

# Analisis Perbandingan Algoritma Linear Search dan Binary Search dalam Efisiensi Pencarian Data

<sup>1</sup>Hilman Firmansyah, <sup>2</sup>Eggi Julian, <sup>3</sup>Atep Ruhiat

<sup>1,2,3</sup>Universitas Sebelas April

<sup>1,2,3</sup>Jl. Angkrek Situ No.19, Sumedang, Jawa Barat 45323

email: [1240160121045@student.unsap.ac.id](mailto:1240160121045@student.unsap.ac.id), [eggi@unsap.ac.id](mailto:eggi@unsap.ac.id), [atepruhiat@unsap.ac.id](mailto:atepruhiat@unsap.ac.id)

## ABSTRACT

The rapid development of information technology has led to a significant increase in data volume, thus requiring information systems to perform data search processes quickly and efficiently. Search algorithms are a crucial component in determining system performance. This study aims to analyze and compare the efficiency of Linear Search and Binary Search algorithms in the data search process. The method used is a literature study with a descriptive and comparative approach to several relevant national journals. The results of the analysis show that Linear Search has the advantage in terms of flexibility because it does not require sorted data, but has a time complexity of  $O(n)$  making it less efficient for large datasets. In contrast, Binary Search has a time complexity of  $O(\log n)$  and has proven to be more efficient on large, sorted datasets. Therefore, the selection of a search algorithm must be adjusted to the characteristics and conditions of the data so that the system can work optimally.

**Kata kunci:** algoritma, linear search, binary search, efisiensi, data.

## 1. Introduction

Perkembangan teknologi informasi saat ini menghasilkan peningkatan volume data yang sangat cepat dan signifikan. Dalam berbagai sistem informasi, aktivitas pencarian data menjadi elemen kunci yang menentukan seberapa cepat dan efisien sistem dapat menemukan informasi yang diperlukan oleh pengguna. Rohman dan Fauziah menyatakan bahwa pencarian data yang tidak efektif dapat mengurangi kinerja sistem, terutama ketika data yang dikelola semakin besar dan rumit. Algoritma pencarian atau searching algorithm merujuk pada metode yang digunakan untuk menemukan data spesifik dalam sekumpulan data. Algoritma ini diaplikasikan di beragam bidang seperti sistem pendidikan, perpustakaan digital, manajemen inventaris, mesin pencari, hingga aplikasi ponsel. [1-3] menekankan bahwa pemilihan algoritma pencarian yang sesuai berdampak besar terhadap kecepatan pencarian dan pemrosesan data, terutama pada sistem yang fokus pada efisiensi waktu. Di antara berbagai algoritma pencarian yang ada, Linear Search dan Binary Search adalah dua metode dasar yang paling umum digunakan dan dibandingkan dalam banyak penelitian. Linear Search melakukan pencarian secara berurutan dengan memeriksa data satu per satu, sedangkan Binary Search membagi data menjadi dua bagian berulang untuk mempercepat pencarian. Meskipun Binary Search menawarkan kecepatan pencarian yang lebih tinggi, penggunaannya memerlukan data dalam urutan yang teratur. Ini berbeda dengan Linear Search yang lebih fleksibel meskipun kurang efisien untuk volume data yang besar [4]. Mengingat perbedaan karakteristik tersebut, penting untuk melakukan analisis lebih dalam mengenai kinerja kedua algoritma dalam berbagai kondisi dan situasi. Oleh karena itu, penelitian ini bertujuan untuk membandingkan algoritma Linear Search dan Binary Search dalam hal efisiensi pencarian data agar dapat menentukan algoritma mana yang paling optimal dalam kondisi tertentu. Studi mengenai

algoritma pencarian masih sangat signifikan dengan kemajuan teknologi saat ini, terutama karena ada peningkatan permintaan terhadap sistem yang dapat mengelola serta menemukan data dengan cepat. [5] menyatakan bahwa pemilihan algoritma pencarian yang tidak tepat dapat memperlambat proses pencarian, yang berdampak negatif pada kualitas dan efisiensi sistem informasi. Dalam berbagai sistem digital, algoritma pencarian memainkan peran yang sangat penting, khususnya pada aplikasi yang memerlukan respon cepat seperti sistem basis data, aplikasi inventaris, mesin pencari, serta sistem pendidikan berbasis web dan mobile[6-9]. menjelaskan bahwa memilih metode pencarian yang efisien dapat memberikan dampak besar dalam pengolahan data, terutama pada sistem dengan volume data yang besar dan rumit. Di samping itu, penelitian mengenai kinerja algoritma pencarian juga sangat penting untuk dilakukan, mengingat setiap algoritma memiliki ciri khas dan batasan yang berbeda. Algoritma seperti Linear Search lebih mudah digunakan pada dataset yang tidak terurut, sedangkan Binary Search lebih cepat namun memerlukan kondisi data yang sudah diurutkan[10-12]. Oleh sebab itu, analisis dan perbandingan antara kedua algoritma ini menjadi sangat penting untuk membantu para pengembang sistem atau peneliti dalam memilih metode pencarian yang paling sesuai dengan kebutuhan sistem yang sedang dibuat[13,14]. Hasil dari penelitian ini juga bisa dijadikan rujukan dalam mengambil keputusan terkait penerapan algoritma pencarian yang paling efektif di berbagai bidang teknologi informasi. Proses pencarian informasi sering kali menjadi elemen penting yang memengaruhi kinerja sebuah sistem, terutama ketika data yang dikelola berukuran besar dan terus bertambah. Namun, tidak semua metode pencarian menghasilkan hasil yang terbaik dalam setiap situasi data. Dari berbagai riset, muncul perbedaan karakteristik dan tingkat keefisienan antara metode Linear Search dan Binary Search, baik dari segi waktu pencarian maupun ketentuan penggunaannya. Oleh karena itu, penting untuk mengenali algoritma mana yang berfungsi lebih baik pada kondisi data tertentu.

**2. Research Method**

Metode penelitian yang digunakan dalam karya tulis ini adalah studi literatur (literature review) dengan pendekatan analisis deskriptif dan komparatif. Studi literatur dilakukan untuk memperoleh pemahaman teoritis dan empiris mengenai algoritma pencarian, khususnya Linear Search dan Binary Search, dengan tujuan membandingkan kelebihan, kekurangan, kebutuhan data, serta efisiensi kedua algoritma berdasarkan penelitian yang telah dilakukan sebelumnya[15]. Penelitian ini tidak melibatkan pengujian langsung, namun berfokus pada analisis hasil penelitian terdahulu yang bersumber dari artikel jurnal nasional yang relevan dan kredibel[16]. Tahap awal penelitian dimulai dengan proses pengumpulan dan seleksi literatur[17]. Penulis melakukan pencarian pada basis data jurnal ilmiah nasional dengan kata kunci terkait algoritma pencarian, efisiensi algoritma, dan struktur data. Kriteria inklusi yang ditetapkan adalah jurnal yang memuat implementasi, pengujian, serta perbandingan performa *Linear Search* dan *Binary Search* dalam konteks efisiensi waktu dan kompleksitas komputasi. Selanjutnya dilakukan Analisis Konsep dan Cara Kerja Algoritma, pada tahap ini peneliti mempelajari dan merangkum cara kerja, karakteristik, syarat penggunaan, serta kompleksitas waktu dari kedua algoritma.

**Table 1. Analisis Konsep dan Cara Kerja Algoritma**

Algoritma	Cara Kerja	Kebutuhan Data	Kompleksitas Waktu
Linear Search	Mencari data satu per satu dari awal hingga data ditemukan atau selesai	Tidak harus terurut	O(n)
Binary Search	Membagi data menjadi dua bagian berulang hingga elemen ditemukan	Data <b>harus terurut</b>	O(log n)

Hasil rangkuman ditampilkan dalam bentuk tabel perbandingan konsep dasar, sehingga dapat dipahami bahwa Binary Search memiliki efisiensi lebih baik, namun dengan syarat data wajib dalam keadaan terurut.

### 3. Result and Analysis

Pada tahap ini, hasil dari beberapa jurnal dianalisis untuk mencari kesesuaian dan perbedaan temuan antar penelitian. Beberapa poin penting yang ditemukan: Linear Search lebih fleksibel, karena tidak memerlukan data terurut dan dapat langsung digunakan pada semua jenis dataset, namun waktu pencariannya akan meningkat seiring bertambahnya jumlah data. Binary Search jauh lebih cepat dan efisien, terutama pada data berukuran besar, namun memerlukan praproses berupa pengurutan data terlebih dahulu. Selisih waktu pencarian menjadi sangat signifikan pada dataset besar, dimana Binary Search mampu menyelesaikan proses pencarian berkali-kali lebih cepat dibanding Linear Search. Dengan demikian, dapat disimpulkan bahwa efisiensi algoritma sangat dipengaruhi oleh kondisi dataset dan kebutuhan sistem. Berdasarkan hasil kajian, penulis juga menyusun evaluasi kondisi penggunaan algoritma sebagai berikut:

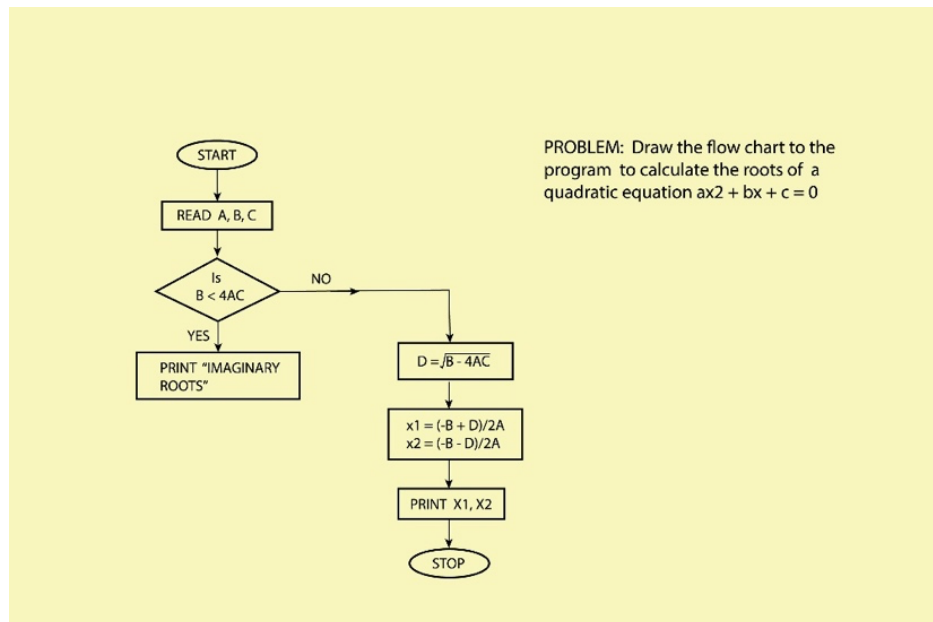
Table 2. Evaluasi Kesesuaian Algoritma

Kondisi Dataset	Algoritma yang Direkomendasikan	Alasan
Data tidak terurut	Linear Search	Tidak membutuhkan proses sorting, langsung bisa digunakan
Data sudah terurut	Binary Search	Lebih cepat dan efisien meskipun memerlukan syarat data terurut
Jumlah data kecil	Linear Search/Binary Search	Perbedaan waktu tidak terlalu signifikan
Jumlah data sangat besar	Binary Search	Performa jauh lebih optimal dan menghemat waktu proses

Pada tahap akhir, penulis menyimpulkan bahwa Binary Search memiliki tingkat efisiensi waktu yang lebih baik, terutama ketika digunakan pada dataset berukuran besar dan sudah dalam kondisi terurut, karena kompleksitas waktunya yang lebih optimal. Di sisi lain, Linear Search dinilai lebih fleksibel, sebab algoritma ini tidak memiliki syarat data harus terurut, sehingga dapat langsung digunakan pada berbagai kondisi dataset, meskipun performanya akan menurun ketika data semakin banyak. Oleh karena itu, pemilihan algoritma pencarian harus disesuaikan dengan kondisi dan karakteristik data yang digunakan, khususnya terkait urutan data serta skala ukuran dataset, agar proses pencarian dapat dilakukan secara efektif dan efisien. Dalam ilmu komputer, algoritma pencarian (*searching algorithm*) didefinisikan sebagai prosedur langkah demi langkah yang sistematis untuk menelusuri dan menemukan lokasi elemen data tertentu (*target key*) di dalam suatu struktur data atau koleksi dataset. Efisiensi sebuah algoritma pencarian menjadi parameter krusial karena berdampak langsung pada kecepatan respon aplikasi, terutama ketika menangani volume data berskala besar. Secara umum, algoritma pencarian dikategorikan berdasarkan metode penelusurannya, di mana *Linear Search* dan *Binary Search* merupakan dua pendekatan yang paling fundamental.

**Algoritma Binary Search**

Berbeda dengan pendekatan sekuensial, *Binary Search* menerapkan paradigma *Divide and Conquer* (memecah dan menaklukkan). Algoritma ini bekerja dengan cara mempartisi rentang pencarian menjadi dua bagian yang sama besar secara berulang-ulang. Proses pencarian selalu dimulai dengan memeriksa elemen yang berada tepat di posisi tengah (*middle*). Jika elemen tengah tersebut adalah nilai yang dicari, proses selesai. Namun, jika nilai yang dicari lebih kecil dari elemen tengah, maka pencarian difokuskan hanya pada separuh bagian kiri array, dan mengabaikan separuh bagian kanan. Sebaliknya, jika nilai yang dicari lebih besar, pencarian dilanjutkan ke separuh bagian kanan. Syarat mutlak agar *Binary Search* dapat beroperasi adalah data harus sudah dalam keadaan terurut (*sorted*) secara *ascending* atau *descending*. Keuntungan signifikan dari metode ini adalah kompleksitas waktunya yang bersifat logaritmik, yaitu  $O(\log n)$ . Efisiensi ini terjadi karena pada setiap langkah perbandingan, algoritma ini mengeliminasi setengah dari kemungkinan lokasi data, sehingga jumlah perbandingan yang dibutuhkan jauh lebih sedikit dibandingkan *Linear Search*, terutama pada data berjumlah ribuan hingga jutaan.



**Gambar 1. Alur Algoritma Linear Search**

*Linear Search*, atau sering disebut sebagai *Sequential Search*, adalah metode pencarian yang paling sederhana dan intuitif. Mekanisme kerja algoritma ini adalah dengan melakukan iterasi (perulangan) pada setiap elemen array satu per satu, dimulai dari indeks pertama ( $index[0]$ ) hingga indeks terakhir ( $index[n-1]$ ). Pada setiap langkah iterasi, nilai data pada posisi saat ini akan dibandingkan dengan nilai yang dicari. Jika nilainya cocok, maka pencarian dihentikan dan indeks dikembalikan; namun jika seluruh elemen telah diperiksa dan data tidak ditemukan, algoritma akan mengembalikan indikator bahwa data tidak tersedia. Keunggulan utama dari *Linear Search* terletak pada fleksibilitasnya, di mana algoritma ini dapat diterapkan pada sekumpulan data yang acak atau tidak terurut (*unsorted*). Namun, kelemahan utamanya adalah kompleksitas waktu yang bersifat linear, yaitu  $O(n)$ . Artinya, dalam skenario terburuk (*worst case*) di mana data berada di posisi paling akhir, waktu eksekusi akan berbanding lurus dengan jumlah total data. Hal ini menjadikannya kurang efisien untuk dataset yang sangat besar.

## Analisis Kompleksitas Waktu

Analisis mendalam terhadap struktur algoritma menunjukkan bahwa perbedaan fundamental antara *Linear Search* dan *Binary Search* terletak pada karakteristik kompleksitas waktunya. *Linear Search* beroperasi dengan kompleksitas waktu  $O(n)$  baik pada kasus rata-rata (*average case*) maupun kasus terburuk (*worst case*). Studi yang dilakukan oleh mengonfirmasi bahwa pola logaritmik ini membuat *Binary Search* sangat stabil, di mana penambahan volume data yang besar hanya berdampak kecil pada penambahan waktu proses. Hal ini mengindikasikan bahwa waktu eksekusi algoritma memiliki korelasi linear yang berbanding lurus dengan jumlah input data. Sebagai implikasinya, apabila volume data ditingkatkan menjadi dua kali lipat, maka estimasi waktu yang dibutuhkan untuk menyelesaikan pencarian juga akan meningkat secara proporsional sebanyak dua kali lipat.

Sebaliknya, *Binary Search* menawarkan efisiensi komputasi yang jauh lebih superior dengan kompleksitas waktu  $O(\log n)$ . Pola logaritmik ini menunjukkan bahwa peningkatan volume data secara masif hanya akan memberikan dampak yang sangat minimal terhadap penambahan waktu pencarian. Efisiensi ini tercapai karena mekanisme *divide and conquer* yang diterapkan, di mana pada setiap langkah iterasi, algoritma secara efektif mengeliminasi setengah dari ruang pencarian yang tersisa, sehingga mempersempit area penelusuran secara eksponensial. Untuk memvisualisasikan disparitas efisiensi secara konkret, penelitian ini menggunakan simulasi skenario pencarian pada *dataset* berskala besar yang berisi 1.000.000 elemen ( $n = 10^6$ ). Dalam skenario terburuk menggunakan *Linear Search* (misalnya data target berada di indeks terakhir atau tidak ditemukan), sistem dipaksa melakukan operasi perbandingan sebanyak 1.000.000 kali. Beban komputasi ini sangat boros sumber daya dan berpotensi menyebabkan *latency* yang tinggi pada sistem *real-time*. Sebaliknya, apabila skenario yang sama diterapkan menggunakan *Binary Search*, jumlah operasi perbandingan maksimum yang diperlukan dapat dihitung dengan rumus  $\log_2(1.000.000) = 19.9$ . Artinya, algoritma ini hanya membutuhkan sekitar 20 langkah perbandingan untuk menemukan data atau menyimpulkan data tidak ada. Perbandingan kontras antara 1.000.000 operasi (*Linear*) melawan 20 operasi (*Binary*) membuktikan adanya kesenjangan efisiensi yang ekstrem, yang menegaskan superioritas *Binary Search* dalam pengelolaan *Big Data*. Kendati *Binary Search* unggul signifikan dari segi kecepatan, implementasinya memiliki "*trade-off*" atau biaya komputasi tambahan, yakni prasyarat data yang harus terurut. Proses pengurutan (*sorting*) itu sendiri memakan waktu, misalnya  $O(n \log n)$  jika menggunakan algoritma *Merge Sort*. Oleh karena itu, efisiensi total sistem bergantung pada karakteristik dinamis data. Jika data bersifat sangat dinamis (sering mengalami operasi penyisipan atau penghapusan data) namun frekuensi pencariannya rendah, *Linear Search* dinilai lebih efektif karena sistem tidak dibebani oleh *overhead* proses pengurutan ulang yang berulang-ulang. Di sisi lain, untuk data yang bersifat statis (jarang berubah) namun memiliki frekuensi akses pencarian yang tinggi, investasi waktu untuk proses *sorting* di awal akan terbayar lunas (*amortized*) oleh kecepatan pencarian *Binary Search* yang berulang kali dilakukan. Sebagaimana disimpulkan oleh, *Linear Search* tetap relevan digunakan untuk data dinamis dengan volume kecil hingga menengah, sedangkan *Binary Search* lebih optimal untuk data statis bervolume besar yang membutuhkan akses temu kembali (*retrieval*) yang cepat.

## 4. Conclusion

Berdasarkan hasil analisis komparatif dan pembahasan yang telah dipaparkan, penelitian ini menyimpulkan bahwa efisiensi algoritma pencarian sangat bergantung pada karakteristik data dan kebutuhan sistem. Algoritma *Linear Search* terbukti unggul dalam aspek fleksibilitas karena tidak memerlukan prasyarat data terurut, menjadikannya solusi ideal untuk *dataset* bervolume kecil atau data

yang bersifat sangat dinamis (sering mengalami perubahan). Namun, kompleksitas waktu linear  $O(n)$  yang dimilikinya menjadi kendala performa yang serius ketika diterapkan pada data berskala besar. Di sisi lain, *Binary Search* menunjukkan superioritas kinerja yang signifikan pada pengolahan *Big Data*. Dengan kompleksitas waktu logaritmik  $O(\log n)$ , algoritma ini mampu memangkas waktu pencarian secara drastis dibandingkan metode sekuensial. Meskipun demikian, efisiensi ini memerlukan *trade-off* berupa biaya komputasi awal untuk proses pengurutan (*sorting*). Oleh karena itu, pemilihan algoritma yang tepat tidak bersifat mutlak, melainkan harus disesuaikan dengan konteks implementasi: gunakan *Binary Search* untuk sistem dengan data statis bervolume besar yang membutuhkan kecepatan akses tinggi, dan gunakan *Linear Search* untuk data dinamis bervolume kecil hingga menengah demi meminimalkan kompleksitas pemeliharaan data.

### Daftar Pustaka

- [1] Fauzi, A., & Irsyad, M. (2021). Implementasi Algoritma Linear Search dan Binary Search pada Aplikasi Kearsipan. *Jurnal Teknologi Informasi*, 5(2), 120-129.
- [2] Nasution, H. (2022). Analisis Perbandingan Kompleksitas Algoritma Pengurutan dan Pencarian pada Struktur Data Array. *Jurnal Sistem Informasi dan Ilmu Komputer*, 6(1), 45-53.
- [3] Rahmayani, R., Firmansyah, E., & Hikmah, H. U. (2025). Inovasi Layanan Antar Jemput Paket Surat PTPos Indonesia Berdasarkan Penjualan dan Minat Beli JOVISHE: *Journal of Visionary Sharia Economy*, 4(1), 33-47.
- [4] Firmansyah, E., Rahman, A. B. A., & Subiyakto, A. A. (2023). Pengukuran Kesiapan Kota Cerdas Berdasarkan SNI ISO 37122: 2019. *Infoman's: Jurnal Ilmu-ilmu Informatika dan Manajemen*, 17(2).
- [5] Zulfikar, W. B., Irfan, M., Ghufroon, M., Jumadi, J., & Firmansyah, E. (2020). Marketplace affiliates potential analysis using cosine similarity and vision-based page segmentation. *Bulletin of Electrical Engineering and Informatics*, 9(6), 2492-2498.
- [6] Ramadhan, N. D., Fhatturohmah, S., Ramadhani, S., & Firmansyah, E. (2023). Analysis of Digital Wallet Usage on Consumptive Lifestyle. *Journal of Islamic Economics and Business*, 3(2), 118-136.
- [7] Sutara, B., & Firmansyah, E. (2021). Design and Build Student Attendance System Using Fingerprint. *J-Tin's- Jurnal Teknik Informatika*, 5(1).
- [8] Wahana, A., Firmansyah, E., Al Rosyid, H. I., Fuadi, R. S., & Maylawati, D. S. A. (2021). Fuzzy Tahani Method in the Recommendation System for Selecting Mountain Tourism Destinations in West Java.
- [9] Tamrin, M. A., Rizki, B., Nodas, A., Rahman, A., & Firmansyah, E. (2020). Perbandingan Penggunaan Metode Topsis dan Metode AHP dalam Penilaian Kinerja pada Karyawan (PT XYZ). *Infoman's: Jurnal Ilmu-ilmu Informatika dan Manajemen*, 14(1).
- [10] Firmansyah, E., Herdiana, D., & Yuniarto, D. (2020, October). Examining readiness of e-Learning implementation using information system readiness impact model. In *2020 8th International Conference on Cyber and IT Service Management (CITSM)* (pp. 1-5). IEEE.
- [11] Firmansyah, E., Rosmawati, R., Fuadi, R. S., Fauzy, D., & Ramdhani, M. A. (2019, December). Design of expert system to determine the proper diet using harmony search method. In *Journal of Physics: Conference Series* (Vol. 1402, No. 7, p. 077006). IOP Publishing.
- [12] Firmansyah, E., & Helmiawan, M. A. (2025). Pengukuran Kesiapan Transformasi Digital Desa Kaduwulung Menuju Desa Cerdas Berbasis SNI ISO 37122: 2019 Melalui Pemetaan Data Desa. *Infoman's: Jurnal Ilmu-ilmu Informatika dan Manajemen*, 19(1).
- [13] Firmansyah, E., Herdiana, D., Yuniarto, D., & Junaedi, D. I. (2021, September). The K-Nearest Neighbor Algorithm for the Classification of Internet Users in Rural Campus. In *2021 9th International Conference on Cyber and IT Service Management (CITSM)* (pp. 1-6). IEEE.
- [14] Syaripudin, U., Zaenal, R., Duri, M. F. A., Firmansyah, E., & Rahman, A. (2019, December). Comparison between Naïve Bayes and certainty factor to predict big five personality. In *Journal of Physics: Conference Series* (Vol. 1402, No. 7, p. 077030). IOP Publishing.
- [15] Pratama, R., & Suhendar, A. (2020). Komparasi Efisiensi Algoritma Linear Search dan Binary Search pada Database Penjualan. *Jurnal Informatika Universitas Pamulang*, 5(2), 1-7.

- [16] Rohman, F., & Fauziah, S. (2018). Penerapan Algoritma Sequential Search pada Aplikasi Kamus Istilah Medis Berbasis Android. *Jurnal Informatika Merdeka Pasuruan*, 3(2), 15-21.
- [17] Siregar, R. R. (2019). Analisis Efisiensi Algoritma Searching pada Pencarian Data Mahasiswa. *Jurnal Teknik Informatika Kaputama (JTIK)*, 3(2), 56-61.