

Analisis Pengujian Data Algoritma Bubble Sort

Yuniar Andi Astuti
Politeknik Unggul LP3M
Kota Medan, Indonesia

andiyuniar45@gmail.com

*Penulis Korespondensi

Diajukan : 23/06/2023
Diterima : 02/08/2023
Dipublikasi : 02/08/2023

ABSTRAK

Ada banyak algoritma untuk mengurutkan data seperti *bucket sort*, *bubble sort*, *insertion sort*, *selection sort*, *heapsort*, *merge sort* dan lain-lain. Dimana masing-masing algoritma memiliki kelebihan dan kekurangan. Pengurutan data adalah salah satu operasi pada struktur data yang digunakan dalam situasi khusus. Pengurutan didefinisikan sebagai pengaturan data atau catatan dalam urutan logis tertentu. Sejumlah algoritma dikembangkan untuk mengurutkan data. Algoritma Bubble Sort sebagai materi yang menarik dalam pembelajaran ilmu algoritma dan pemrograman yang seringkali materi di dalamnya ini agak sulit dipahami, oleh sebab itu tujuan dari penelitian ini agar materi *Bubble Sort* bisa dipahami sehingga proses pembelajaran bisa berjalan dengan baik. Metode yang dilakukan dalam penelitian adalah studi literatur dan pengujian data yang berupa angka secara acak dengan algoritma *Bubble Sort*. Hasil pengujian data dengan algoritma *bubble sort* dan penerapan program dengan *Dev C++* kemudian pengujian data dilakukan pada deret angka secara acak yaitu dengan membandingkan data di sebelahnya, jika data tidak diurutkan maka akan berpindah tempat sebaliknya jika data sudah terurut maka posisi data tidak berubah. Algoritma pengurutan data *bubble sort* merupakan algoritma yang paling sederhana dan mudah dipahami dibandingkan dengan algoritma pengurutan lainnya. Diharapkan penelitian ini mempermudah dan menjadi sumber belajar yang bermanfaat untuk mahasiswa atau generasi selanjutnya terutama masyarakat Indonesia.

Kata Kunci: Algoritma, *bubble sort*, mudah, sederhana

I. PENDAHULUAN

Ada banyak algoritma untuk mengurutkan data seperti *bucket sort*, *bubble sort*, *insertion sort*, *selection sort*, *heapsort*, *merge sort* dan lain-lain. Dimana masing-masing algoritma memiliki kelebihan dan kekurangan. (Alkareem Alyasseri dkk., 2015; Nasser dkk., 2014; Sameer dkk., 2016) Pengurutan data adalah salah satu operasi pada struktur data yang digunakan dalam situasi khusus. Pengurutan didefinisikan sebagai pengaturan data atau catatan dalam urutan logis tertentu. Sejumlah algoritma dikembangkan untuk mengurutkan data. (Shabaz & Kumar, 2019)

Algoritma pengurutan memainkan peran penting dalam bidang ilmu komputer. Banyak aplikasi menggunakan algoritma pengurutan (Purnomo & Putra, 2023). Pengurutan data sangat penting dalam konsep ilmu komputer yang berperan dalam manajemen memori, manajemen file, dan aplikasi yang nyata lainnya (Vilchez, 2020). Algoritma pengurutan yang berbeda memiliki perbedaan dalam hal kompleksitas waktu, penggunaan memori, efisiensi, dan faktor lainnya (Zutshi & Goswami, 2021).

Pada penelitian yang dilakukan oleh Rihartanto dkk dengan judul “*Performance of Parallel Computing in Bubble Sort Algorithm*”, algoritma *bubble sort* dengan menggunakan

pemrosesan paralel memiliki kinerja yang baik dalam proses pengurutan terutama pada data yang besar dibandingkan proses serial (Rihartanto. et.Al, 2017)

Penelitian dilakukan oleh (Rizka Poetra, D., Hayati, N. 2022). Dari segi performa waktu eksekusi algoritma quick sort jauh lebih cepat dibandingkan dengan algoritma bubble sort, namun dari segi performa akurasi keduanya menghasilkan akurasi yang sama baiknya. (Rizka Poetra & Hayati, 2022)

Algoritma merupakan urutan langkah-langkah penyelesaian suatu masalah, langkah-langkah tersebut berupa kumpulan perintah atau instruksi yang dibuat secara jelas dalam penyelesaian suatu masalah. Kumpulan perintah atau instruksi yang dimaksud dibangun dari suatu bahasa pemrograman yang dapat dimengerti oleh komputer. Kumpulan perintah atau instruksi tersebut berupa langkah-langkah atau tahapan-tahapan yang diberikan untuk menyelesaikan suatu pekerjaan atau suatu masalah secara cepat, lebih mudah, dan efisien. Algoritma dapat kita asumsikan kurang lebih sama dengan suatu prosedur atau kegiatan yang sering kita lakukan dalam kehidupan kita sehari-hari. Misalnya dalam membuat kue, algoritmanya adalah resep kue atau cara atau urutan dari pembuatan kue, langkah-langkahnya sebagai berikut: siapkan bahan-bahan yang digunakan misalnya tepung, telur, gula, mentega dan lain-lain, pada tahap awal campurkan telur, tepung dan kocok sampai mengental, tambahkan mentega dan tepung kemudian dibentuk lalu dipanggang dan akhirnya kue siap untuk di makan.

Salah satu pemanfaatan algoritma dalam dunia pemrograman yaitu algoritma pengurutan atau sorting algorithm, yang sering digunakan dalam proses pengurutan suatu data. Pengurutan data atau yang lebih dikenal istilah sorting merupakan salah satu hal yang amat sangat penting bagi pengelolaan data. Karena dengan pengurutan, proses pengelolaan data dalam dilakukan secara cepat, lebih mudah, dan efisien. Algoritma pengurutan atau sorting algorithm adalah algoritma yang menyimpan suatu daftar atau deret baik data berupa angka atau teks, pada suatu urutan tertentu, biasanya urutannya dari kecil ke besar (ascending) atau sebaliknya descending, pada pengurutan ini diperlukan untuk mengoptimalkan kecepatan pemrosesan. Semakin efisien suatu algoritma, maka pada saat dieksekusi atau dijalankan akan menghabiskan waktu yang lebih cepat.

Dari berbagai macam algoritma pengurutan yaitu *Bubble sort*, *Quick sort*, *Shell sort*, *Insertion sort*, Namun, yang dipakai sebagai dasar pengenalan dan pemahaman terhadap proses belajar mengajar dalam suatu institusi yaitu *bubble sort*. Algoritma pengurutan inilah yang menjadi fokus pembahasan pada karya ilmiah ini. Yang menjadi domain pada kesempatan ini adalah mencoba untuk menganalisis dan melakukan pengujian data dari algoritma bubble sort.

II. STUDI LITERATUR

Algoritma

Menurut Gun Gun dan Maulana algoritma adalah metode efektif yang diekspresikan sebagai rangkaian terbatas. Algoritma juga merupakan kumpulan perintah untuk menyelesaikan suatu masalah. Perintah-perintah ini dapat diterjemahkan secara bertahap dari awal hingga akhir. Masalah tersebut dapat berupa apa saja, dengan syarat untuk setiap permasalahan memiliki kriteria kondisi awal yang harus dipenuhi sebelum menjalankan sebuah algoritma. Algoritma juga memiliki pengulangan proses (iterasi) dan juga memiliki keputusan hingga keputusan selesai. (Maulana G, 2017)

Ada 3 definisi tentang algoritma yang dijelaskan oleh Suarga (2012:1), diantaranya : a) Teknik penyusunan langkah-langkah penyelesaian masalah dalam bentuk kalimat dengan jumlah kata terbatas tetapi tersusun secara logis dan sistematis. b) Suatu prosedur yang jelas untuk menyelesaikan suatu persoalan dengan menggunakan langkah-langkah tertentu dan terbatas jumlahnya c) Susunan langkah yang pasti, yang bila diikuti maka akan mentransformasi data input menjadi output yang berupa informasi. Algoritma merupakan suatu prosedur untuk menyelesaikan suatu masalah yang tersusun secara logis dan sistematis serta akan memperoleh data masukan menjadi keluaran yang diinginkan berupa informasi. (Suarga, 2002)

Pengurutan Data (Sorting)

Sorting adalah proses pengurutan data yang sebelumnya disusun secara acak atau tidak teratur menjadi urut dan teratur menurut suatu aturan tertentu. Biasanya pengurutan terbagi menjadi dua yaitu Ascending (pengurutan dari karakter/angka kecil ke karakter/angka besar dan Descending (pengurutan dari karakter/angka besar ke karakter/angka kecil). (Kumalasari, 2017)

Bubble Sort

Bubble sort mengurutkan data dengan cara membandingkan elemen pertama dengan elemen kedua, dimana cara kerjanya adalah dengan berulang-ulang melakukan proses looping (perulangan) terhadap elemen-elemen struktur data yang belum diurutkan. Nilai dari masing-masing elemen akan dibandingkan secara terus menerus sampai pengurutan selesai. Seringkali dikatakan metode ini kurang efisien namun sangat mudah dipahami. (Suryani, 2013)

Penyajian Psedoucode algoritma Bubble sort dapat dilihat dibawah ini:

```
for i=1 to (n-1) do
  for j=1 to (n-i) do
    if A[j] > A[j+1] then
      temp=A[j] A[j]=A[j+1] A[j+1]=temp
    end if
  end for
end for
```

III. METODE

Penyusunan penelitian ini menggunakan metode pengumpulan data yang bertujuan untuk memperoleh data-data yang diperlukan sehingga dapat ditarik menjadi informasi. Diharapkan dari informasi tersebut dapat diketahui bahwa algoritma bubble sort mudah dalam proses pengurutan data. Berikut metode yang digunakan dalam penelitian ini

1. Studi literatur, yaitu suatu metode untuk mendapatkan informasi dan melakukan pengumpulan data dengan membaca dan mempelajari berbagai literatur-literatur antara lain bersumber dari buku, jurnal, modul, referensi internet, dan lain-lain yang mana sumber-sumber tersebut berhubungan dengan masalah yang diangkat sehingga dapat membantu dalam menyelesaikan permasalahan yang ada.
2. Pengujian Data, Ini adalah tahap penerapan algoritma bubble sort. Pengujian data dilakukan pada deret bilangan secara acak

IV. HASIL DAN PEMBAHASAN

Analisis Bubble Sort

Bubble Sort sangat mudah diimplementasikan, tetapi cenderung sangat lambat untuk dijalankan. (Chauhan & Duggal, 2020) Algoritma bekerja dengan menganalisis berbagai item dari kiri ke kanan dan membandingkan setiap pasangan elemen yang berdekatan, tukar jika item kiri lebih besar dari kanan. Dengan kata lain, elemen terbesar telah menggelembung ke bagian atas array. Algoritme terus melakukan proses ini hingga menelusuri daftar tanpa menukar elemen apa pun yang berarti semua elemen array atau daftar telah diurutkan. (Kunjwal, 2015)

Bubble sort adalah algoritma pengurutan tertua dan paling sederhana untuk diimplementasikan. Algoritma ini juga cukup mudah dipahami. Algoritma ini bekerja dengan cara membandingkan nilai setiap elemen pada tabel dengan elemen setelahnya, dan menukar nilainya jika sesuai dengan kondisi yang dibutuhkan. Proses ini akan terus berulang sampai semua elemen pada tabel sudah diproses dan elemen pada tabel sudah terurut. *Bubble sort* harus melakukan banyak perbandingan ketika lebih banyak elemen data perlu disortir. Meskipun algoritma bubble sort cukup mudah diimplementasikan dan tidak efisien jika dibandingkan dengan algoritma sorting lainnya. *Bubble sort* adalah algoritma paling sederhana yang bekerja dengan menukar elemen di sebelahnya jika urutannya salah. (Yulian dkk., t.t.)

Bubble sort adalah algoritma pengurutan dengan membandingkan elemen yang dipilih dengan elemen berikutnya (Cheema dkk., 2017). Algoritma jenis ini termasuk dalam algoritma sort comparison. Ini karena perbandingan dibuat antara setiap elemen array yang disediakan oleh proses. Algoritma ini bekerja dengan mem-parsing elemen array dari kiri ke kanan kemudian membandingkan setiap elemen yang lokasinya tidak jauh dari nilai dan menukar lokasi jika elemen kiri lebih besar dari kanan. Proses ini diulang sampai setiap item dalam urutan yang benar. Misalnya pada contoh dataset berupa nilai 26, 10, 13, 5, 8, 3, 21, 14 dan 9, proses pengurutan data menggunakan algoritma *Bubble Sort* adalah sebagai berikut.

Urutan data dalam array:

26	10	13	5	8	3	21	14	9
----	----	----	---	---	---	----	----	---

Iterasi i=1 j=1

10	26	13	5	8	3	21	14	9
----	----	----	---	---	---	----	----	---

Iterasi i=1 j=2

10	13	26	5	8	3	21	14	9
----	----	----	---	---	---	----	----	---

Iterasi i=1 j=3

10	13	5	26	8	3	21	14	9
----	----	---	----	---	---	----	----	---

Iterasi i=1 j=4

10	13	5	8	26	3	21	14	9
----	----	---	---	----	---	----	----	---

Iterasi i=1 j=5

10	13	5	8	3	26	21	14	9
----	----	---	---	---	----	----	----	---

Iterasi i=1 j=6

10	13	5	8	3	21	26	14	9
----	----	---	---	---	----	----	----	---

Iterasi i=1 j=7

10	13	5	8	3	21	26	14	9
----	----	---	---	---	----	----	----	---

Iterasi i=1 j=8

10	13	5	8	3	21	14	26	9
----	----	---	---	---	----	----	----	---

Iterasi i=2 j=1

10	13	5	8	3	21	14	26	9
----	----	---	---	---	----	----	----	---

Iterasi i=2 j=2

10	5	13	8	3	21	14	26	9
----	---	----	---	---	----	----	----	---

Iterasi i=2 j=3

10	5	8	13	3	21	14	26	9
----	---	---	----	---	----	----	----	---

Iterasi i=2 j=4

10	5	8	3	13	21	14	26	9
----	---	---	---	----	----	----	----	---

Iterasi i=2 j=5

10	5	8	3	13	21	14	26	9
----	---	---	---	----	----	----	----	---

Iterasi i=2 j=6

10	5	8	3	13	14	21	9	26
----	---	---	---	----	----	----	---	----

Iterasi i=2 j=7

10	5	8	3	13	14	9	21	26
----	---	---	---	----	----	---	----	----

Iterasi i=2 j=8

10	5	8	3	13	14	9	21	26
----	---	---	---	----	----	---	----	----

Iterasi i=3 j=1

5	10	8	3	13	14	9	21	26
---	----	---	---	----	----	---	----	----

Iterasi i=3 j=2

5	8	10	3	13	14	9	21	26
Iterasi i=3 j=3								
5	8	3	10	13	14	9	21	26
Iterasi i=3 j=4								
5	8	3	10	13	14	9	21	26
Iterasi i=3 j=5								
5	8	3	10	13	14	9	21	26
Iterasi i=3 j=6								
5	8	3	10	13	9	14	21	26
Iterasi i=3 j=7								
5	8	3	10	13	9	14	21	26
Iterasi i=3 j=6								
5	8	3	10	13	9	14	21	26
Iterasi i=4 j=1								
5	8	3	10	13	9	14	21	26
Iterasi i=4 j=2								
5	3	8	10	13	9	14	21	26
Iterasi i=4 j=3								
5	8	3	10	13	9	14	21	26
Iterasi i=4 j=4								
5	8	3	10	13	9	14	21	26
Iterasi i=4 j=5								
5	8	3	10	9	13	14	21	26
Iterasi i=5 j=1								
5	8	3	10	9	13	14	21	26
Iterasi i=5 j=2								
5	3	8	10	9	13	14	21	26
Iterasi i=5 j=3								
5	3	8	10	9	13	14	21	26
Iterasi i=5 j=4								
5	3	8	9	10	13	14	21	26
Iterasi i=6 j=1								
3	5	8	9	10	13	14	21	26

Data sudah terurut pada iterasi i=6 dan j=1 dan proses selanjutnya tidak terjadi tukar tempat atau swap karena angka disebelah kiri lebih kecil dari angka disebelah kanan.

Program Sorting data dengan Dev C++

```
1 #include <iostream>
2 using namespace std;
3
4 void masukan(int a[], int n)
5 { for (int i=1; i<=n; i++)
6   { cout<<"Data ke "<<i<<" = ";
7     cin>>a[i]; }
8 }
9
10 void urutkan(int a[], int n)
11 { int temp;
12   for (int i=0; i<n; i++)
13     for (int j=0; j<(n-1); j++)
14       if (a[j]>a[j+1])
15         { temp=a[j];
16           a[j]=a[j+1];
17           a[j+1]=temp; }
18 }
19
20 void tampil(int a[], int n)
21 { for (int i=0; i<n; i++)
22   cout<<a[i]<<" ";
23 }
24
25 int main() {
26   int a[100];
27   int jml;
28   cout<<"Banyak data = ";cin>>jml;
29   masukan(a, jml);
30   urutkan(a, jml);
31   tampil(a, jml);
32 }
33
```

Output

```
C:\Users\Ruang104\Documents\sorting.exe
Banyak data = 9
Data ke 1 = 26
Data ke 2 = 19
Data ke 3 = 13
Data ke 4 = 5
Data ke 5 = 8
Data ke 6 = 3
Data ke 7 = 21
Data ke 8 = 14
Data ke 9 = 9

Setelah diurutkan 3 5 8 9 10 13 14 21 26
-----
Process exited after 57.5 seconds with return value 0
Press any key to continue . . .
```

V. KESIMPULAN

Algoritma *bubble sort* adalah salah satu metode yang digunakan dalam pengurutan data yaitu dengan menukarkan posisi data yang telah dibandingkan sesuai dengan kriteria yang diinginkan. Pengurutan data dengan algoritma *bubble sort* sangat efisien jika jumlah datanya sedikit, sedangkan jika jumlah datanya besar, pengujian data pada algoritma *bubble sort* akan memakan waktu lama, algoritma *bubble sort* merupakan algoritma yang sederhana dan mudah dipahami.

VI. UCAPAN TERIMA KASIH

Terima kasih kepada tempat mengabdikan kami di Politeknik Unggul LP3M yang sudah memberikan motivasi terhadap kami dan terima kasih kepada keluarga kami yang paling kami sayangi.

VII. REFERENSI

- Alkareem Alyasseri, Z. A., Al-Attar, K., & Naser Yousif, M. (2015). Hybrid Bubble and Merge Sort Algorithms Using Message Passing Interface. *Journal of Computer Science & Computational Mathematics*, 67–73. <https://doi.org/10.20967/jscsm.2015.04.002>
- Chauhan, Y., & Duggal, A. (2020). Different Sorting Algorithms comparison based upon the Time Complexity Anuj Duggal Different Sorting Algorithms comparison based upon the Time Complexity. *IJRAR19L2026 International Journal of Research and Analytical Reviews (IJRAR)* www.ijrar.org, 114. <https://doi.org/10.1729/Journal.24472>
- Cheema, S. M., Sarwar, N., & Yousaf, F. (2017). Contrastive analysis of bubble & merge sort proposing hybrid approach. *2016 6th International Conference on Innovative Computing Technology, INTECH 2016*, 371–375. <https://doi.org/10.1109/INTECH.2016.7845075>
- Kumalasari, D. (2017). Analisa Perbandingan Kompleksitas Algoritma Bubble Sort, Cocktail Sort Dan Comb Sort Dengan Bahasa Pemrograman C++. Dalam *Journal Speed-Sentra Penelitian Engineering dan Edukasi* (Vol. 9). CDROM.
- Kunjwal, P. (2015). *BUBBLE SORT*.
- Maulana, G. (2017). *Pembelajaran Dasar Algoritma dan Pemrograman menggunakan EL-goritma berbasis Web*.
- Nasser, M., Abdi Alkareem Alyasseri, Z., & Al-Attar, K. (2014). Parallelize Bubble Sort Algorithm Using OpenMP Healthcare Monitoring System View project Parallelize Bubble Sort Algorithm Using OpenMP. Dalam *International Journal of Advanced Research in Computer Science and Software Engineering* (Vol. 4, Nomor 1). <http://www.booksshouldbefree.com/>
- Purnomo, R., & Putra, T. D. (2023). Theoretical Analysis of Standard Selection Sort Algorithm. *Sinkron*, 8(2), 666–673. <https://doi.org/10.33395/sinkron.v8i2.12153>
- Rihartanto, R., Susanto, A., & Rizal, A. (2017). Performance of parallel computing in bubble sort algorithm. *Indonesian Journal of Electrical Engineering and Computer Science*, 7(3), 861–866. <https://doi.org/10.11591/ijeecs.v7.i3.pp861-866>
- Rizka Poetra, D., & Hayati, N. (2022). Performa Algoritma Bubble Sort Dan Quick Sort Pada Framework Flutter Dan Dart SDK (Studi Kasus Aplikasi E-Commerce). *Hal; Jl. Sawo Manila*, 9(2), 7806700. <http://jurnal.mdp.ac.id>
- Sameer, S., Al-Dabbagh, M., & Hazim Barnouti, N. (2016). Parallel Quicksort Algorithm using OpenMP Computer Science Department of IMA View project FAST HYBRID STRING MATCHING ALGORITHM BASED ON THE QUICK-SKIP AND TUNED BOYER-MOORE ALGORITHMS View project International Journal of Computer Science and Mobile Computing Parallel Quicksort Algorithm using OpenMP. *Int. Journal of Computer Science & Mobile Computing*, 5(6), 372–382. <https://doi.org/10.6084/M9.FIGSHARE.3470033>

Shabaz, M., & Kumar, A. (2019). SA sorting: A novel sorting technique for large-scale data. *Journal of Computer Networks and Communications*, 2019. <https://doi.org/10.1155/2019/3027578>

Suarga. (2002). *ALGORITMA (LOGIKA PEMROGRAMAN)*.

Suryani, D. (2013). Perbandingan Metode Bubble Sort dan Selection Sort Terhadap Efisiensi Memori. *Prosiding Bali*.

Vilchez, R. N. (2020). Modified Selection Sort Algorithm Employing Boolean and Distinct Function in a Bidirectional Enhanced Selection Technique. *International Journal of Machine Learning and Computing*, 10(1), 93–98. <https://doi.org/10.18178/ijmlc.2020.10.1.904>

Yulian, D., Saputra, R., Andryana, S., Sholihati, I. D., Fakultas,), Komunikasi, T., Informatika, D., Nasional, U., Sawo Manila, J., Minggu, P., Selatan, J., Khusus, D., & Jakarta, I. (t.t.). *ANALISIS PERBANDINGAN ALGORITMA BUBBLE SORT DAN SELECTION SORT PADA SISTEM PENDUKUNG KEPUTUSAN PEMILIHAN TEMPAT KOST BERBASIS IOS (IPHONE OPERATING SYSTEM)*.

Zutshi, A., & Goswami, D. (2021). Systematic review and exploration of new avenues for sorting algorithm. *International Journal of Information Management Data Insights*, 1(2). <https://doi.org/10.1016/j.jjime.2021.100042>