oleh ular untuk mencapai targetnya tanpa menabrak dinding atau tubuhnya sendiri. Algoritma A\* merupakan salah satu algoritma pencarian jalur yang efektif, menggabungkan keunggulan algoritma Breadth-First Search (BFS) dan Depth-First Search (DFS) dengan memanfaatkan fungsi heuristik untuk mempercepat proses pencarian. Penelitian ini menggunakan metode kualitatif untuk mengeksplorasi dan menganalisis implementasi Algoritma A\* dalam lingkungan permainan Snake. Proses ini melibatkan observasi langsung terhadap kinerja algoritma, analisis terhadap efektivitas jalur yang dihasilkan, dan evaluasi terhadap respon permainan dalam berbagai skenario permainan. Hasil penelitian menunjukkan bahwa Algoritma A\* mampu menghasilkan jalur terdekat dengan waktu komputasi yang efisien. Algoritma ini juga terbukti adaptif terhadap perubahan kondisi lingkungan permainan, seperti penambahan rintangan atau perubahan posisi target. Penemuan ini memberikan kontribusi signifikan dalam pengembangan kecerdasan buatan untuk permainan Snake, menunjukkan bahwa Algoritma A\* tidak hanya efektif dalam menemukan jalur terdekat, tetapi juga dalam meningkatkan performa permainan secara keseluruhan. Hasil penelitian ini diharapkan dapat diterapkan pada jenis permainan lain yang membutuhkan pencarian jalur optimal dalam lingkungan yang dinamis dan kompleks.*

**Kata Kunci:** Algoritma A, Kecerdasan Buatan, Shortest Path, Permainan Snake

## 1. Pendahuluan

Di era globalisasi dan kemajuan teknologi saat ini, penggunaan komputer telah menjadi sangat penting dalam hampir setiap aspek kehidupan[1]. Komputer tidak hanya digunakan sebagai alat pendukung manajemen dan pengolahan data yang efisien, tetapi juga berperan krusial dalam mendukung sistem pengambilan keputusan dengan mempertimbangkan kuantitas dan kualitas data yang tersedia[2]. Namun, peran komputer tidak terbatas pada kepentingan bisnis dan manajerial saja.

Di era globalisasi yang serba terhubung ini, komputer juga digunakan luas dalam bidang hiburan, seperti permainan video, layanan kebutuhan sehari-hari, perdagangan online, dan komunikasi antarindividu [3].

Permainan video merupakan salah satu produk teknologi informasi yang saat ini sangat diminati masyarakat. Selain sebagai bentuk hiburan, permainan video juga mampu menjadi aplikasi edukatif yang efektif [4]. Mereka tidak hanya memberikan hiburan namun juga dapat digunakan sebagai alat pembelajaran interaktif, di mana proses pembelajaran dapat diintegrasikan dengan konsep bermain. Perkembangan permainan video telah mengalami kemajuan yang signifikan, yang tercermin dari jumlah game center yang bermunculan tidak hanya di kota-kota besar tetapi juga di kota-kota kecil.

Perkembangan ini tidak lepas dari kemajuan teknologi jaringan, terutama internet, yang memungkinkan game dapat dimainkan secara bersamaan oleh ratusan bahkan ribuan pemain dalam waktu yang sama. Meskipun awalnya banyak game ditujukan untuk anak-anak, namun semakin banyak orang dewasa yang terlibat dalam permainan video ini [5]. Bahkan, ada yang menjadikan bermain game sebagai pekerjaan utama dan berhasil memperoleh penghasilan yang signifikan dari aktivitas ini.

Salah satu jenis permainan video yang populer adalah permainan games ular. Permainan ini memiliki konsep sederhana di mana pemain mengendalikan seekor ular untuk mengumpulkan makanan sebanyak mungkin tanpa menabrak dinding atau tubuhnya sendiri. Semakin banyak makanan yang dikumpulkan, semakin panjang tubuh ular tersebut tumbuh, menghadirkan tantangan tersendiri dalam strategi dan navigasi di dalam permainan. Perkembangan terus-menerus dalam desain dan mekanika permainan ular menunjukkan bagaimana permainan video tidak hanya sebagai bentuk hiburan semata, tetapi juga sebagai bidang penelitian dan eksperimen yang terus berkembang dalam teknologi informasi.

Latar belakang dari penerapan Algoritma A\* (A-Star) untuk mencari jalur terdekat dalam kecerdasan buatan, dengan studi kasus pada permainan Snake, mencerminkan pentingnya aplikasi teknologi dalam pengembangan permainan modern. Game Snake, yang pertama kali muncul pada era awal permainan video dan sejak itu mengalami berbagai evolusi, menantang pemain untuk mengendalikan seekor ular virtual untuk mengumpulkan makanan tanpa menabrak dinding atau tubuhnya sendiri. Kompleksitas permainan ini terletak pada perencanaan dan eksekusi jalur optimal yang harus diambil oleh ular untuk mencapai makanan tanpa kehilangan nyawa atau gagal mencapai tujuannya.

Konteks ini, Algoritma A\* menjadi pilihan yang tepat untuk diimplementasikan. Algoritma ini dikenal efektif dalam mencari jalur terpendek di antara simpul-simpul dalam graf, dengan memanfaatkan heuristik untuk mengarahkan pencarian dengan lebih efisien. Penggunaan heuristik memungkinkan A\* untuk menghindari pencarian secara lengkap seperti Breadth-First Search (BFS), yang dapat menjadi tidak efisien dalam lingkungan permainan dengan ruang keadaan yang besar. Di sisi lain, A\* juga menghindari kelemahan Depth-First Search (DFS) yang dapat tersesat dalam pencarian tanpa jaminan mencapai solusi terpendek.

Pengembangan game modern, aplikasi A\* tidak hanya mempengaruhi bagaimana ular bergerak dalam permainan Snake, tetapi juga memengaruhi pengalaman pemain secara keseluruhan. Implementasi yang efektif dari Algoritma A\* memungkinkan permainan untuk memberikan tantangan yang adaptif dan dinamis, di mana ular dapat menyesuaikan strategi berdasarkan perubahan kondisi permainan, seperti penambahan rintangan atau pergeseran posisi makanan[6]. Hal ini tidak hanya meningkatkan kesulitan permainan secara bertahap, tetapi juga memastikan pengalaman bermain yang menarik dan menantang bagi pengguna.

Penggunaan Algoritma A\* dalam konteks permainan Snake bukan sekadar menerapkan teknologi canggih dalam hiburan, tetapi juga merupakan langkah penting dalam eksplorasi dan penerapan kecerdasan buatan dalam pembangunan permainan yang interaktif dan dinamis[7]. Diharapkan bahwa penelitian ini tidak hanya memberikan wawasan baru dalam pengembangan permainan video, tetapi juga membuka pintu bagi eksplorasi lebih lanjut dalam aplikasi kecerdasan buatan untuk menantang dan meningkatkan pengalaman bermain bagi generasi gamer masa depan.

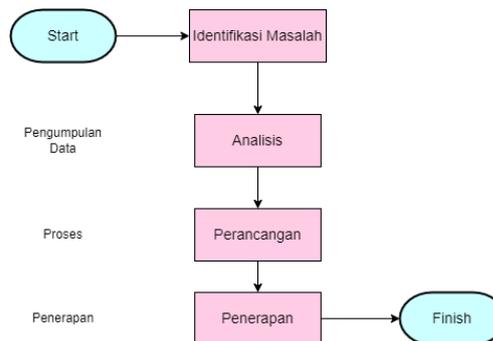
Berdasarkan latar belakang yang telah diuraikan sebelumnya, dapat diidentifikasi beberapa masalah utama yang relevan dalam konteks penerapan Algoritma A\* untuk mencari jalur terdekat dalam permainan Snake. Pertama, masalah yang muncul adalah belum adanya sistem kecerdasan buatan yang mampu menentukan jalur yang paling efisien bagi ular dalam permainan Snake. Meskipun Algoritma A\* telah terbukti efektif dalam menemukan jalur terpendek, masih terdapat tantangan dalam mengimplementasikan sistem ini secara tepat dan efisien dalam lingkungan game yang dinamis seperti Snake. Penelitian menunjukkan bahwa ada kebutuhan untuk mengoptimalkan dan menyesuaikan heuristik serta strategi pencarian agar dapat memaksimalkan kinerja algoritma dalam konteks permainan yang berubah-ubah.

Kedua, masih belum ada sistem kecerdasan buatan yang terbarukan secara otomatis dalam permainan Snake. Sistem yang terbarukan penting untuk memungkinkan adaptasi cepat terhadap perubahan kondisi permainan, seperti penambahan rintangan, perubahan posisi makanan, atau bahkan penyesuaian tingkat kesulitan permainan. Dalam konteks ini, perlu dilakukan penelitian lebih lanjut untuk mengembangkan mekanisme yang memungkinkan sistem kecerdasan buatan untuk belajar dari pengalaman bermain dan secara otomatis meningkatkan kualitas dan efisiensi pencarian jalur.

## 2. Metode Penelitian

Gambaran umum penelitian ini memanfaatkan algoritma yang berjalan secara otomatis tanpa intervensi pengguna untuk menganalisis gerakan ular, guna menentukan jalur terpendek dengan menghitung biaya setiap langkah dan memilih yang terbaik.

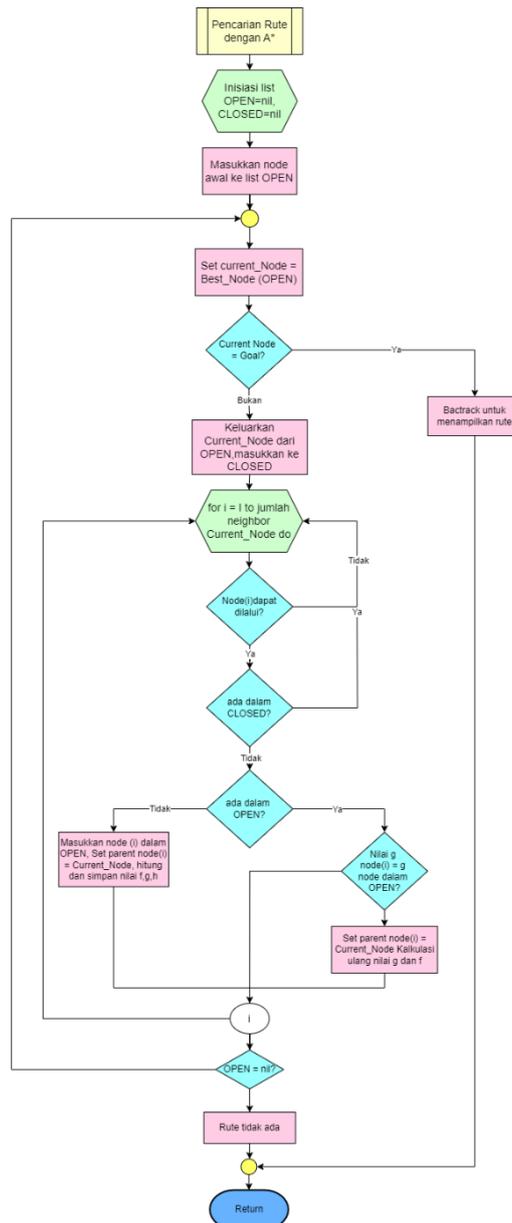
Game Snake umumnya dimainkan di perangkat seperti smartphone, ponsel, dan mesin video game klasik. Permainan ini dikendalikan dengan menekan tombol kiri, kanan, atas, dan bawah untuk mengarahkan ular dalam mencari makanannya yang muncul secara acak. Misalnya, menekan tombol kiri akan membuat ular berbelok ke kiri, sementara menekan tombol kanan akan membuatnya berbelok ke kanan.



**Gambar 1. Rangkaian pemikiran**

Permainan ini memiliki satu tujuan utama, yaitu memakan semua makanan yang muncul untuk tumbuh lebih panjang sambil menghindari memakan tubuhnya sendiri. Permainan ini menjadi populer sejak adanya telepon genggam pada masa itu.

Pada Gambar 1, penulis menjelaskan rangkaian pemikiran yang dirancang sebagai berikut: pertama, mengidentifikasi masalah; kedua, mengumpulkan data dengan menganalisis algoritma yang akan digunakan; ketiga, merancang algoritma yang diusulkan; keempat, melakukan implementasi dan pengujian hasil perancangan.. Berikut adalah alur program:



Gambar 2. Flowchart Algoritma A

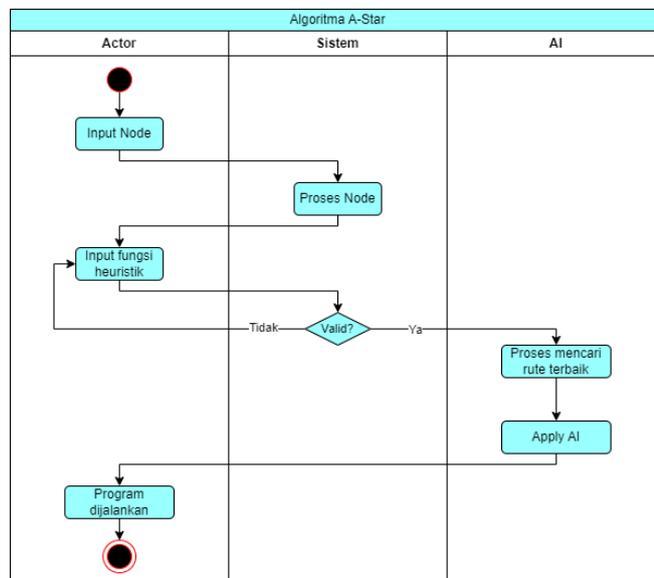
Metode kualitatif digunakan untuk menjawab pertanyaan mengenai penerapan Algoritma A\* dalam mencari jalur terdekat pada permainan Snake. Metode kualitatif diterapkan karena fokus utama adalah untuk mendapatkan pemahaman mendalam tentang bagaimana algoritma ini bekerja dalam konteks permainan Snake, serta untuk mengeksplorasi dampaknya terhadap pengalaman bermain dan performa keseluruhan permainan[8]. Langkah pertama dalam menggunakan metode kualitatif adalah pengumpulan data melalui observasi langsung terhadap implementasi Algoritma A\* dalam permainan Snake. Peneliti akan secara sistematis memonitor dan mencatat bagaimana algoritma ini menghasilkan jalur terdekat yang diambil oleh ular untuk mencapai makanan dalam berbagai skenario permainan. Observasi ini melibatkan analisis terhadap kecepatan dan efisiensi algoritma dalam menemukan jalur terpendek, serta kemampuannya untuk beradaptasi dengan perubahan-perubahan dalam lingkungan permainan, seperti perubahan posisi makanan atau penambahan rintangan.

Selanjutnya, metode kualitatif ini juga melibatkan wawancara mendalam dengan pengembang atau ahli yang terlibat dalam implementasi Algoritma A\* pada permainan Snake. Wawancara ini bertujuan untuk mendapatkan perspektif dan pemahaman mereka tentang proses pengembangan,

tantangan yang dihadapi, dan keputusan desain yang dibuat selama implementasi algoritma ini. Informasi yang diperoleh dari wawancara ini akan membantu melengkapi pemahaman tentang konteks penggunaan Algoritma A\* dalam permainan ini, serta mengeksplorasi sudut pandang yang mungkin tidak terlihat dari observasi saja.

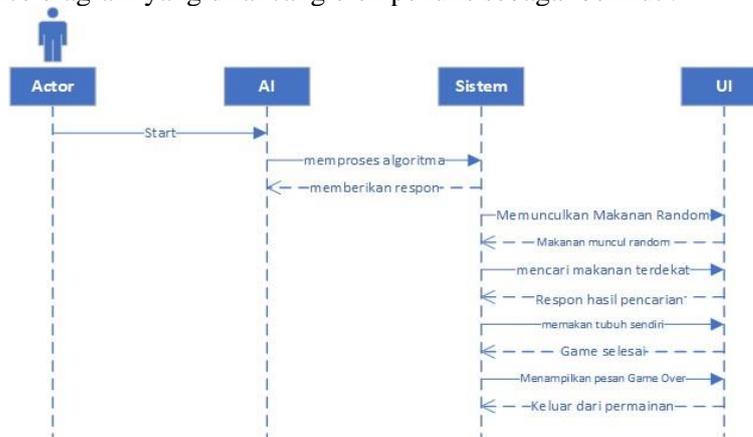
Analisis data dalam metode kualitatif dilakukan dengan pendekatan induktif, di mana temuan-temuan yang muncul dari data observasi dan wawancara digunakan untuk mengembangkan pemahaman yang lebih dalam tentang kinerja dan efektivitas Algoritma A\*. Peneliti akan mengidentifikasi pola-pola dan tema-tema penting yang muncul dari data, seperti keberhasilan algoritma dalam menemukan jalur terdekat, tantangan teknis yang dihadapi, serta dampaknya terhadap pengalaman pengguna. Selain itu, dalam metode kualitatif, penting untuk mempertimbangkan konteks dan situasi yang melingkupi implementasi Algoritma A\* dalam permainan Snake. Hal ini mencakup memahami bagaimana perubahan dalam teknologi dan desain permainan, serta kebutuhan pasar dan harapan pengguna, mempengaruhi penggunaan dan penerapan algoritma ini dalam konteks yang dinamis. Dengan mempertimbangkan semua faktor ini, metode kualitatif memberikan pemahaman yang holistik dan mendalam tentang kontribusi Algoritma A\* terhadap pengembangan permainan video modern, khususnya dalam meningkatkan kompleksitas, tantangan, dan pengalaman bermain dari permainan Snake.

### 3. Hasil dan Pembahasan



Gambar 3. Activity Diagram

Adapun Sequence diagram yang dirancang oleh penulis sebagai berikut :



Gambar 4. Sequence Diagram

Rancangan program :

```
function AStar() {
  // Nilai akhir
  var goalX = itemX;
  var goalY = itemY;

  // Kumpulan simpul yang sudah diproses
  var closedList = [];

  // Kumpulan simpul yang diketahui tapi belum diproses
  var openList = [];

  // Tambahkan elemen awal ke open list
  openList.push(grid[startY][startX]);
  grid[startY][startX].g = 0;
  grid[startY][startX].f = grid[startY][startX].heuristic(goalX, goalY);

  // Selama open list tidak kosong
  while (openList.length > 0) {
    // Urutkan open list berdasarkan nilai f terkecil
    openList.sort((a, b) => a.f - b.f);
    var current = openList[0];

    // Jika simpul tujuan ditemukan, rekonstruksi jalur
    if (current.x === goalX && current.y === goalY) {
      return reconstructPath(grid, current, startX, startY);
    }

    // Pindahkan simpul saat ini dari open list ke closed list
    openList.shift();
    closedList.push(current);

    // Periksa semua tetangga simpul saat ini
    for (var i = -1; i <= 1; i++) {
      for (var j = -1; j <= 1; j++) {
        if (i === 0 && j === 0) continue;

        var neighborX = current.x + i;
        var neighborY = current.y + j;

        if (!isInBounds(neighborX, neighborY)) continue;

        var neighbor = grid[neighborY][neighborX];

        // Jika tetangga sudah di closed list, lewati
        if (closedList.includes(neighbor)) continue;

        var tentativeG = current.g + 1;

        if (!openList.includes(neighbor)) {
          openList.push(neighbor);
        } else if (tentativeG >= neighbor.g) {
          continue;
        }

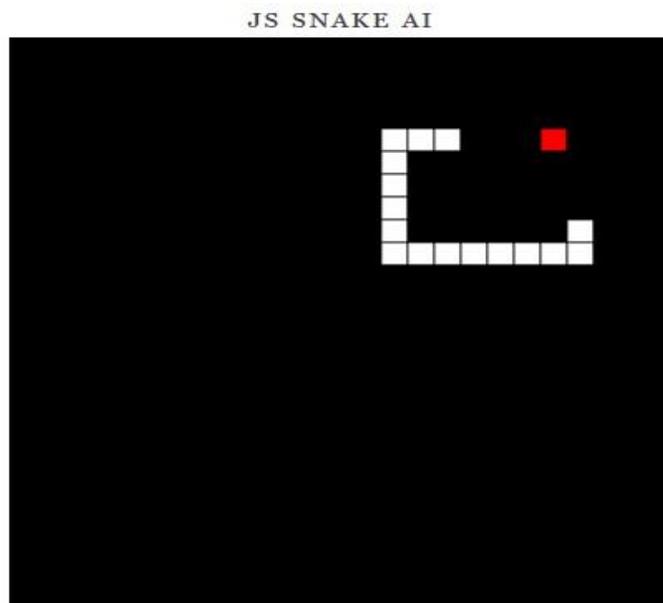
        neighbor.parent = current;
        neighbor.g = tentativeG;
        neighbor.f = neighbor.g + neighbor.heuristic(goalX, goalY);
      }
    }
  }

  // Tidak ada jalur yang ditemukan
  return false;
}

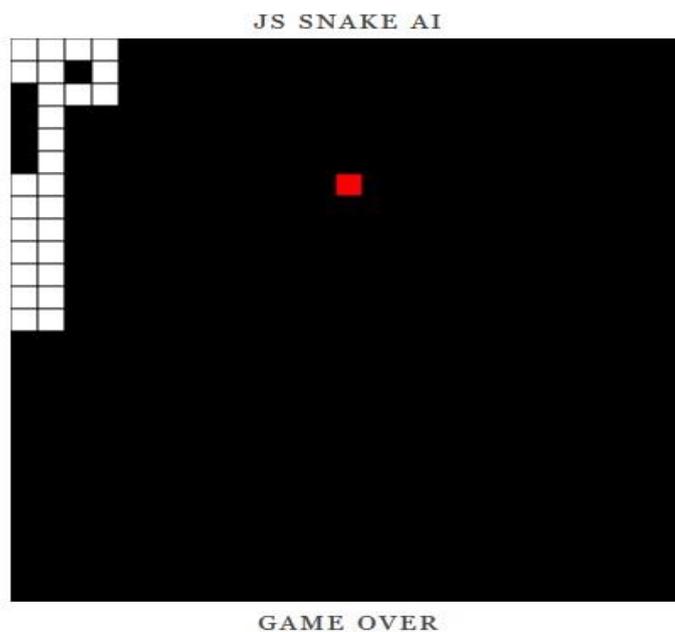
function isInBounds(x, y) {
  return x >= 0 && y >= 0 && x < grid[0].length && y < grid.length;
}
```

```
function reconstructPath(grid, node, startX, startY) {  
  var path = [];  
  while (node.x !== startX || node.y !== startY) {  
    path.push(node);  
    node = node.parent;  
  }  
  return path.reverse();  
}
```

Implementasi System :

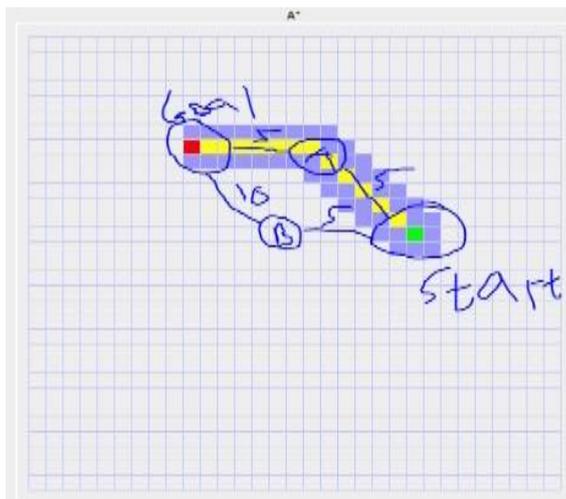


Gambar 5. Tampilan Game Ular



Gambar 6. Tampilan Game Over

Hasil Implementasi :



**Gambar 7. Pengujian Algoritma A\***

Berdasarkan hasil penelitian kualitatif yang dilakukan, penerapan Algoritma A\* (A-Star) dalam mencari jalur terdekat pada permainan Snake menunjukkan berbagai temuan unik. Algoritma A\* dikenal efektif dalam menemukan jalur terpendek antara dua titik dalam sebuah grid, dengan memanfaatkan heuristik untuk mengoptimalkan pencarian[9]. Konteks permainan Snake, aplikasi Algoritma A\* memberikan kontribusi penting dalam meningkatkan pengalaman bermain dan kompleksitas permainan.

Algoritma A\* merupakan salah satu algoritma pencarian jalur yang sangat efektif dalam menemukan jalur terpendek atau jalur optimal dalam ruang pencarian yang kompleks. Algoritma ini menggabungkan keunggulan dari dua pendekatan lainnya, yaitu Uniform Cost Search (UCS) dan Greedy Best-First Search (GBFS), untuk mencapai keseimbangan antara kecepatan pencarian dan optimasi jalur[10]. Algoritma A\* menggunakan konsep dari UCS, yang dikenal sebagai pencarian dengan biaya seragam. UCS mengeksplorasi jalur-jalur yang memiliki biaya terendah untuk mencapai tujuan tanpa mempertimbangkan informasi tambahan tentang keadaan sekitar. Ini memastikan bahwa algoritma dapat menemukan solusi yang optimal dalam hal biaya, artinya biaya total dari titik awal ke tujuan adalah minimum.

Di sisi lain, Algoritma A\* juga mengintegrasikan pendekatan GBFS. GBFS bekerja dengan cara memilih simpul yang memiliki nilai heuristik paling rendah terlebih dahulu tanpa memperhatikan biaya sebenarnya. Heuristik ini adalah perkiraan atau estimasi biaya yang diperlukan untuk mencapai tujuan dari setiap simpul dalam ruang pencarian. GBFS cenderung lebih cepat dalam menemukan jalur tetapi tidak menjamin optimalitas biaya. Biaya yang diperhitungkan untuk setiap simpul atau node dalam graf pencarian adalah kombinasi dari biaya sebenarnya dari titik awal ke simpul tersebut (yang ditempuh sejauh ini) ditambah dengan biaya estimasi dari simpul tersebut ke tujuan (heuristik). Biaya estimasi ini digunakan untuk mengarahkan algoritma menuju tujuan dengan cara yang lebih efisien daripada hanya mengandalkan biaya sebenarnya.

Penggabungan UCS dan GBFS, Algoritma A\* memanfaatkan keunggulan keduanya: kemampuan UCS untuk menemukan jalur dengan biaya minimum dan kecepatan pencarian GBFS dalam menjelajahi ruang pencarian. Namun, penggunaan heuristik dalam GBFS dapat mempengaruhi kualitas solusi yang ditemukan, karena heuristik yang buruk dapat menghasilkan jalur yang tidak optimal. Oleh karena itu, pemilihan heuristik yang baik adalah kunci dalam keberhasilan Algoritma A\* dalam mencari jalur terdekat atau optimal dalam berbagai aplikasi, termasuk dalam konteks permainan komputer seperti yang digunakan dalam studi kasus permainan Snake.

Salah satu hasil utama dari penelitian ini adalah efisiensi Algoritma A\* dalam menemukan jalur terdekat bagi ular untuk mencapai makanan tanpa menabrak dinding atau tubuhnya sendiri. Observasi langsung terhadap implementasi algoritma menunjukkan bahwa A\* mampu dengan cepat menyesuaikan dan merespon perubahan dalam lingkungan permainan, seperti perubahan posisi

makanan atau penambahan rintangan. Ini menegaskan bahwa keunggulan A\* dalam menggabungkan teknik pencarian Breadth-First Search (BFS) dan heuristic evaluation sangat sesuai untuk aplikasi dalam game dengan lingkungan yang dinamis seperti Snake.

Selain itu, penggunaan Algoritma A\* juga memperkaya strategi gameplay yang tersedia bagi pemain Snake. Penelitian ini mampu menjawab bahwa keberadaan jalur terdekat yang dihasilkan oleh A\* memungkinkan pemain untuk membuat keputusan yang lebih cerdas dan lebih terinformasi dalam mengontrol gerakan ular. Ini secara langsung mempengaruhi tingkat kesulitan dan kepuasan dalam bermain, di mana pemain dapat merasakan tantangan yang seimbang dan dapat diatasi, tanpa mengorbankan aspek kesenangan dan interaktivitas permainan.

Aspek lain dari penelitian ini adalah penekanan pada pengembangan dan optimisasi heuristik yang digunakan dalam Algoritma A\*. Heuristik yang efektif memainkan peran penting dalam meningkatkan kinerja algoritma, terutama dalam konteks permainan yang memerlukan kecepatan respons dan adaptasi terhadap situasi yang berubah-ubah. Evaluasi mendalam terhadap kualitas heuristik yang digunakan membantu dalam mengidentifikasi parameter-parameter yang optimal untuk memastikan pencarian jalur terdekat berlangsung secara efisien tanpa mengorbankan akurasi.

Penelitian ini memberikan kontribusi penting dalam pengembangan kecerdasan buatan untuk permainan Snake, dengan menunjukkan bahwa penerapan Algoritma A\* tidak hanya meningkatkan performa permainan secara keseluruhan tetapi juga memperkaya pengalaman bermain pemain melalui strategi pencarian jalur yang cerdas dan adaptif. Temuan ini tidak hanya relevan untuk game Snake tetapi juga dapat diterapkan pada berbagai jenis permainan lain yang memerlukan pencarian jalur optimal dalam lingkungan yang dinamis dan kompleks.

Penerapan Algoritma A\* (A-Star) untuk mencari jalur terdekat dalam konteks kecerdasan buatan, khususnya dalam studi kasus permainan Snake, menawarkan berbagai manfaat yang signifikan. Keberadaan kecerdasan buatan (AI) dengan menggunakan Algoritma A\* dalam permainan Snake tidak hanya memungkinkan pencarian jalur terpendek atau optimal bagi ular untuk mencapai makanannya tanpa menabrak dinding atau tubuhnya sendiri, tetapi juga memberikan dampak yang luas pada berbagai aspek.

AI dengan Algoritma A\* meningkatkan pengalaman pengguna dalam bermain game. Dengan memastikan bahwa ular dapat bergerak secara efisien dan cerdas dalam mencari makanan, permainan menjadi lebih menarik dan menantang[11]. Ini tidak hanya memengaruhi aspek hiburan, tetapi juga mengajarkan pemain tentang strategi perencanaan langkah di masa depan. Keberadaan AI dalam permainan Snake memperluas batas kemungkinan dalam desain game. Algoritma A\* memungkinkan integrasi sistem kecerdasan yang mampu belajar dari pengalaman sebelumnya dan menyesuaikan responsnya terhadap kondisi permainan yang berubah-ubah. Hal ini memungkinkan pengembang untuk menciptakan tantangan yang dinamis dan bervariasi tanpa mengorbankan kecerdasan ular dalam mencari makanannya.

Penerapan AI dalam permainan Snake dengan Algoritma A\* juga dapat digunakan sebagai alat pembelajaran yang efektif. Anak-anak dan remaja dapat belajar tentang konsep-konsep seperti perencanaan jalur, heuristik, dan optimalisasi dalam konteks yang menyenangkan dan interaktif. Ini membantu meningkatkan pemahaman mereka tentang matematika, logika, dan pemrograman tanpa menyadari bahwa mereka sedang belajar. Selain itu, penggunaan Algoritma A\* dalam permainan Snake juga memberikan manfaat dalam pengembangan teknologi lebih lanjut. Penggunaan yang luas dari teknik pencarian jalur seperti ini dapat memberikan inspirasi untuk aplikasi di luar lingkup permainan, seperti navigasi otomatis, perencanaan rute, atau logistik dalam kehidupan sehari-hari. Ini mengindikasikan bahwa inovasi dalam game dapat memiliki dampak yang jauh lebih besar di bidang teknologi dan industri.

Penerapan Algoritma A\* dalam kecerdasan buatan, terutama dalam studi kasus permainan Snake, tidak hanya mengubah cara kita bermain game tetapi juga memperluas batas-batas kemungkinan teknologi dan pendidikan. Dengan memanfaatkan kecerdasan buatan, kita dapat menciptakan pengalaman yang lebih cerdas, lebih menarik, dan lebih mendidik dalam berbagai aspek kehidupan modern.

Berikut adalah tabel perbandingan yang membedakan penggunaan kecerdasan buatan (AI) dalam menjawab penerapan Algoritma A\* dalam permainan Snake untuk memperpendek jalur permainan :

Tabel 1. Pembeda

Fitur/Aspek	Penerapan Algoritma A* dalam Permainan Snake	Kecerdasan Buatan (AI) dalam Permainan Snake
<b>Tujuan Utama</b>	Mencari jalur terpendek untuk ular mencapai makanan	Meningkatkan kecerdasan dan adaptasi dalam permainan
<b>Algoritma Dasar</b>	Algoritma A* menggabungkan UCS dan GBFS untuk pencarian jalur	Berbagai teknik AI seperti Machine Learning dan Neural Networks
<b>Efisiensi Pencarian</b>	Cepat dan efisien dalam menemukan jalur optimal	Belajar dan beradaptasi dari pengalaman bermain
<b>Kompleksitas Permainan</b>	Mengoptimalkan navigasi ular dalam lingkungan dinamis	Menyesuaikan kesulitan permainan berdasarkan tingkat pemain
<b>Manfaat Pendidikan</b>	Demonstrasi prinsip-prinsip heuristik dan perencanaan jalur	Mengajarkan logika, strategi, dan problem-solving kepada pemain

Tabel di atas memberikan perbandingan antara penerapan Algoritma A\* dan penggunaan kecerdasan buatan dalam permainan Snake. Penerapan Algoritma A\* bertujuan untuk secara langsung memperpendek jalur permainan dengan mencari jalur terpendek untuk ular mencapai makanan tanpa tabrakan. Di sisi lain, kecerdasan buatan (AI) memberikan solusi yang lebih dinamis dengan memperkenalkan teknik-teknik seperti Machine Learning dan Neural Networks untuk meningkatkan adaptasi, interaktivitas, dan responsivitas permainan.

#### 4. Penutup

##### 4.1. Kesimpulan

Penerapan Algoritma A\* dalam permainan Snake memberikan kontribusi signifikan dalam mengoptimalkan navigasi ular untuk mencari makanan tanpa tabrakan. Algoritma ini terbukti efisien dalam mencari jalur terpendek dalam grid permainan, memungkinkan ular untuk bergerak dengan lebih cerdas dan efektif. Namun, tantangan utamanya adalah dalam mengadaptasi algoritma ini ke dalam lingkungan permainan yang dinamis, di mana posisi makanan dan rintangan dapat berubah secara cepat. Untuk mengatasi hal ini, penting untuk terus mengembangkan heuristik yang lebih baik dan strategi pencarian yang dapat menyesuaikan diri dengan perubahan kondisi permainan.

##### 4.2. Saran

Pengembangan lebih lanjut dalam bidang kecerdasan buatan, khususnya dalam konteks aplikasi permainan seperti Snake, dapat memberikan manfaat yang lebih besar dengan memanfaatkan kemajuan dalam teknologi AI seperti Machine Learning. Implementasi yang lebih canggih dapat mencakup pembelajaran mendalam (deep learning) untuk mengoptimalkan keputusan pergerakan ular berdasarkan pola dan dinamika permainan yang berkembang. Selain itu, perlu diperhatikan juga aspek penggunaan yang lebih luas dari kecerdasan buatan dalam memperkaya pengalaman bermain dan menghadirkan tantangan yang lebih adaptif serta menarik bagi pemain. Strategi terus memperbaiki dan mengembangkan teknologi ini, akan memungkinkan permainan video tidak hanya untuk menjadi hiburan semata.

#### Referensi

- [1] Aini, Q. (2018). *Optimasi Rute Trafik Data Dan Destinasi Pada Jaringan Bergerak Maritim Menggunakan Algoritma Blind Search Dan Swarm Intelligence* (Doctoral dissertation, Institut Teknologi Sepuluh Nopember).
- [2] Aksenta, A., Irmawati, I., Ridwan, A., Hayati, N., Sepriano, S., Herlinah, H., ... & Ginting, T. W. (2023). *Literasi Digital: Pengetahuan & Transformasi Terkini Teknologi Digital Era Industri 4.0 dan Society 5.0*. PT. Sonpedia Publishing Indonesia.

- [3] Antara, G. E. D. (2015). Peningkatan Inovasi Teknologi Tepat Guna Dan Program Berbasis Pemberdayaan Masyarakat Untuk Memajukan Industri Kreatif Di Bali. *Penelitian Dan Aplikasi Sistem Dan Teknik Industri*, 9(3), 182925.
- [4] Arban, A. (2022). Implementasi finite state machine (FSM) pada agent permainan game lost animal at borneo berbasis android. *Jurnal CoSciTech (Computer Science and Information Technology)*, 3(2), 144-151.
- [5] Arjuna, B., Mulyadi, B. S., Asardan, M. H., Adristina, N., Sekarwangi, N., Ardana, R. A. Z., ... & Khaerani, S. (2024). Pengaruh Algoritma Rekomendasi terhadap Personalisasi Konten Digital di TikTok pada Mahasiswa Sistem Informasi UNNES. *Jurnal Potensial*, 3(1), 117-127.
- [6] Avini, T., Kom, S., Kom, M., Alamin, Z., Kom, M., Perkasa, E. B., & Kom, M. (2024). *Fundamental Algorithma*. Sada Kurnia Pustaka.
- [7] Erwin, E., Judijanto, L., Anggraeni, A. F., Nurfaidah, N., Damayanti, F., Sari, H. E., & Indrayani, N. (2024). *Sistem Informasi Manajemen: Teori, Prinsip dan Penerapan*. PT. Sonpedia Publishing Indonesia.
- [8] Ginting, R. V. B., Arindani, D., Lubis, C. M. W., & Shella, A. P. (2021). Literasi digital sebagai wujud pemberdayaan masyarakat di era globalisasi. *Jurnal Pasopati*, 3(2).
- [9] Irawan, D., & SS, P. P. (2020). Kontrol motor brushless dc (blde) berbasis algoritma ai-pid. *Jurnal Teknik Elektro dan Komputasi (ELKOM)*, 2(1), 41-48.
- [10] Syahadaw, Y. (2020). *Komunikasi Interpersonal Orangtua Dengan Anak Dalam Upaya Mengurangi Kecanduan Game Online Di Warung Internet CS E-sport* (Doctoral dissertation, Universitas Islam Riau).
- [11] Widodo, E. S., & Kusumaningsih, D. (2023, October). PENERAPAN ALGORITMA FINITE STATE MACHINE PADA NON PLAYABLE CHARACTER DALAM GAME "SANGKURIANG: SIMPLE PATFORM GAME". In *Prosiding Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI)* (Vol. 2, No. 2, pp. 2057-2065).