Pengurutan Stok Barang Toko Andis's Collection Menggunakan Algoritma Divide and Conquer

Indyah Hartami Santi[#*1], Putri Merly Deleo Karina[#2], Asrul Hidayatul Kusna[#3], Salsabilla Anida Putri Priyanto[#4]
[#1234]Fakultas Teknologi Informasi, Universitas Islam Balitar, Blitar, Indonesia
Korespodensi author: indyahartamisanti@gmail.com

Info Artikel

Diajukan: 27 April 2024
Diterima: 7 Juni 2024
Diterbitkan: 7 Juni 2024

Keywords:
Pengurutan, Stok Barang, Algoritma Divide and Conquer, Kompleksitas

Kata Kunci:
Sorting, Stock, Divide and Conquer Algorithm, Complexity

Abstract

This research was conducted at Andy's Collection Store, with the aim of knowing the sorting system based on stock of goods and price of goods using the Divide and Conquer algorithm, as well as calculating the algorithm complexity and time complexity of the Divide and Conquer algorithm. The Divide and Conquer algorithm is a variant of several top down programming strategies by creating sub-subs of a large problem. The result of the research is to fly a sorting application based on stock and price of goods using the Divide and Conquer algorithm. From this application, testing was carried out on the complexity of the algorithm to produce the worst case and the best case. Apart from that, it produces time complexity where the testing time data is assessed, there is a significant increase in data processing from 500 data to 1000 data.

Abstrak

Penelitian ini dilakukan di Toko Andy's Collection, dengan tujuan untuk mengetahui sistem pengurutan berdasarkan stok barang dan harga barang dengan menggunakan algoritma divide and conquer, serta menghitung kompleksitas algoritma dan kompleksitas waktu algoritma divide and conquer. Algoritma Divide and conquer merupakan varian dari beberapa strategi pemrograman top down dengan membuat sub-sub dari problem yang besar.Hasil penelitian adalah terbangnnya aplikasi pengurutan berdasarkan stok dan harga barang dengan menggunakan algortma Divide and conquer. Dari aplikasi tersebut dilakukan pengujian terhadap kompleksitas algoritma menghasilkan kasus worst case dan best case. Selain itu menghasilkan kompleksitas waktu dimana dilai data waktu pengujian terjadi kenaikan signifikan pada proses data dari 500 data ke 1000 data.

Cara mensitasi artikel:
I.H. Santi, P.M.D. Karina, A.H. Kusna, S.A.P. Priyanto. "Pengurutan Stok Barang Toko Andy's Collection Menggunakan Algoritma Divide and Conquer." *Jurnal Teknologi Informasi: Teori, Konsep, dan Implementasi (JTI-TKI), vol. 15, no. 1, pp. 35-40, Maret 2024, https://doi.org/10.36382/jti-tki.v15i1.522*

PENDAHULUAN

Penggunaan teknologi seperti jaringan area lokal, jaringan area nirkabel, jaringan global, intranet, internet, dan ekstranet menjadi semakin umum di masyarakat. Hal ini mendukung penyebaran informasi yang lebih cepat. Sistem penyimpanan barang mempunyai pengertian barang disimpan oleh suatu perusahaan/dealer untuk dijual kembali di kemudian hari.

Sistem pengelolaan barang yang disimpan di gudang umumnya dilakukan dilakukan secara manual, namun istem pengelolaan barang juga bisa dilakukan secara otomatis. Otomatis yang dimaksud adalah pengelolaan dengan menggunakan aplikasi stok barang. Perusahaan dan pedagang yang menggunakan aplikasi biasanya tidak mengetahui cara membuat aplikasi, namun hanya bisa menggunakan aplikasi saja. Hasil pengumpulan data di Toko Andi's Collection

Manajemen gudang yang efisien penting dan dianggap menguntungkan bagi perusahaan dagang. Seringkali tidak mungkin memperkirakan status persediaan di toko yang sibuk tanpa mengetahui cara memposisikan produk dengan benar di gudang. Kekurangan persediaan dapat menimbulkan kekecewaan pelanggan toko dan menurunkan loyalitas pelanggan, terutama jika pesanan sudah diterima. Meskipun kekurangan stok menyebabkan kerugian, namun kelebihan stok tidak selalu berarti menguntungkan. Faktanya, banyak faktor yang berdampak negatif terhadap kelebihan stok, termasuk terbuangnya ruang gudang yang tidak cocok untuk menyimpan barang lain yang terjual lebih baik atau lebih menguntungkan. Stagnasi modal juga disebabkan oleh kelebihan persediaan

Ada beberapa cara untuk meningkatkan kualitas sistem penyimpanan, salah satunya adalah dengan menggunakan teknologi modern. Ketika informasi atau aplikasi berkualitas tinggi, informasi yang digunakan untuk mendukung aktivitas perdagangan seseorang atau perusahaan memiliki kontrol sistem maksimum yang memungkinkan kinerja dipantau, dan informasi yang dihasilkan dan ditampilkan adalah jika mengantisipasi kemungkinan kesalahan dalam suatu peristiwa atau kesalahan fatal yang akan terjadi di kemudian hari. Di sisi lain sumber daya manusia harus lebih kreatif dan mampu memanfaatkan perkembangan teknologi untuk memenuhi kebutuhan yang semakin kompleks.

Penelitian ini merupakan penelitian lanjutan yang sudah dilakukan sebelumnya [1], dimana penelitian terdahulu telah berhasil dibangun sebuah sistem informasi pengolahan data transaksi jual beli. Dan selanjutnya untuk mengurangi permasalahan yang terjadi pada sistem dalam mengelola stok barang, maka diperlukan pengelolaan data stok barang menjadi lebih efektif dan menghasilkan laporan stok yang akurat. Penelitian ini akan mengimplementasikan sistem pengurutas pada stok barang dengan menggunakan algoritma *Divide and Conquer*.

Cara kerja algoritma *Divide and Conquer* didasarkan pada prinsip membagi suatu permasalahan yang besar menjadi beberapa bagian yang lebih kecil. Tujuan pembagi menjadi bagian yang lebih kecil ini adalah agar lebih mudah penyelesaiannya. Algoritma membagi masalah menjadi dasar dari algoritma ini dengan mencari nilai minimum dan maksimum yang dimasukkan oleh pengguna.

Sistem *sorting stock* barang ini juga dirancang bertujuan untuk dapat mengurangi kesalahan yang terjadi, terutama dalam data barang masuk, barang keluar dan dapat membantu perusahaan/pedagang dalam melakukan pencarian barang apa saja yang habis dan barang apa saja yang stok masih melimpah. Sehingga dalam pemantauan stok barang menjadi optimal dan meminimalkan terjadinya kesalahan

Beberapa penelitian terdahulu yang tertarik dengan pembahasan pengurutan dilakukan oleh [2] dan [3]. sedangkan penelitian lain dengan menggunakan metode *divide and conquer* dilakukan dengan obyek aplikasi *evoting*[4], translasi bahasa inggris–bahasa indonesia [5], pemanfaatan GPS untuk mengetahui lokasi tindak kejahatan[6], simulasi *voice recognition*[7], pembelajaran[8], aplikasi belajar ilmu tajwid[9], pengaturan ruang *meeting*[10]. Sedangkan penelitian lain dilakukan pada pembuatan aplikasi surat kontrak kerja[11], permainan final fantasy III[12], proses mengetahui ketaatan remaja pada protokol kesehatan[13]. selain itu terdapat penelitian yang mengomparasikan algoritma *divide and conquer* dengan algorima *brute force* seperti yang dilakukan oleh [14] dan [15]

METODE

A. Jenis Penelitian

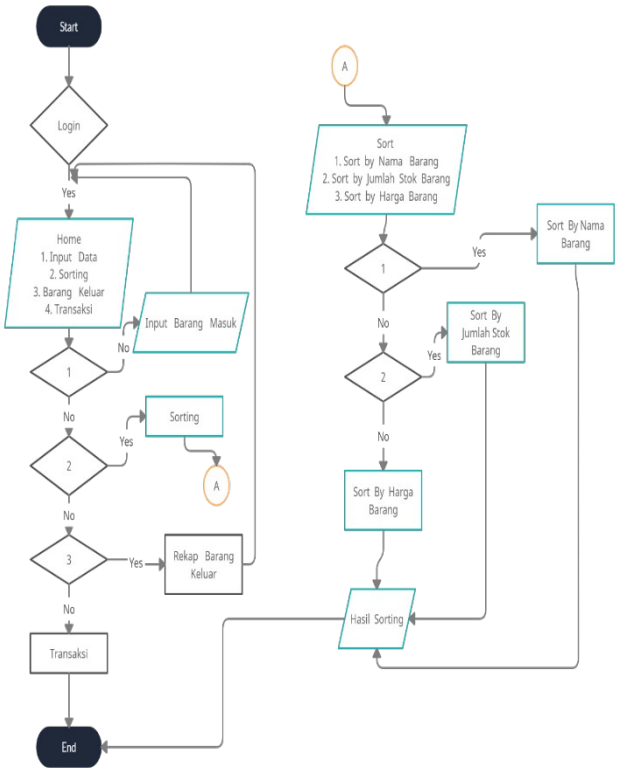
Jenis penelitian ini adalah deskriptif dengan tahapan meliputi pengumpulan data, mengolah data dan menganalisis data.

B. Prosedur

Prosedur yang dilakukan dalam penelitian ini adalah mengikuti tahapan algoritma *divide and conquer* yaitu :

1. Tentukan konstanta pembagian data
2. Proses pembagian data menjadi 2 bagian, kiri dan kanan jika jumlah data lebih dari 2. Apabila jumlah data sama dengan 2 maka dilakukan proses pengurutan data
3. Data masing-masing bagian baik kanan dan kiri, dilakukan proses pengurutan
4. Selanjutnya hasil pengurutan bagian kanan dan kiri, dilakukan pengurutan data secara gabungan.

Berikut alur program implementasi algoritma *divide and conquer* tersaji dapat digambarkan pada gambar 1 berikut:



Gambar 1. Alur Algoritma Divide and Conquer

HASIL DAN PEMBAHASAN

A. Pengumpulan Data

Tahap awal pengupulan data dilakukan dengan cara survey ke Toko Andy's Collection. Dari hasil survei dilanjutkan dengan wawancara kepada penilik toko. Dari kegiatan survei dan wawancara ini diperoleh data terkait sistem yang ada pada toko dan data barang yang ada di toko. Pada penelitian ini mengambil sampel data khusus pada 8 jenis barang jilbab. Data ke delapan jenis jilbab tersebut harga dan jumlah stok dapat disajikan seperti pada tabel 1 berikut:

Tabel 1. Data barang, harga dan stok

NO	NAMA	HARGA	STOK
1	Pashmina Ceruty	22000	25
2	Bella Square	14000	36
3	Paris Premium	12000	14
4	Hijab Sport	10000	47

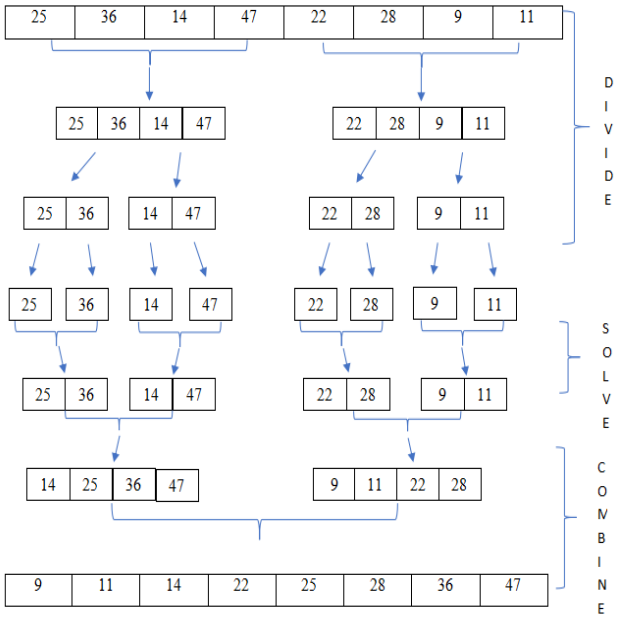
5	Pasmina Plisket	25000	22
6	Paris Voal Premium	18000	28
7	Bergo	17000	9
8	Pasmina Plisket Non full	23000	11

B. Pengolahan Data

Atas dasar hasil pengumpulan data pada tabel 1 maka selanjutnya dilakukan proses pengolahan data. Pengolahan data ini dimulai dengan melakukan proses *merge sort* terhadap stok barang maupun harga barang, kemudian dilakukan perancangan *pseuducode* algoritma, dan selanjutnya pembuatan tampilan sistem aplikasinya.

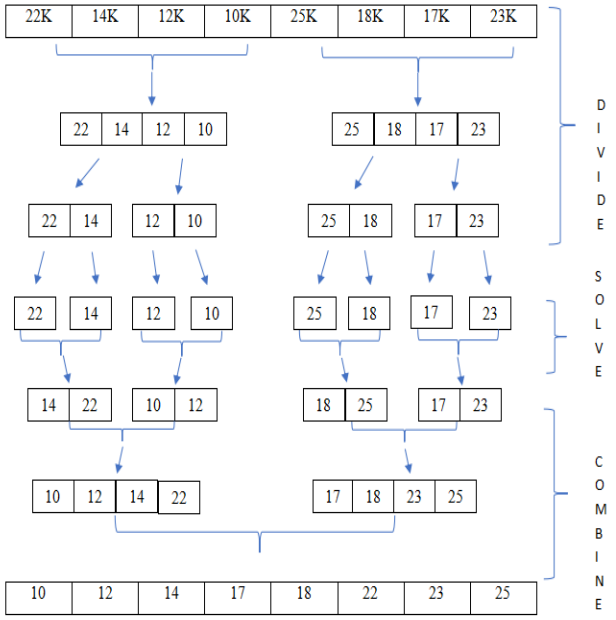
1) *Merge sort*

Gambaran proses *merge sort* stok barang pada penelitian ini dapat dilihat pada gambar 2 berikut:



Gambar 2. Proses merge sort terhadap stok barang

Sedangkan gambaran proses *merge sort* terhadap harga disajikan pada gambar 3 berikut:



Gambar 3. Proses merge sort terhadap harga barang

2) *Pseudocode* algoritma

Pseudocode dapat diartikan dengan pengembangan algoritma yang menggunakan kode tertentu untuk menjelaskan cara kerja sesuatu atau cara menyelesaikan suatu masalah. Sederhananya, *pseudocode* adalah suatu algoritma yang telah diubah menjadi kode tertentu.

Pseudocode merge sort:

```
mergersort(data)
if data memiliki dua elemen
        mergesort (setenga kiri dari data);
        mergesort (setengah kanan dari data
merge (bagian kiri dan kanan menjadi satu urutan)
```

Pseudocode merge sort produk :

```
Program Sorting Produk
{mencetak urutan Produk menggunakan algoritma divide and
conquer dengan Teknik merge sort}
DEKLARASI
id[] , tengah , i1,i2,i3: integer

ALGORITMA
Id[]← read database
id{
merge (id[] , id[0] , id[n-1])
tengah = (id[0] + id[n-1]) / 2;
il = 0;
i2 = id[0];
i3 = tengah + 1;
while kedua sub larik dari id[] memiliki elemen
   if id[i2] < id[i3]
     temp[il++] = id[i2++];
   else
     temp[il++] = id[i3++];
masukkan ke dalam temp sisa elemen dari id[];
   masukkan ke id[] isi dari temp;
}
```

3) Tampilan aplikasi

a) login

Tampilan aplikasi login pada sistem pengelolaan ini digunakan untuk memulai aplikasi. Tampilan login ini membatasi siapa saja yang dapat mengakses aplikasi ini atau melihat data yang terdapat dalam sistem. Tampilan halaman login tersajikan seperti pada gambar 4 berikut:

Gambar 4. Tampilan Login

b) Tampilan Menu

Setelah administrator berhasil login, admin akan diarahkan ke halaman HOME. Terdapat tombol di bagian bawah halaman BERANDA untuk membantu *administrator* mengumpulkan data. Button ini terdiri dari *form input* data, *sorting item*, barang keluar dan transaksi. Tampilan halaman menu utama aplikasi ini dilihat pada gambar 5 berikut:

Gambar 5. Tampilan halaman menu

c) Tampilan Sorting Item

Gambar 6. Tampilan Sorting Item

Pada tampilan Sorting Item terdapat 2 pilihan yaitu admin akan menyorting hijab atau gamis. Selain itu juga terdapat 3 pilihan dimana akan melakukan penyortiran by name, b y stok quantity, atau by price. Selanjutnya juga terdapat 2 button dimana bagian back digunakan untuk kembali ke m enu home dan bagian proses digunakan untuk memproses penyortiran. Tampilan Sorting Item dapat dilihat pada ga mbar 6.

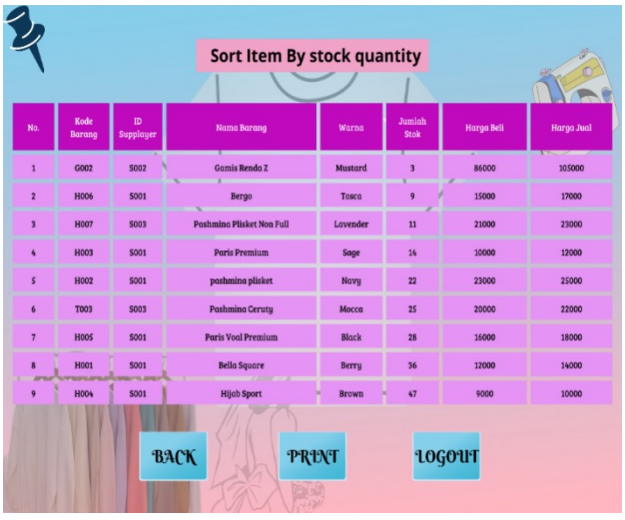
d) Tampilan Sorting by Name

Tampilan *sorting by name* terdapat list hasil penyortiran dan juga terdapat beberapa *button* diantaranya *button back* , *print* dan *logout*. Tampilan *sorting by name* dapat dilihat pada gambar 7 berikut:

Gambar 7. Tamipilan halaman sorting by name

e) Tampilan Sorting by Stok Quantity

Tampilan *sorting by stock quantity* terdapat *list* hasil penyortiran dan juga terdapat beberapa *button* diantaranya *button back, print*, dan *logout.* Tampilan *sorting by stock quantity* dapat dilihat pada gambar 8 berikut:

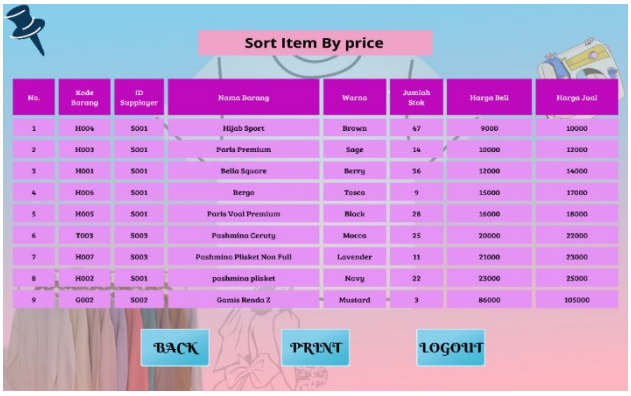


Gambar 8. Tampilan Sort by stok quantity

f) Tampilan Sorting by price

Tampilan *sorting by price* terdapat list hasil penyortiran dan juga terdapat beberapa button diantaranya

button back, print, dan *logout.* Tampilan *sorting by price* dapat dilihat pada gambar 9 berikut:



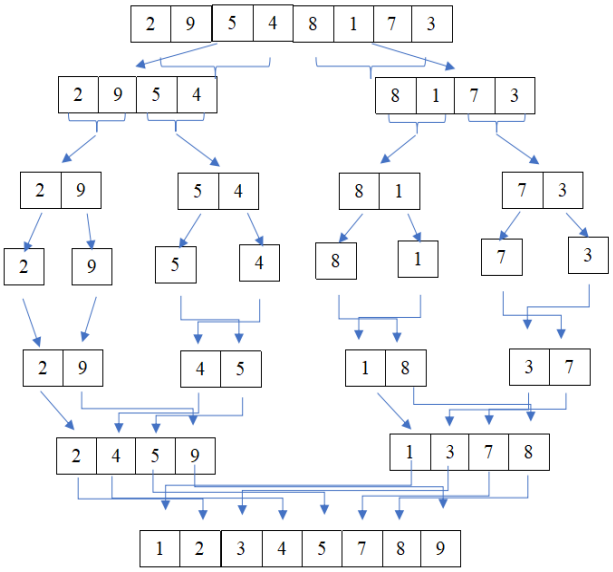
Gambar 9. Tampilan Sort by price

C. *Analisa Data*

Tahap analisa data dilakukan dengan cara pengujian terhadap algoritma yang digunakan yaitu algoritma *divide and conquer*. Pengujian ini dilakukan dengan menghitung kompleksitas algoritmanya dan kompleksitas waktu.

1) Kompleksitas Algoritma

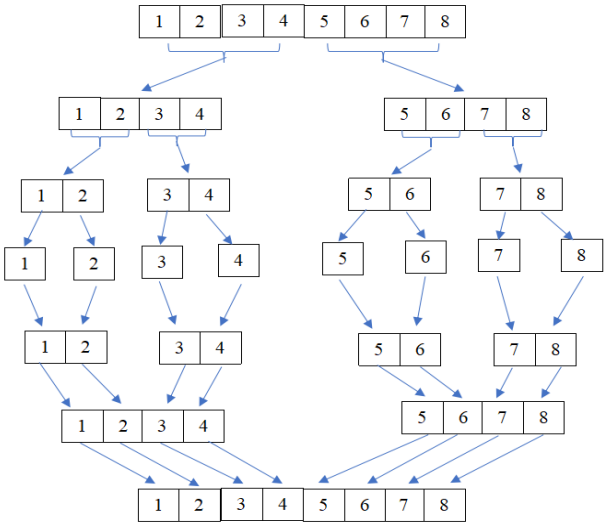
Ukuran jumlah komputasi yang diperlukan suatu algoritma untuk menyelesaikan masalah disebut sebagai kompleksitas algoritma. Sebuah algoritma yang dapat menyelesaikan suatu masalah dengan waktu yang cepat mempunyai kompleksitas yang rendah, dan sebaliknya algoritma yang memerlukan waktu penyelesaian yang lama mempunyai kompleksitas yang tinggi. Dua jenis kompleksitas algoritma yang diketahui yaitu kompleksitas waktu dan kompleksitas ruang. Kompleksitas waktu atau ruang memungkinkan menentukan laju peningkatan waktu atau ruang yang dibutuhkan oleh suatu algoritma seiring dengan meningkatnya ukuran *input* n.



Gambar 10. Kasus Worst Case

Kasus terburuk terjadi ketika setiap pemanggilan fungsi penggabungan rekursif memiliki nilai maksimum yang berbeda untuk setiap elemen dalam array. Ini akan menyebabkan fungsi penggabungan mengganti array dan melakukan pengurutan. Oleh karena itu, kompleksitas kasus terburuk adalah O (n log n). Di bawah ini adalah contoh kasus terburuk dari algoritma pembagian dan penaklukan. Pengurutan gabungan dilakukan berdasarkan inventaris, seperti yang ditunjukkan pada Gambar 10.

Kasus terbaik untuk metode ini merupakan suatu kondisi dimana suatu elemen mempunyai nilai maksimal yang lebih kecil dari seluruh nilai elemen lainnya, dengan menggunakan proses perhitungan yang sama seperti kasus terburuk. Berikut contoh kasus *best case* pada algoritma *divide*, dimana dilakukan *merge sort* berdasarkan stok barang seperti pada gambar 11 berikut:



Gambar 11. Kasus Best Case

2) Kompleksitas Waktu

Kompleksitas waktu dari algoritma memerlukan ukuran input n dan waktu eksekusi. Waktu eksekusi suatu algoritma dapat dinyatakan sebagai fungsi dari n, karena waktu eksekusi itu sendiri bertambah seiring dengan bertambahnya ukuran n. Ukuran *input* algoritma n tergantung pada masalah yang diselesaikan oleh algoritma. Waktu eksekusi suatu algoritma untuk *input* tertentu n adalah jumlah operasi atau langkah yang dilakukan. Setiap baris kode semu memerlukan waktu tertentu untuk dieksekusi. Setelah bentuk fungsi ditentukan, notasi asimtotik O dapat digunakan untuk menentukan bentuk algoritma. Dengan menentukan bentuk algoritma, dapat memperkirakan berapa lama waktu eksekusi akan bertambah seiring bertambahnya ukuran masukan. Kompleksitas waktu proses RECRUSIVE T(n) = waktu eksekusi kasus terburuk untuk mengurutkan n data, dimisalkan:

$n = 2^k$,

K = semua integer.

```
If (kiri < kanan) {
  Int tengah = (kiri + kanan) / 2;
  Mergesort(A,kiri,tengah;          T(n/2)
  Mergesort(A,center+1,kanan);      T(n/2)
  Merge(A,kiri,kanan+1,kanan);      O(n/2 + n/2 = n)
}
```

Dari algoritma diatas dapat dijelaskan bahwa:
1) Terdapat 2 rekrusif *merge sort*
2) Kompleksitas waktu T(n/2)
3) Proses *merge sort* memerlukan waktu O(n). ini dilakukan proses penggabungan hasil dari *merge sort rekrusif*

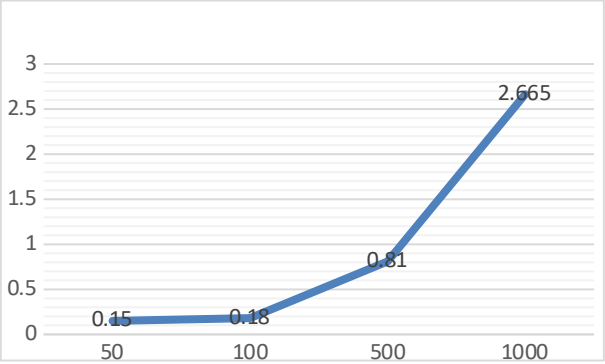
Berdasar pada penjelasan diatas maka dilakukan proses perhitungan kompleksitas waktu seperti berikut:
1) Untuk n>1 -----> T(n) = 2T(n/2) + O(n)
2) Untuk n=1 -----> T(1) = O(1) n=1

Hasil waktu eksekusi dari *algoritma divide and conquer* dengan n=data dapat dilihat pada table 2 berikut:

Table 2. Hasil waktu eksekusi

DATA	RUNNING TIME (ms)
50	0.15
100	0.18
500	0.81
1000	2.665

Hasil waktu eksekusi dari algoritma *divide and conquer* dengan n=data dapat digambarkan seperti pada gambar 12 berikut:



Gambar 12. Grafik hasil eksekusi kompleksitas waktu

KESIMPULAN DAN SARAN

Berdasarkan hasil pengolahan data diperoleh hasil sebuah sistem proses pengurutan barang berdasarkan pada harga dan stok barang dengan menggunakan *algoritma divide and conquer*. Sedangkan dari hasil analisa data diperoleh hasil 2 gambaran kompleksitas algoritma yaitu kasus *wosth case* dan *best case*. Disamping itu juga dihasilkan nilai kompleksitas waktu dimana data dari 500 ke 1000 data mengalami peningkatan kompleksitas waktu yang signifikan.

REFERENSI

[1] IH Santi, PMD Karina, Rancang Bangun Sistem Informasi Pengolahan Data Transaksi Jual Beli Di Toko Andis's Collection, Jurnal Penelitian Multidisiplin Ilmu 1 (2), 193-214, 2022.
[2] Written By Fidelson Tanzil, Tanzil, F., & Subject Content Coordinator - Basic Programming | School of Computer Science. (n.d.). *Selection sort*. School of Computer Science. Retrieved October 31, 2022
[3] Zaliluddin, D., & Rohaeti, E. 303 Analisis dan Perancangan Aplikasi Pegelolaan Data Kaegori Kesejahteraan Penduduk di Desa Cibunut Menggunakan Algoritma Divide and Conquer. In Prosiding Industrial Research Workshop and National Seminar (Vol. 9, pp. 303-312).(2018, October).
[4] Adnan, A. (2013, June). Metode Divide and Conquer Parallel dan Parallel-Reduce Pada Cilk for Untuk Aplikasi E-Voting Berbasis Sistem Prosesor Multicore. In Seminar Nasional Aplikasi Teknologi Informasi 2013.
[5] Farabi, I., Lubis, H. B., & Lianda, R. Penggunaan Algoritma Divide and Conquer untuk Pembedaan Konteks Kata Kerja dan Kata Benda dalam Translasi Bahasa Inggris–Bahasa Indonesia.
[6] Faozi, A. Pemanfaatan GPS Untuk Mengetahui Lokasi Tindak Kejahatan Dengan Algoritma Divide And Conquerer. Jurnal Teknik Informatika dan Sistem Informasi, 1(2), 11-15.(2021).
[7] Halim, C. A. (2010). Perancangan Program Simulasi Voice Recoqnition Untuk Identifikasi Menggunakan Algoritma FFT dan Divide and Conquer. Tersedia: http://library. binus. ac. id/eColls/eThesisdoc/Bab, 2(2010), 2.
[8] Hardan, J. Perancangan dan Implementasi Perangkat Lunak Pembelajaran Algoritma Divede and Conquer Berbasis Multimedia (Studi Kasus: Mata Kuliah Teknik Penyelesaian Persoalan) (Doctoral dissertation, Universitas Islam Negeri Sultan Syarif Kasim Riau).(2010)
[9] Suryani, D., Irfan, M., Uriawan, W., & Zulfikar, W. B. . Implementasi Algoritma Divide and Conquer pada Aplikasi Belajar Ilmu Tajwid. Jurnal Online Informatika, 1(1), 13-19. (2016)
[10] Wahyu, S., & Kurniawan, R. Rancang Bangun Pengaturan Ruang Meeting Dengan Algoritma Divide and Conquer Pada Hotel Amalia Bandar Lampung, Jurnal Informatika, 19(1), 79-84.(2019).
[11] S Harjono, Pengembangan Aplikasi Pembuat Surat Kontrak Kerja, Peringatan dan Perjalanan Dinas dengan Algoritma Divide And Conquer, INFOMATEK: Jurnal Informatika, Manajemen dan …, 2023 - journal.unpas.ac.id
[12] JM Adinulhaq, S Safuan, Penerapan Algoritma Divide and Conquer untuk Berburu Monster dalam Permainan Final Fantasy III, JURNAL KOMPUTER DAN …, 2023 - jurnal.unimus.ac.id
[13] V Rahmansyah Putra, Penerapan Algoritma Divide and Conquer Untuk mengetahui ketaatan remaja dalam mematuhi protokol kesehatan disaat wabah, 2021 - eprints.umpo.ac.id
[14] FAT Tobing, JR Tambunan, Analisis Perbandingan efisiensi algoritma Brute Force dan Divide and Conquer dalam proses pengurutan angka, Ultimatics: Jurnal Teknik …, 2020 - ejournals.umn.ac.id
[15] A Gunadi, Analisis Komparasi Algoritma Sorting Antara Metode Brute Force dengan Divide and Conquer, JURNAL ILMU KOMPUTER INDONESIA, 2020 - ejournal-pasca.undiksha.ac.id