

Implementasi Algoritma Bubble Sort Terhadap 2 Buah Model Varian Pengurutan Data Menggunakan Bahasa Program Java

Endang Sunandar¹; Indrianto²

¹Jurusan Sistem Komputer Universitas Raharja

Jalan Jendral Sudirman No 40 Modernland Cikokol Tangerang

²Jurusan Teknik Informatika Institut Teknologi PLN

Jalan Lingkar Luar Barat Duri Kosambi, Cengkareng Jakarta Barat

¹ endang.sunandar@raharja.info

² indrianto@itpln.ac.id

ABSTRACT

There are various kinds of data sorting methods that we know of which are the Bubble Sort, Selection Sort, Insertion Sort, Quick Sort, Shell Sort, Heap Sort, and Radix Sort methods. All of these methods have advantages and disadvantages of each, whose use is determined based on needs. Each method has a different algorithm, where different algorithms affect the execution time. One interesting algorithm to be implemented on 2 variant models of data sorting is the Bubble Sort algorithm, the reason is that this algorithm has a fairly long and detailed process flow to produce an ordered data sequence from a previously unordered data sequence. Two (2) data sorting variant models that will be implemented using the Bubble Sort algorithm are: Ascending data sorting variants moving from left to right, and Descending data sorting variants moving from left to right. And the device used in implementing the Bubble Sort algorithm is the Java programming language.

Keywords: Bubble Sort, Ascending, Descending, Variants, Java

ABSTRAK

Terdapat berbagai macam metode pengurutan data yang kita ketahui diantaranya adalah metode Bubble Sort, Selection Sort, Insertion Sort, Quick Sort, Shell Sort, Heap Sort, dan Radix Sort. Seluruh metode tersebut memiliki kelebihan dan kelemahan masing-masing, yang penggunaannya ditentukan berdasarkan kebutuhan. Setiap metode memiliki algoritma yang berbeda, dimana perbedaan algoritma ini mempengaruhi waktu eksekusi. Salah satu algoritma yang menarik untuk diimplementasikan terhadap 2 buah model varian pengurutan data adalah algoritma Bubble Sort, alasannya adalah bahwa algoritma ini memiliki alur proses yang cukup panjang dan detail untuk menghasilkan deretan data terurut dari deretan data tidak terurut sebelumnya. Dua (2) buah model varian pengurutan data yang akan diimplementasikan menggunakan algoritma Bubble Sort ini adalah : varian pengurutan data secara Ascending bergerak dari kiri ke kanan, dan varian pengurutan data secara Descending bergerak dari kiri ke kanan. Dan perangkat yang digunakan dalam pengimplementasian algoritma Bubble Sort ini adalah bahasa pemrograman Java.

Kata kunci: Bubble Sort, Ascending, Descending, Varian, Java

1. PENDAHULUAN

Berbagai pola algoritma pengurutan data yang ada memiliki kelebihan dan kekurangan masing-masing. Penggunaan masing-masing algoritma tersebut sangat tergantung dari kebutuhan dan kondisi yang ada. Umumnya pemilihan suatu algoritma pengurutan data didasarkan pada kesederhanaan / keringkasan alurnya dan kecepatan proses eksekusinya. Beberapa metode pengurutan data yang ada diantaranya adalah Selection Sort, Insertion Sort, Bubble Sort, Merge Sort, Quick Sort, Shell Sort, Heap Sort, dan Radix Sort. Dalam penelitian ini kajian yang dilakukan adalah tentang implementasi algoritma Bubble Sort terhadap 2 buah model varian pengurutan data. Pertimbangan memilih algoritma Bubble Sort adalah karena algoritma ini adalah salah satu algoritma pengurutan data yang memiliki alur cukup panjang dan detail dalam setiap langkahnya untuk menghasilkan deretan data terurut dari deretan data tidak terurut sebelumnya. Kemudian dalam algoritma ini dikenal dengan apa yang namanya Tahap dan Langkah. Tahap adalah fase pengurutan data pada suatu periode tertentu, sedangkan Langkah adalah urutan-urutan pengurutan yang berada pada setiap Tahap. Ilustrasinya adalah sbb:

Tahap 1 :

Langkah 1 :

Langkah 2 :

.....

Langkah (n-1) :

Tahap 2 :

Langkah 1 :

Langkah 2 :

.....

Langkah (n-2) :

Tahap m :

Langkah 1 :

Langkah 2 :

.....

Langkah (n-m) :

Hasil yang diharapkan dari penelitian ini adalah untuk memberikan gambaran alur logika dari 2 buah model varian pengurutan data dengan algoritma Bubble Sort menggunakan bahasa program Java.

Penelitian sebelumnya yang pernah ada adalah “Penggunaan Algoritma Sorting Bubble Sort Untuk Penentuan Nilai Prestasi Siswa” [1]. Dalam penelitian ini digambarkan bagaimana kumpulan data nilai dari setiap siswa dikumpulkan dan diurutkan secara Descending dengan algoritma Bubble Sort untuk menentukan urutan ranking prestasi siswa, menggunakan bahasa program C++. Penelitian selanjutnya adalah “Analisis Perbandingan Algoritma Bubble Sort, Merge Sort, dan Quick Sort dalam Proses Pengurutan Kombinasi Angka dan Huruf” [2]. Dalam penelitian ini menjelaskan tentang perbandingan dari ke-tiga algoritma pengurutan data tersebut dalam hal mengurutkan kombinasi angka dan huruf. Selanjutnya adalah penelitian dengan judul “Analisis Perbandingan Penggunaan Algoritma Pengurutan Data Dengan Metoda Bubble Sort, Metoda Selection Sort, Metoda Insertion Sort”[3]. Dalam pembahasan ini dijelaskan tentang percobaan proses pengurutan terhadap 7 buah data tidak terurut, untuk algoritma Bubble Sort membutuhkan

36 proses perbandingan data, algoritma Selection Sort membutuhkan 20 proses perbandingan data, dan Insertion Sort membutuhkan 16 proses perbandingan data.

2. METODE / PERANCANGAN PENELITIAN

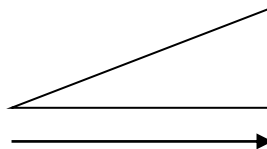
Pada penelitian ini, digunakan metode algoritma Bubble Sort terhadap 2 buah model varian pengurutan data yaitu pengurutan data secara : Ascending bergerak dari Kiri ke Kanan, dan Descending bergerak dari Kiri ke Kanan. Untuk memudahkan gambaran tentang alur proses dari algoritma Bubble Sort terhadap 2 buah model varian pengurutan data ini, kita menggunakan deretan data array seperti di bawah. Dan deretan data yang digunakan hanya 6 buah saja dengan alasan bahwa jika deretan datanya lebih banyak maka penjelasan alur proses dari algoritma Bubble Sort ini akan menjadi lebih banyak dan lebih panjang. Berikut deretan datanya :

Index	0	1	2	3	4	5
Urutan	1	2	3	4	5	6
Data Awal	32	15	3	88	41	37

Dan pembahasan algoritma Bubble Sort dari masing-masing varian akan dibahas pada setiap sub-bab di bawah ini :

2.1. Ascending dari Kiri ke Kanan

Algoritma Bubble Sort dalam metode pengurutan data varian model ini adalah bahwa alur proses akan mendorong nilai data yang paling besar ke posisi paling kanan, demikian seterusnya untuk nilai data yang besar berikutnya diletakkan di posisi sebelah kiri dari nilai data terbesar sebelumnya yang sudah diletakkan di posisi sebelah kanan. Gambaran ilustrasinya adalah seperti ini:



Tahap 1 : (sumber datanya adalah Data Awal)

1. Data (1) dan data (2) dibandingkan, jika data (2) < dari data (1) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
2. Data (2) dan data (3) dibandingkan, jika data (3) < dari data (2) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
3. Data (3) dan data (4) dibandingkan, jika data (4) < dari data (3) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
4. Data (4) dan data (5) dibandingkan, jika data (5) < dari data (4) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
5. Data (5) dan data (6) dibandingkan, jika data (6) < dari data (5) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.

Hasil dari pengurutan Tahap 1 adalah :

Urutan	1	2	3	4	5	6
Hasil Tahap 1	15	3	32	41	37	88

Hasil dari algoritma Tahap 1 ini adalah bahwa proses algoritma menempatkan / mendorong nilai yang paling besar ke posisi paling kanan.

Tahap 2 : (sumber datanya adalah Hasil Tahap 1)

Hasil dari pengurutan Tahap 2 adalah :

Urutan	1	2	3	4	5	6
Hasil Tahap 2	3	15	32	37	41	88

Tahap 3 : (sumber datanya adalah Hasil Tahap 2)

1. Data (1) dan data (2) dibandingkan, jika data (2) < dari data (1) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
2. Data (2) dan data (3) dibandingkan, jika data (3) < dari data (2) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
3. Data (3) dan data (4) dibandingkan, jika data (4) < dari data (3) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.

Hasil dari pengurutan Tahap 3 adalah :

Urutan	1	2	3	4	5	6
Hasil Tahap 3	3	15	32	37	41	88

Tahap 4 : (sumber datanya adalah Hasil Tahap 3)

1. Data (1) dan data (2) dibandingkan, jika data (2) < dari data (1) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
2. Data (2) dan data (3) dibandingkan, jika data (3) < dari data (2) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.

Hasil dari pengurutan Tahap 4 adalah :

Urutan	1	2	3	4	5	6
Hasil Tahap 4	3	15	32	37	41	88

Tahap 5 : (sumber datanya adalah Hasil Tahap 4)

1. Data (1) dan data (2) dibandingkan, jika data (2) < dari data (1) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.

Hasil dari pengurutan Tahap 5 adalah :

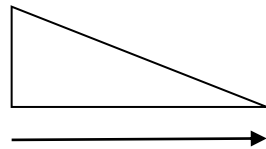
Urutan	1	2	3	4	5	6
Hasil Tahap 5	3	15	32	37	41	88

Sampai dengan Tahap 5 ini (terakhir) proses pengurutan data Ascending yang bergerak dari Kiri ke Kanan selesai. Disini bisa dianalisa kaitan antara Tahap, Langkah, dan Jumlah Data. Jika misal variable Tahap = T, variable Langkah = H, dan variable Jumlah Data = N, maka dapat dibuat persamaan matematika sebagai berikut :

$$N = T + H, \text{ dan } T = N - H, \text{ dan } H = N - T$$

2.2. Descending dari Kiri ke Kanan

Algoritma Bubble Sort dalam metode pengurutan data varian model ini adalah kita akan mendorong nilai data yang paling kecil ke posisi paling kanan, demikian seterusnya untuk nilai data yang kecil berikutnya diletakkan di posisi sebelah kiri dari nilai terkecil sebelumnya yang sudah diletakkan di posisi sebelah kanan. Gambaran ilustrasinya adalah seperti ini :



Tahap 1 : (sumber datanya adalah Data Awal)

1. Data (1) dan data (2) dibandingkan, jika data (2) > dari data (1) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
2. Data (2) dan data (3) dibandingkan, jika data (3) > dari data (2) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
3. Data (3) dan data (4) dibandingkan, jika data (4) > dari data (3) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
4. Data (4) dan data (5) dibandingkan, jika data (5) > dari data (4) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
5. Data (5) dan data (6) dibandingkan, jika data (6) > dari data (5) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.

Hasil dari pengurutan Tahap 1 adalah :

Urutan	1	2	3	4	5	6
Hasil Tahap 1	32	15	88	41	37	3

Tahap 2 : (sumber datanya adalah Hasil Tahap 1)

1. Data (1) dan data (2) dibandingkan, jika data (2) > dari data (1) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.

2. Data (2) dan data (3) dibandingkan, jika data (3) > dari data (2) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
3. Data (3) dan data (4) dibandingkan, jika data (4) > dari data (3) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
4. Data (4) dan data (5) dibandingkan, jika data (5) > dari data (4) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.

Hasil dari pengurutan Tahap 2 adalah :

Urutan	1	2	3	4	5	6
Hasil Tahap 2	32	88	41	37	15	3

Tahap 3 : (sumber datanya adalah Hasil Tahap 2)

1. Data (1) dan data (2) dibandingkan, jika data (2) > dari data (1) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
2. Data (2) dan data (3) dibandingkan, jika data (3) > dari data (2) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
3. Data (3) dan data (4) dibandingkan, jika data (4) > dari data (3) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.

Hasil dari pengurutan Tahap 3 adalah :

Urutan	1	2	3	4	5	6
Hasil Tahap 3	88	41	37	32	15	3

Tahap 4 : (sumber datanya adalah Hasil Tahap 3)

1. Data (1) dan data (2) dibandingkan, jika data (2) > dari data (1) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.
2. Data (2) dan data (3) dibandingkan, jika data (3) > dari data (2) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.

Hasil dari pengurutan Tahap 4 adalah :

Urutan	1	2	3	4	5	6
Hasil Tahap 4	88	41	37	32	15	3

Tahap 5 : (sumber datanya adalah Hasil Tahap 4)

1. Data (1) dan data (2) dibandingkan, jika data (2) > dari data (1) maka saling ditukarkan posisinya, jika tidak maka posisinya tetap tidak ada pertukaran. Posisi data lainnya tetap, tidak ada pertukaran.

Hasil dari pengurutan Tahap 5 adalah :

Urutan	1	2	3	4	5	6
Hasil Tahap 5	88	41	37	32	15	3

Sampai dengan Tahap 5 ini (terakhir) proses pengurutan data Descending yang bergerak dari Kiri ke Kanan selesai. Disini bisa dianalisa kaitan antara Tahap, Langkah, dan Jumlah Data. Jika misal variable Tahap = T, variable Langkah = H, dan variable Jumlah Data = N, maka dapat dibuat persamaan matematika sebagai berikut :

$$N = T + H, \text{ dan } T = N - H, \text{ dan } H = N - T$$

3. HASIL DAN PEMBAHASAN

Untuk penggambaran lebih jelas mengenai analisis algoritma Bubble Sort yang dimaksud, dapat dilihat dari listing program Java di bawah ini, dimana terdapat 2 buah model varian pengurutan datanya, yaitu pengurutan data secara Ascending bergerak dari Kiri ke Kanan, dan pengurutan data secara Descending bergerak dari Kiri ke Kanan. Berikut listing programnya :

3.1. Ascending Kiri ke Kanan

❖ Program Java – Pengurutan Data secara Ascending bergerak dari Kiri ke Kanan :

```
package analisa_algoritma;
/** Program Pengurutan Data Metode BUBBLE SORT
 * Bandung - 28 Mei 2020
 * @author Ir. Endang Sunandar, M.Kom
 */
public class BubbleSort_Ascending {
    public static void main(String args[]){
        int data[]={32,15,3,88,41,37};
        int pd=data.length;int buffer=0;int lp=0;
        System.out.println("Algoritma BUBBLE SORT - Pengurutan ASCENDING");
        System.out.println("Data Awal adalah : 32,15,3,88,41,37");
        System.out.println("Panjang data array adalah : "+pd+" data");
        System.out.println("-----");
        for (int r=0;r<(pd-1);r++){
            lp=r+1;System.out.println("Hasil Tahap ke - "+(lp)+" : ");
            for (byte i=0;i<(pd-lp);i++){
                if (data[i]>data[i+1]){
                    buffer=data[i];
                    data[i]=data[i+1];
                    data[i+1]=buffer;
                }//akhir if
                System.out.print(" Langkah ke - "+(i+1)+" ---> ");
                for (int buf=0;buf<pd;buf++){
                    System.out.print(data[buf]+" ");
                }//akhir for buf
                System.out.println("");
            }//akhir for i
            System.out.println("");
        }//akhir looping r
        System.out.println("-----");
    }//akhir public static void
} //akhir public class
```

❖ **Hasil Output dari eksekusi program diatas adalah :**

Algoritma BUBBLE SORT - Pengurutan ASCENDING

Data Awal adalah : 32,15,3,88,41,37

Panjang data array adalah : 6 data

Hasil Tahap ke - 1 :

Langkah ke - 1 ---> 15 32 3 88 41 37

Langkah ke - 2 ---> 15 3 32 88 41 37

Langkah ke - 3 ---> 15 3 32 88 41 37

Langkah ke - 4 ---> 15 3 32 41 88 37

Langkah ke - 5 ---> 15 3 32 41 37 88

Hasil Tahap ke - 2 :

Langkah ke - 1 ---> 3 15 32 41 37 88

Langkah ke - 2 ---> 3 15 32 41 37 88

Langkah ke - 3 ---> 3 15 32 41 37 88

Langkah ke - 4 ---> 3 15 32 37 41 88

Hasil Tahap ke - 3 :

Langkah ke - 1 ---> 3 15 32 37 41 88

Langkah ke - 2 ---> 3 15 32 37 41 88

Langkah ke - 3 ---> 3 15 32 37 41 88

Hasil Tahap ke - 4 :

Langkah ke - 1 ---> 3 15 32 37 41 88

Langkah ke - 2 ---> 3 15 32 37 41 88

Hasil Tahap ke - 5 :

Langkah ke - 1 ---> 3 15 32 37 41 88

BUILD SUCCESSFUL (total time: 0 seconds)

3.2. Descending Kiri ke Kanan

❖ **Program Java – Pengurutan Data secara Descending bergerak dari Kiri ke Kanan :**

```
package analisa_algoritma;
/** Program Pengurutan Data Metode BUBBLE SORT
 * Bandung - 28 Mei 2020
 * @author Ir. Endang Sunandar, M.Kom
 */
public class BubbleSort_Ascending {
    public static void main(String args[]){
        int data[]={32,15,3,88,41,37};
        int pd=data.length;int buffer=0;int lp=0;
        System.out.println("Algoritma BUBBLE SORT - Pengurutan DESCENDING");
        System.out.println("Data Awal adalah : 32,15,3,88,41,37");
        System.out.println("Panjang data array adalah : "+pd+" data");
        System.out.println("-----");
        for (int r=0;r<(pd-1);r++){
            lp=r+1;System.out.println("Hasil Tahap ke - "+(lp)+" : ");
            for (byte i=0;i<(pd-lp);i++){
                if (data[i]<data[i+1]){
                    buffer=data[i];
                    data[i]=data[i+1];
```



```

        data[i+1]=buffer;
    }//akhir if
    System.out.print(" Langkah ke - "+(i+1)+" ---> ");
    for (int buf=0;buf<pd;buf++){
        System.out.print(data[buf]+" ");
    }//akhir for buf
    System.out.println("");
} //akhir for i
System.out.println("");
} //akhir looping r
System.out.println("-----");
} //akhir public static void
} //akhir public class

```

❖ **Hasil Output dari eksekusi program diatas adalah :**

Algoritma BUBBLE SORT - Pengurutan DESCENDING

Data Awal adalah : 32,15,3,88,41,37

Panjang data array adalah : 6 data

Hasil Tahap ke - 1 :

Langkah ke - 1 ---> 32 15 3 88 41 37

Langkah ke - 2 ---> 32 15 3 88 41 37

Langkah ke - 3 ---> 32 15 88 3 41 37

Langkah ke - 4 ---> 32 15 88 41 3 37

Langkah ke - 5 ---> 32 15 88 41 37 3

Hasil Tahap ke - 2 :

Langkah ke - 1 ---> 32 15 88 41 37 3

Langkah ke - 2 ---> 32 88 15 41 37 3

Langkah ke - 3 ---> 32 88 41 15 37 3

Langkah ke - 4 ---> 32 88 41 37 15 3

Hasil Tahap ke - 3 :

Langkah ke - 1 ---> 88 32 41 37 15 3

Langkah ke - 2 ---> 88 41 32 37 15 3

Langkah ke - 3 ---> 88 41 37 32 15 3

Hasil Tahap ke - 4 :

Langkah ke - 1 ---> 88 41 37 32 15 3

Langkah ke - 2 ---> 88 41 37 32 15 3

Hasil Tahap ke - 5 :

Langkah ke - 1 ---> 88 41 37 32 15 3

BUILD SUCCESSFUL (total time: 0 seconds)

4. KESIMPULAN DAN SARAN

Berdasarkan hasil analisis yang dilakukan pada penggunaan algoritma Bubble Sort terhadap 2 buah model varian pengurutan data ini dapat disimpulkan beberapa hal sebagai berikut :

Pada model pengurutan data secara Ascending yang bergerak dari kiri ke kanan, alur proses yang dilakukan adalah secara bertahap melakukan proses penempatan nilai data terbesar ke posisi sebelah kanan. Demikian seterusnya hingga nilai data terkecil berada pada posisi paling kiri.

Pada model pengurutan data secara Descending yang bergerak dari kiri ke kanan, alur proses yang dilakukan adalah secara bertahap melakukan proses penempatan nilai data terkecil ke posisi sebelah kanan. Demikian seterusnya hingga nilai data terbesar berada pada posisi paling kiri.

Teknik pemrograman dalam bahasa Java untuk 2 buah model varian pengurutan data ini hampir persis sama, hanya ada beda sedikit pada penggunaan operation relationanl < atau > pada suatu data pada posisi [i] yang dibandingkan dengan data yang pada posisi [i+1].

Masih ada 2 buah model varian pengurutan data lainnya yaitu : Ascending bergerak dari Kanan ke Kiri, dan Descending bergerak dari Kanan ke Kiri. Jika melihat ilustrasi obyek masing-masing pengurutan maka model varian pengurutan data Ascending yang bergerak dari Kiri ke Kanan = model varian pengurutan data Descending yang bergerak dari Kanan ke Kiri. Dan model varian pengurutan data Descending yang bergerak dari Kiri ke Kanan = model varian pengurutan data Ascending yang bergerak dari Kanan ke Kiri. Dan dalam teknik pemrograman komputernya hanya dibedakan pada penggunaan operator relasi (< atau >) dan posisi start pergerakan perbandingan data (dari kiri atau dari kanan).

DAFTAR PUSTAKA

Beberapa sumber pustaka untuk penelitian ini adalah :

- [1] Gunawan Indra, Sumarno, Tambunan Heru Satria, [2019], "*Penggunaan Algoritma Sorting Bubble Sort Untuk Penentuan Nilai Prestasi Siswa*", Jurnal Sistem Informasi, Vol.8, No.2, Mei 2019 : 296 - 304, STIKOM Tunas Bangsa Pematang Siantar.
- [2] Sonita Anisya, Nurtaneo Febrian, [2015], "*Analisis Perbandingan Algoritma Bubble Sort, Merge Sort, dan Quick Sort dalam Proses Pengurutan Kombinasi Angka dan Huruf*", Jurnal Pseudocode, Vol. 2, No.2, 2015, Program Studi Teknik Informatika, Fakultas Teknik Universitas Bengkulu.
- [3] Gultom Darma Indra,[], "*Analisis Perbandingan Penggunaan Algoritma Pengurutan Data Dengan Metoda Bubble Sort, Metoda Selection Sort, Metoda Insertion Sort*", Jurnal Tekinkom, Jurusan Teknik Komputer, Politeknik Bisnis Indonesia, Jl Sriwijaya No 9(C-E) Pematang Siantar Sumatera Utara.
- [4] Reina, Gautama Josef Bernadi, [2013], "*Perbandingan Bubble Sort Dengan Insertion Sort Pada Bahasa Pemrograman C Dan Fortran*", Jurnal ComTech, Vol.4, No.2, Desember 2013, 1106-1115.
- [5] Shabaz Mohammad, Kumar Ashok, [2019], "*SA Sorting : A Novel Sorting Technique for Large-Scale Data*", Hindawi Journal of Computer Networks and Communications, Vol. 2019, Article ID 3027578.
- [6] Akhter Naeem, Idrees Muhammad, Rehman-Furqan-Ur,[2016], "*Sorting Algorithms – A Comparative Study*", International Journal of Computer Science and Information Security (IJCSIS), Vol.14, No.12, December 2016.
- [7] Triveei Dhwaneel, Trivedi Prathmesh, Singh Suraj,[2013], "*Min-Max Select Bubble Sorting Algorithm*", International Journal of Applied Information Systems (IJ AIS) – ISSN : 2249-0868, Foundation of Computer Science FCS, New York, USA.
- [8] Rohil Harish, Manisha, [2014], "*Run Time Bubble Sort – An Enhancement of Bubble Sort*", International Journal of Computer Trends and Technology (IJCTT), Vol.14, No.1-Aug 2014- ISSN : 2231-2803.

- [9] Saadeh Reem, Qataweh Mohammad, [2019], “*Performance Evaluation of Parallel Bubble Sort Algorithm On Supercomputer IMan1*”, International Journal of Computer Science & Information Technology (IJCSIT), Vol.11, No.3 : June 2019.
- [10] Olabiyisi S.O, Adewole O.A, [2008], “*On the Software Complexity Measures of Bubble Sort Algorithm*”, Medwell Journals, 2008.
- [11] Prayitno Prayitno, Basukesti Agus, Indrianingsih Yuliani, [2013], “Load Balancing Menggunakan Metode Bubble Sort Pada Sistem Database Dengan Multi Server”, Compiler, Vol.2,No.2, 2013, Print ISSN : 2252 – 3839, Online ISSN : 2549-2403.
- [12] Wang Shiyang, Wang Mujiangshan, [2018], “The Strong Connectivity of Bubble-Sort Star Graphs”, The Computer Journal, Vol.62, Issue 5, 15 August 2018