

MODUL III

SORTING AND SEARCHING

3.1 PERMASALAHAN

3.1.1 Spin Off Rental PS Rijal

Kalian masih ingat dengan sosok bernama Rijal yang memiliki rental PS? Ia telah kembali pada cerita kali ini dengan banyak tokoh baru lain-nya.

Di siang hari yang sangat terik ini. Rijal sedang sibuk di meja kasir rental PS miliknya ketika Eysar masuk. Eysar adalah teman dekat Rijal semasa perkuliahan dulu. Biasanya, Eysar datang ke sini dengan semangat untuk bermain PES, tapi kali ini wajahnya terlihat murung.

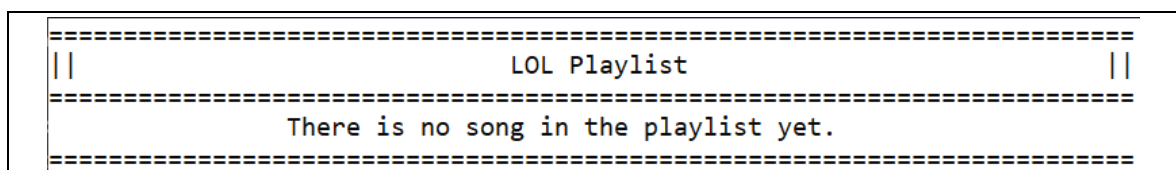
“Kenapa Sar? Biasanya langsung nyari stick, kok sekarang malah bengong?” tanya Rijal, penasaran.

Eysar duduk di sofa yang biasa digunakan pelanggan untuk menunggu giliran bermain. "Nggak tahu, Jal. Lagi galau Jal. Lagi nggak mood buat main" jawabnya sambil menghela napas. Rijal mengerutkan dahi. "LOL, galau kenapa tuh? Cerita dah. Kalau duduk doang di sini, nggak seru kan".

Eysar tersenyum tipis. "Kayaknya saya mau bikin playlist deh di Spotify. Buat nemenin kalau lagi galau begini". Rijal tertawa kecil. "LOL, playlist galau nih. Biar playlist-nya penuh nuansa sendu kayak hatimu?" godanya.

“LOL” kata Eysar sambil mengangguk, mengeluarkan ponsel dan membuka aplikasi Spotify yang masih kosong tanpa lagu di playlist barunya. Dengan sabar, ia mulai memilih lagu-lagu yang cocok. Rijal, yang duduk di sebelahnya sambil mengecek pemesanan pelanggan lain, sesekali memberikan rekomendasi lagu.

“Apa nama playlist buat galamu tu Sar?” kata Rijal yang tertarik dengan playlist Eysar. Eysar tertawa “Rijal Ganteng gimana?”. “LOL” kata Rijal sambil ikut tertawa. Eysar berkata sambil bersemangat “Nah itu aja gimana Jal? LOL Playlist?”. “LOL, iyadah” kata Rijal sambil tertawa.



***tampilan di saat belum ada lagu yang ditambahkan**

“Saya masukin sepuluh lagu dulu ga sih,” kata Eysar sambil menambahkan lagu-lagu pilihan-nya ke dalam playlist.

***Note : gunakan double linked list**

===== LOL Playlist =====		
1	Billy Joel - Piano Man (1973)	
5:39	Folk	
2	Bruno Mars dan Lady Gaga - Die With A Smile (2024)	
4:11	Pop	
3	.Feast - Nina (2024)	
4:37	Indonesian Rock	
4	Wave to Earth - Bad (2023)	
4:23	Korean Rock, Thai Indie	
5	Why Don't We - 8 Letters (2018)	
3:10	Pop	
6	The Smiths - There Is a Light That Never Goes Out (1986)	
4:4	Indie	
7	Pamungkas - To the Bone (2020)	
5:44	Indonesian Indie	
8	One Ok Rock - We are (2017)	
4:15	Pop Rock	
9	Lee Young Ji - Small girl (2024)	
3:9	R&B	
10	Green Day - 21 Guns (2009)	
5:21	Punk Pop	
=====		

***tampilan sesudah ada lagu yang ditambahkan**

***Note : atributnya ada judul lagu, nama artis, tahun rilis, durasi (dalam menit dan detik), dan genre**

Setelah playlistnya terisi dengan sepuluh lagu, Eysar mulai berpikir untuk mengurutkan lagu-lagu ini dengan lebih spesifik lagi. Kali ini, ia ingin mengurutkannya berdasarkan durasi lagu agar alurnya terasa lebih pas.

Rijal, yang sedang membereskan stik PS di rak, mendekat lagi setelah mendengar keinginan Eysar. “Kalau urutin dari durasi, berarti lagu paling pendek dulu, baru yang lebih panjang kan?”.

Eysar mengangguk. Dengan sedikit bantuan dari Rijal, ia memeriksa durasi setiap lagu di playlist dan mulai menyusun ulang. Satu per satu, lagu-lagu tersebut diurutkan dari yang terpendek hingga yang terpanjang. Setelah selesai, playlist itu terlihat lebih rapi dan teratur.

***Note : Mengurutkan menggunakan algoritma Bubble Sort (Asc) berdasarkan durasi menit dan detik.**

LOL Playlist		
1	Lee Young Ji - Small girl (1973)	
3:9	R&B	
2	Why Don't We - 8 Letters (2024)	
3:10	Pop	
3	The Smiths - There Is a Light That Never Goes Out (2024)	
4:4	Indie	
4	Bruno Mars dan Lady Gaga - Die With A Smile (2023)	
4:11	Pop	
5	One Ok Rock - We are (2018)	
4:15	Pop Rock	
6	Wave to Earth - Bad (1986)	
4:23	Korean Rock, Thai Indie	
7	.Feast - Nina (2020)	
4:37	Indonesian Rock	
8	Green Day - 21 Guns (2017)	
5:21	Punk Pop	
9	Billy Joel - Piano Man (2024)	
5:39	Folk	
10	Pamungkas - To the Bone (2009)	
5:44	Indonesian Indie	

*** tampilan sesudah lagu di urutkan berdasarkan menit dan detik**

Namun, Eysar tiba-tiba punya ide lain. "Jal, kalau saya urutinnya berdasarkan lagu terbaru yang rilis". Rijal tersenyum mengerti. "Jadi urutannya berdasarkan tahun rilis terbaru kan Sar?"

Eysar mengangguk lagi. Ia pun mengaktifkan opsi pengurutan otomatis berdasarkan tahun rilis, sehingga lagu-lagu terbaru akan berada di urutan teratas playlist.

***Note : Mengurutkan menggunakan algoritma Selection Sort (Desc) berdasarkan tahun rilis.**

LOL Playlist		
1	Bruno Mars dan Lady Gaga - Die With A Smile (2024)	
4:11	Pop	
2	.Feast - Nina (2024)	
4:37	Indonesian Rock	
3	Lee Young Ji - Small girl (2024)	
3:9	R&B	
4	Wave to Earth - Bad (2023)	
4:23	Korean Rock, Thai Indie	
5	Pamungkas - To the Bone (2020)	
5:44	Indonesian Indie	
6	Why Don't We - 8 Letters (2018)	
3:10	Pop	
7	One Ok Rock - We are (2017)	
4:15	Pop Rock	
8	Green Day - 21 Guns (2009)	
5:21	Punk Pop	
9	The Smiths - There Is a Light That Never Goes Out (1986)	
4:4	Indie	
10	Billy Joel - Piano Man (1973)	
5:39	Folk	

***tampilan sesudah lagu di urutkan berdasarkan tahun**

Akhirnya setelah cukup lama menyusun dan menata, Eysar menatap playlist-nya dengan senyum lega. Playlist itu kini rapi, terurut dari durasi, dan lagu-lagu terbaru.

Rijal menepuk pundaknya sambil tersenyum.

“Playlist siap menemani galaumu tuh Sar” candanya.

Eysar tertawa kecil. “LOL, makasi udah bantu ya Jal”.

Setelah playlist-nya tertata rapi, Eysar merasa playlist itu sudah cukup sempurna. Namun, suatu hari saat kembali ke rental PS Rijal, ia ingin mencari tahu letak salah satu lagu favoritnya yang ada di playlist.

“Jal, saya lagi pengen dengerin satu lagu di playlist nih, tapi lupa letaknya di urutan ke berapa” kata Eysar sambil membuka Spotify.

Rijal tertawa kecil. “LOL, lagu di playlist-mu kan udah ada sepuluh lebih sekarang. Kalau di-scroll satu-satu bisa lama, Sar”.

***Note: silahkan menambahkan lagu lagi secara bebas sampai 15 lagu.**

Eysar mengangguk setuju, lalu mulai mengetik judul lagu yang dia ingat di kolom pencarian dalam playlist-nya. Setelah beberapa detik, lagu yang ia cari muncul, lengkap dengan nomor urutnya.

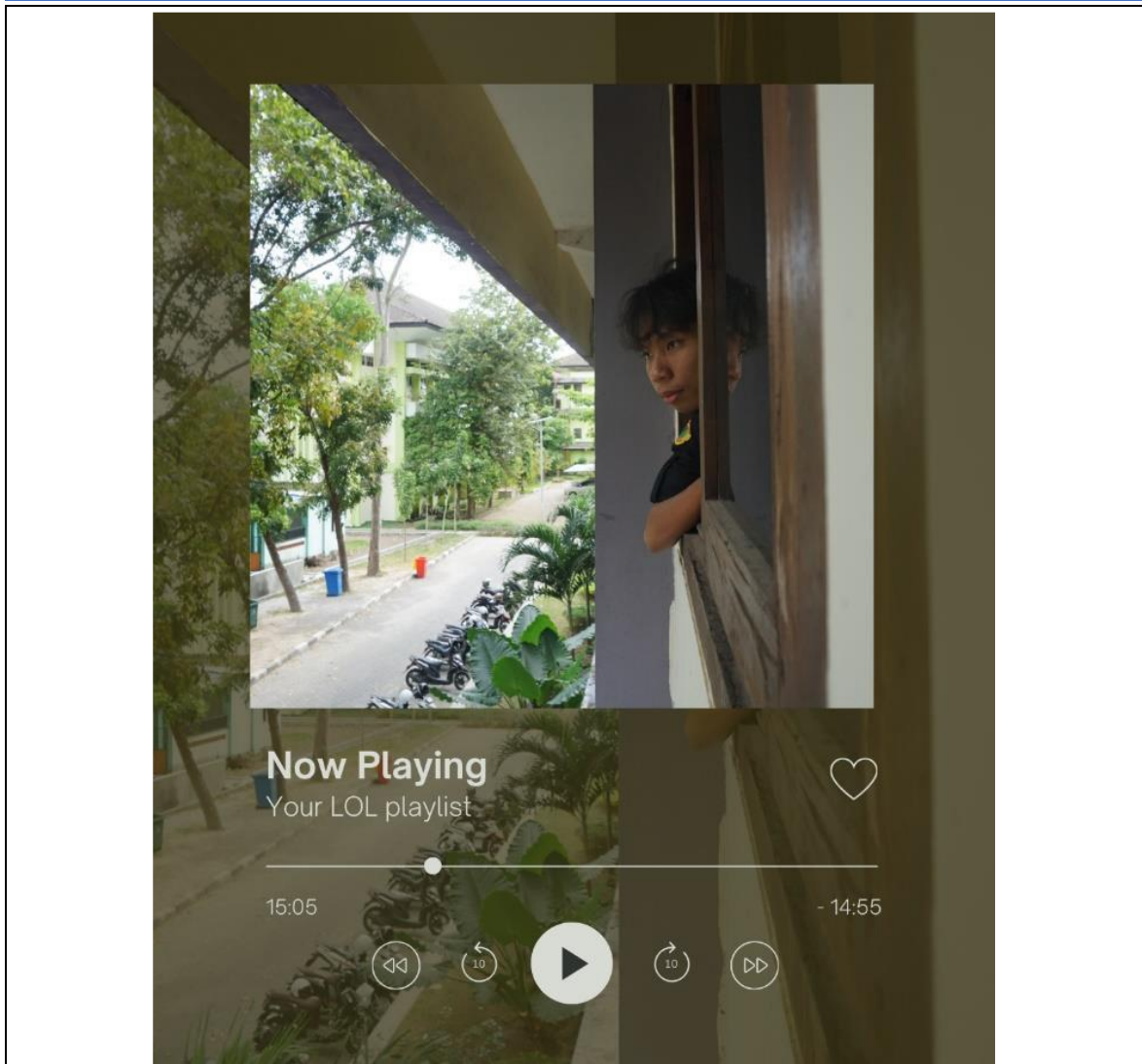
“Yes, ketemu! Lagunya ada di urutan nomor lima!” seru Eysar dengan senyum puas. Rijal tersenyum, lalu menggodanya. “Nah, enak kan? Makanya playlist-mu diatur. Biar pas nyari gampang.”

```
=====
||                               LOL Playlist                               ||
=====
Judul lagu yang ingin dicari : We are
Lagu yang sesuai dengan judul 'We are' ditemukan pada posisi ke - 7
=====
      7 |One Ok Rock - We are (2017)
      4:15                Pop Rock
=====
```

***tampilan hasil pencarian berdasarkan judul lagu menggunakan algoritman linear search.**

Eysar tertawa. “Benar juga, Jal. Sekarang nggak ribet lagi kalau mau cari-cari lagu favorit!”

Sejak saat itu, Eysar semakin senang menyusun dan memperbaiki playlist-nya, memastikan lagu-lagu favoritnya mudah ditemukan kapan saja. Rijal, yang melihat semangat temannya, hanya bisa tertawa kecil sambil kembali melayani pelanggan di rental PS.



3.1.2 Persoalan Readme

1. Lakukan pengurutan kembali setelah data di tambahkan sesuai ketentuan pada no 1 menggunakan algoritma yang sudah kalian buat sesuai *level*, lakukan penambahan secara *asc*, *desc*, dan *random* lalu lakukan *sorting* ke data tersebut sesuai perintah *level* yang kalian ambil dan bandingkan efektifitas waktu setiap *sorting*. Hitung waktu yang dibutuhkan setiap jenis algoritma *sorting* dalam melakukan pengurutan data bisa menggunakan `System.nanoTime()`; atau algoritma lain yang sesuai
2. Lakukan analisa dari hasil pengurutan yang telah dilakukan, bandingkan hasil setiap algoritma sorting masing-masing level manakah algoritma yang paling efektif dan efisien sesuai dengan data yang ada, jika hasil analisa dari setiap algoritma yang sudah ditentukan menunjukkan hasil yang tidak efektif/tidak efisien satusama lain, lakukan proses pengurutan dengan algoritma berbeda dan buktikan kenapa algoritma tersebut lebih baik dari sebelumnya.

3.2 HASIL PERCOBAAN

3.2.1 Spin Off Rental PS Rizal

1. Algoritma

- a. Kelas “Node” menyimpan informasi lagu seperti judul, artis, tahun rilis, durasi, dan genre; konstruktor menginisialisasi atribut dan menyediakan metode “getDurasiTotal” untuk menghitung durasi total dalam detik.
- b. Kelas “EasyAjalah” mengelola playlist lagu dan menyediakan metode untuk menambah, mengurutkan, mencari, dan menampilkan playlist.
- c. Metode “addLagu” menambahkan lagu baru ke akhir playlist atau sebagai lagu pertama jika playlist masih kosong.
- d. Membuat metode “urutkanBerdasarkanDurasi” untuk mengurutkan lagu berdasarkan durasi. Menggunakan algoritma bubble sort, metode ini mengurutkan lagu dalam playlist berdasarkan durasi secara menaik.
- e. Membuat metode “urutkanBerdasarkanRilis” untuk mengurutkan lagu berdasarkan tahun rilis. Dengan algoritma selection sort, metode ini mengurutkan playlist berdasarkan tahun rilis secara menurun.
- f. Metode “swapData” menukar data antara dua lagu dalam playlist saat melakukan pengurutan.
- g. Metode “cariLagu” mencari dan menampilkan informasi lagu berdasarkan judul yang diberikan, atau memberi tahu jika lagu tidak ditemukan.
- h. Metode “tampilkanPlaylist” menampilkan semua lagu dalam playlist dengan informasi lengkap atau memberi tahu jika playlist kosong.
- i. Kelas “Main” menginisialisasi objek “EasyAjalah”, menambahkan lagu, mengurutkan dan menampilkan playlist, serta mencari lagu tertentu.

2. Source Code

```
public class Node {
    String judul;
    String artis;
    int rilis;
    int menit;
    int detik;
    String genre;
    Node next;
    Node prev;

    public Node(String artis, String judul, int rilis, int menit, int
detik, String genre) {
        this.artis = artis;
        this.judul = judul;
        this.rilis = rilis;
```

```

        this.menit = menit;
        this.detik = detik;
        this.genre = genre;
        this.next = null;
        this.prev = null;
    }
    public int getDurasiTotal() {
        return menit * 60 + detik;
    }
}

public class EasyAjalah {
    Node head;
    Node tail;

    public void addLagu(String judul, String artis, int rilis, int
menit, int detik, String genre) {
        Node newNode = new Node(judul, artis, rilis, menit, detik,
genre);
        if (head == null) {
            head = tail = newNode;
        } else {
            tail.next = newNode;
            newNode.prev = tail;
            tail = newNode;
        }
    }

    public void urutkanBerdasarkanDurasi() {
        if (head == null){
            return;
        }
        boolean swapped;
        do {
            swapped = false;
            Node current = head;
            while (current.next != null) {
                if
current.getDurasiTotal() >
current.next.getDurasiTotal()) {
                    swapData(current, current.next);
                    swapped = true;
                }
                current = current.next;
            }
        } while (swapped);
    }

    public void urutkanBerdasarkanRilis() {
        if (head == null){
            return;
        }
        Node current = head;
        while (current != null) {
            Node maxNode = current;
            Node nextNode = current.next;
            while (nextNode != null) {
                if (nextNode.rilis > maxNode.rilis) {
                    maxNode = nextNode;
                }
                nextNode = nextNode.next;
            }
        }
    }
}

```

```

        if (maxNode != current) {
            swapData(current, maxNode);
        }
        current = current.next;
    }
}

private void swapData(Node node1, Node node2) {
    String tempJudul = node1.judul;
    String tempArtis = node1.artis;
    int tempRilis = node1.rilis;
    int tempMenit = node1.menit;
    int tempdetik = node1.detik;
    String tempGenre = node1.genre;

    node1.judul = node2.judul;
    node1.artis = node2.artis;
    node1.rilis = node2.rilis;
    node1.menit = node2.menit;
    node1.detik = node2.detik;
    node1.genre = node2.genre;
    node2.judul = tempJudul;
    node2.artis = tempArtis;
    node2.rilis = tempRilis;
    node2.menit = tempMenit;
    node2.detik = tempdetik;
    node2.genre = tempGenre;
}

public Node cariLagu(String judul) {
    Node current = head;
    int pos = 1;
    boolean ditemukan = false;

    System.out.println("=====
=====");
    System.out.println("||                                LOL
Playlist                                ||");

    System.out.println("=====
=====");
    System.out.println("Judul lagu yang ingin dicari : " +
    judul);

    while (current != null) {
        if (current.judul.equalsIgnoreCase(judul)) {
            System.out.println("Lagu yang sesuai dengan judul '"
            + judul + "' ditemukan pada posisi ke - " + pos);

            System.out.println("=====
=====");
            System.out.printf("%7d | %s - %s (%d)%n", pos,
            current.artis, current.judul, current.rilis);
            System.out.printf("%7d:%d      %s%n", current.menit,
            current.detik, current.genre);

            System.out.println("=====
=====");
            ditemukan = true;
            break;
        }
    }
}

```



```

        current = current.next;
        pos++;
    }
    if (!ditemukan) {
        System.out.println("Lagu dengan judul '" + judul + "'
tidak ditemukan dalam playlist.");

System.out.println("=====
=====");
    }
    return null;
}
public void tampilkanPlaylist() {
    Node current = head;
    int pos = 1;

System.out.println("=====
=====");
    System.out.println("||                               LOL
Playlist                               ||");

System.out.println("=====
=====");
    if(head==null){
        System.out.println("There is no song in the playlist
yet.");

System.out.println("=====
=====");
        }else{
            while (current != null) {
                System.out.printf("%7d | %s - %s (%d)%n", pos,
current.artis, current.judul, current.rilis);
                System.out.printf("%7d:%d      %s%n", current.menit,
current.detik, current.genre);

System.out.println("=====
=====");
                current = current.next;
                pos++;
            }
        }
    }

public class Main {
    public static void main(String[] args) {
        EasyAjalah playlist = new EasyAjalah();
        playlist.addLagu("Lee Young Ji", "Small girl", 1973, 3, 9,
"R&B");
        playlist.addLagu("Why Don't We", "8 Letters", 2024, 3, 10,
"Pop");
        playlist.addLagu("The Smiths", "There Is a Light That Never
Goes Out", 2024, 4, 4, "Indie");
        playlist.addLagu("Bruno Mars dan Lady Gaga", "Die With A
Smile", 2023, 4, 11, "Pop");
        playlist.addLagu("One Ok Rock", "We are", 2018, 4, 15, "Pop
Rock");
        playlist.addLagu("Wave to Earth", "Bad", 1986, 4, 23, "Korean
Rock, Thai Indie");
    }
}

```

```

        playlist.addLagu(".Feast", "Nina", 2020, 4, 37, "Indonesian
Rock");
        playlist.addLagu("Green Day", "21 Guns", 2017, 5, 21, "Punk
Pop");
        playlist.addLagu("Billy Joel", "Piano Man", 2024, 5, 39,
"Folk");
        playlist.addLagu("Pamungkas", "To the Bone", 2009, 5, 44,
"Indonesian Indie");

        playlist.tampilkanPlaylist();

System.out.println("=====
=====");
        System.out.println("=                                LOL Playlist
=");
        System.out.println("=                                Pengurutan Berdasarkan Durasi
=");

System.out.println("=====
=====");

        playlist.urutkanBerdasarkanDurasi();
        playlist.tampilkanPlaylist();

System.out.println("=====
=====");
        System.out.println("=                                LOL Playlist
=");
        System.out.println("=                                Pengurutan Berdasarkan Tahun
Rilis      =");

System.out.println("=====
=====");

        playlist.urutkanBerdasarkanRilis();
        playlist.tampilkanPlaylist();

        playlist.cariLagu("Runaway Baby");

        playlist.addLagu("Cinta Pertama dan Terakhir", "Sherina",
2009, 4, 13, "Pop");
        playlist.addLagu("To the Bone", "Pamungkas", 2020, 5, 12,
"Indie Pop");
        playlist.addLagu("Dia", "Anji", 2016, 3, 37, "Pop");
        playlist.addLagu("Pupus", "Dewa 19", 2000, 5, 6, "Rock");
        playlist.addLagu("A Whole New World", "Zayn Malik & Zhavia
Ward", 2019, 4, 2, "Soundtrack");
        playlist.addLagu("Rewrite the Stars", "Zac Efron & Zendaya",
2018, 3, 37, "Soundtrack");
        playlist.addLagu("Rewrite the Stars", "James Arthur & Anne-
Marie", 2018, 3, 47, "Pop");
        playlist.addLagu("Say You Won't Let Go", "James Arthur",
2016, 3, 31, "Pop");
        playlist.addLagu("Let Her Go", "Passenger", 2012, 4, 12,
"Folk Rock");
        playlist.addLagu("See You Again", "Wiz Khalifa ft. Charlie
Puth", 2015, 3, 49, "Hip Hop");
        playlist.addLagu("Blinding Lights", "The Weeknd", 2019, 3,
22, "Pop");

```

```

playlist.addLagu("Shape of You", "Ed Sheeran", 2017, 3, 53,
"Pop");
playlist.addLagu("Stay", "The Kid LAROI & Justin Bieber",
2021, 2, 21, "Pop");
playlist.addLagu("Someone Like You", "Adele", 2011, 4, 45,
"Pop");
playlist.addLagu("Levitating", "Dua Lipa", 2020, 3, 23,
"Pop");
playlist.addLagu("Rolling in the Deep", "Adele", 2010, 3, 49,
"Pop");
playlist.addLagu("Bad Guy", "Billie Eilish", 2019, 3, 14,
"Pop");
playlist.addLagu("Memories", "Maroon 5", 2019, 3, 9, "Pop");
playlist.addLagu("Counting Stars", "OneRepublic", 2013, 4,
17, "Pop Rock");
playlist.addLagu("Perfect", "Ed Sheeran", 2017, 4, 23,
"Pop");
playlist.addLagu("Uptown Funk", "Mark Ronson ft. Bruno Mars",
2014, 4, 30, "Funk Pop");
playlist.addLagu("Havana", "Camila Cabello", 2018, 3, 37,
"Pop");
playlist.addLagu("Cheap Thrills", "Sia", 2016, 3, 31, "Pop");
playlist.addLagu("Closer", "The Chainsmokers ft. Halsey",
2016, 4, 5, "EDM");
playlist.addLagu("Love Story", "Taylor Swift", 2008, 3, 57,
"Country Pop");
playlist.addLagu("Numb", "Linkin Park", 2003, 3, 7, "Rock");
playlist.addLagu("Can't Stop the Feeling!", "Justin
Timberlake", 2016, 3, 56, "Pop");
playlist.addLagu("Believer", "Imagine Dragons", 2017, 3, 24,
"Rock");
playlist.addLagu("Toxic", "Britney Spears", 2003, 3, 19,
"Pop");
playlist.addLagu("Senorita", "Shawn Mendes & Camila Cabello",
2019, 3, 11, "Pop");
playlist.addLagu("Photograph", "Ed Sheeran", 2014, 4, 18,
"Pop");
playlist.addLagu("Viva La Vida", "Coldplay", 2008, 4, 2,
"Alternative");
playlist.addLagu("Rockstar", "Post Malone ft. 21 Savage",
2017, 3, 38, "Hip Hop");
playlist.addLagu("Thinking Out Loud", "Ed Sheeran", 2014, 4,
41, "Pop");
playlist.addLagu("Don't Let Me Down", "The Chainsmokers ft.
Daya", 2016, 3, 28, "EDM");
playlist.addLagu("Shake It Off", "Taylor Swift", 2014, 3, 39,
"Pop");
playlist.addLagu("Old Town Road", "Lil Nas X", 2019, 2, 37,
"Country Rap");
playlist.addLagu("Faded", "Alan Walker", 2015, 3, 32, "EDM");
playlist.addLagu("Dance Monkey", "Tones and I", 2019, 3, 29,
"Pop");
playlist.addLagu("All of Me", "John Legend", 2013, 4, 29,
"Pop");
playlist.addLagu("Kangen", "Dewa 19", 1992, 5, 25, "Rock");
playlist.addLagu("Sampai Jadi Debu", "Banda Neira", 2013, 4,
5, "Indie");
playlist.addLagu("Sesuatu di Jogja", "Adhitia Sofyan", 2016,
4, 15, "Folk");
playlist.addLagu("Kasih Tak Sampai", "Padi", 2001, 4, 52,
"Rock");

```

```

        playlist.addLagu("Akad", "Payung Teduh", 2017, 4, 4, "Pop");
        playlist.addLagu("Pelangi", "Efek Rumah Kaca", 2007, 4, 45,
"Indie");
        playlist.addLagu("Biru", "Efek Rumah Kaca", 2007, 3, 50,
"Indie");
        playlist.addLagu("Stasiun Balapan", "Didi Kempot", 1999, 4,
42, "Campursari");
        playlist.addLagu("Mungkin Nanti", "Peterpan", 2004, 4, 3,
"Pop Rock");
        playlist.addLagu("Cinta Luar Biasa", "Andmesh Kamaleng",
2019, 4, 23, "Pop");
        playlist.addLagu("Separuh Aku", "Noah", 2012, 4, 20, "Pop
Rock");
        playlist.addLagu("Adu Rayu", "Yovie & Nuno ft. Tulus, Glenn
Fredly", 2019, 4, 18, "Pop");
        playlist.addLagu("Menunggumu", "Peterpan ft. Chrisye", 2004,
4, 3, "Pop Rock");
        playlist.addLagu("Berharap Tak Berpisah", "Reza Artamevia",
2002, 4, 25, "Pop");
        playlist.addLagu("Kisah Kasih di Sekolah", "Chrisye", 2002,
4, 11, "Pop");
        playlist.addLagu("Jangan Menyerah", "D'Masiv", 2009, 4, 39,
"Pop");
        playlist.addLagu("Bento", "Iwan Fals", 1989, 4, 36, "Rock");
        playlist.addLagu("Risalah Hati", "Dewa 19", 2000, 5, 3,
"Rock");
        playlist.addLagu("Juwita Malam", "Slank", 2007, 3, 40, "Pop
Rock");
        playlist.addLagu("Laskar Pelangi", "Nidji", 2008, 5, 7, "Pop
Rock");
        playlist.addLagu("Meraih Bintang", "Via Vallen", 2018, 4, 8,
"Dangdut");
        playlist.addLagu("Bidadari Tak Bersayap", "Anji", 2018, 4,
23, "Pop");
        playlist.addLagu("Cinta Pertama dan Terakhir", "Sherina",
2009, 4, 13, "Pop");
        playlist.addLagu("To the Bone", "Pamungkas", 2020, 5, 12,
"Indie Pop");
        playlist.addLagu("Dia", "Anji", 2016, 3, 37, "Pop");
        playlist.addLagu("Pupus", "Dewa 19", 2000, 5, 6, "Rock");
        playlist.addLagu("A Whole New World", "Zayn Malik & Zhavia
Ward", 2019, 4, 2, "Soundtrack");
        playlist.addLagu("Rewrite the Stars", "Zac Efron & Zendaya",
2018, 3, 37, "Soundtrack");
        playlist.addLagu("Rewrite the Stars", "James Arthur & Anne-
Marie", 2018, 3, 47, "Pop");
        playlist.addLagu("Say You Won't Let Go", "James Arthur",
2016, 3, 31, "Pop");
        playlist.addLagu("Let Her Go", "Passenger", 2012, 4, 12,
"Folk Rock");
        playlist.addLagu("See You Again", "Wiz Khalifa ft. Charlie
Puth", 2015, 3, 49, "Hip Hop");

        playlist.tampilkanPlaylist();
        long startBubbleSort = System.nanoTime();
        playlist.urutkanBerdasarkanDurasi();
        long endBubbleSort = System.nanoTime();
        long startSelectionSort = System.nanoTime();
        playlist.urutkanBerdasarkanRilis();
        long endSelectionSort = System.nanoTime();

```

```
        System.out.println("Waktu Bubble Sort : " + (endBubbleSort -
startBubbleSort) + " nanosekon");
        System.out.println("Waktu Selection Sort : " +
(endSelectionSort - startSelectionSort) + " nanosekon");
    }
}
```

3.3 ANALISIS DATA

3.3.1 Spin Off Rental PS Rijal

```
public class Node {
    String judul;
    String artis;
    int rilis;
    int menit;
    int detik;
    String genre;
    Node next;
    Node prev;

    public Node(String artis, String judul, int rilis, int menit, int
detik, String genre) {
        this.artis = artis;
        this.judul = judul;
        this.rilis = rilis;
        this.menit = menit;
        this.detik = detik;
        this.genre = genre;
        this.next = null;
        this.prev = null;
    }
    public int getDurasiTotal() {
        return menit * 60 + detik;
    }
}
```

Kode di atas berfungsi sebagai elemen dalam *linkedlist* nya yang menyimpan informasi tentang lagu, judul, artis, rilis, durasi menit dan durasi detik, genre, dan referensi ke node *next* dan *prev*.

```
public class EasyAjalah {
    Node head;
    Node tail;

    public void addLagu(String judul, String artis, int rilis, int menit,
int detik, String genre) {
        Node newNode = new Node(judul, artis, rilis, menit, detik, genre);
        if (head == null) {
            head = tail = newNode;
        } else {
            tail.next = newNode;
            newNode.prev = tail;
            tail = newNode;
        }
    }
}
```

Kode di atas adalah metode yang berfungsi sebagai menambahkan lagu baru ke dalam *playlist* nya. Membuat node lagu baru dan menambahkannya ke akhir *list*, jika *playlist* kosong, lagu pertama akan menjadi *head* dan *tail* dari *list*.

```
public void urutkanBerdasarkanDurasi() {
    if (head == null){
        return;
    }
    boolean swapped;
```

```

        do {
            swapped = false;
            Node current = head;
            while (current.next != null) {
                if (current.getDurasiTotal() >
current.next.getDurasiTotal()) {
                    swapData(current, current.next);
                    swapped = true;
                }
                current = current.next;
            }
        } while (swapped);
    }

```

Kode di atas adalah metode mengurutkan playlist berdasarkan durasi lagu secara naik dengan bubble sort. Melakukan looping melalui linkedlist, menukan posisi node jika durasi lagu lebih besar dari durasi lagu berikutnya sehingga seluruh lagu akan terurut berdasarkan durasi terpendek hingga terpanjang.

```

public void urutkanBerdasarkanRilis() {
    if (head == null){
        return;
    }
    Node current = head;
    while (current != null) {
        Node maxNode = current;
        Node nextNode = current.next;
        while (nextNode != null) {
            if (nextNode.rilis > maxNode.rilis) {
                maxNode = nextNode;
            }
            nextNode = nextNode.next;
        }
        if (maxNode != current) {
            swapData(current, maxNode);
        }
        current = current.next;
    }
}

```

Kode di atas merupakan metode mengurutkan playlist berdasarkan tahun rilis lagu secara turun menggunakan selection sort. Mencari node dengan tahun rilis tertinggi di sisa list, dan menukarnya dengan node saat ini jika nilai rilisnya lebih rendah, lalu playlist akan terurut berdasarkan lagu terbaru hingga terlama.

```

private void swapData(Node node1, Node node2) {
    String tempJudul = node1.judul;
    String tempArtis = node1.artis;
    int tempRilis = node1.rilis;
    int tempMenit = node1.menit;
    int tempdetik = node1.detik;
    String tempGenre = node1.genre;

    node1.judul = node2.judul;
    node1.artis = node2.artis;
    node1.rilis = node2.rilis;
    node1.menit = node2.menit;
}

```

```

        node1.detik = node2.detik;
        node1.genre = node2.genre;
        node2.judul = tempJudul;
        node2.artis = tempArtis;
        node2.rilis = tempRilis;
        node2.menit = tempMenit;
        node2.detik = tempdetik;
        node2.genre = tempGenre;
    }

```

Kode di atas merupakan metode menukar data antara dua node dalam linkedlist. Menggunakan variabel sementara untuk menukar atribut judul, artis, rilis, menit, detik, dan genre dari dua node tanpa mengubah posisi node.

```

public Node cariLagu(String judul) {
    Node current = head;
    int pos = 1;
    boolean ditemukan = false;
    System.out.println("=====
=====");
    System.out.println("||                                LOL
Playlist                                ||");
    System.out.println("=====
=====");
    System.out.println("Judul lagu yang ingin dicari : " + judul);

    while (current != null) {
        if (current.judul.equalsIgnoreCase(judul)) {
            System.out.println("Lagu yang sesuai dengan judul '" +
judul + "' ditemukan pada posisi ke - " + pos);
            System.out.println("=====
=====");
            System.out.printf("%7d | %s - %s (%d)%n", pos,
current.artis, current.judul, current.rilis);
            System.out.printf("%7d:%d      %s%n", current.menit,
current.detik, current.genre);
            System.out.println("=====
=====");
            ditemukan = true;
            break;
        }
        current = current.next;
        pos++;
    }
    if (!ditemukan) {
        System.out.println("Lagu dengan judul '" + judul + "' tidak
ditemukan dalam playlist.");
        System.out.println("=====
=====");
    }
    return null;}

```

Kode di atas merupakan metode mencari lagu dalam playlist berdasarkan judulnya. Melakukan pencarian dengan melakukan looping seluruh node dalam playlist, jika lagu dengan judul yang dicari ditemukan, metode ini akan menampilkan informasi lengkap lagu dan posisinya dalam playlist, jika tidak ditemukan maka akan menampilkan pesan bahwa lagu tersebut tidak ada.


```

public void tampilkanPlaylist() {
    Node current = head;
    int pos = 1;
    System.out.println("=====
=====");
    System.out.println("||                                LOL
Playlist                                ||");
    System.out.println("=====
=====");
    if(head==null){
        System.out.println("There is no song in the playlist yet.");
        System.out.println("=====
=====");
    }else{
        while (current != null) {
            System.out.printf("%7d | %s - %s (%d)%n", pos,
current.artis, current.judul, current.rilis);
            System.out.printf("%7d:%d          %s%n", current.menit,
current.detik, current.genre);
            System.out.println("=====
=====");
            current = current.next;
            pos++;
        }
    }
}
}

```

Kode di atas merupakan metode untuk menampilkan seluruh lagunya dalam playlist secara berurutan. Melakukan looping dari head hingga tail linkedlistnya dan mencetak informasi lengkap dari setiap lagu, termasuk posisi dalam playlist, artis, judul, rilis, durasi menit, durasi deti, dan genrenya.

```

public class Main {
    public static void main(String[] args) {
        EasyAjalah playlist = new EasyAjalah();
        playlist.addLagu("Lee Young Ji", "Small girl", 1973, 3, 9, "R&B");
        playlist.addLagu("Why Don't We", "8 Letters", 2024, 3, 10, "Pop");
        playlist.addLagu("The Smiths", "There Is a Light That Never Goes
Out", 2024, 4, 4, "Indie");
        playlist.addLagu("Bruno Mars dan Lady Gaga", "Die With A Smile",
2023, 4, 11, "Pop");
        playlist.addLagu("One Ok Rock", "We are", 2018, 4, 15, "Pop
Rock");
        playlist.addLagu("Wave to Earth", "Bad", 1986, 4, 23, "Korean
Rock, Thai Indie");
        playlist.addLagu(".Feast", "Nina", 2020, 4, 37, "Indonesian
Rock");
        playlist.addLagu("Green Day", "21 Guns", 2017, 5, 21, "Punk
Pop");
        playlist.addLagu("Billy Joel", "Piano Man", 2024, 5, 39, "Folk");
        playlist.addLagu("Pamungkas", "To the Bone", 2009, 5, 44,
"Indonesian Indie");

        playlist.tampilkanPlaylist();

        System.out.println("=====
=====");
    }
}

```

```

        System.out.println("=
Playlist                                LOL
                                =");
        System.out.println("=
Durasi                                Pengurutan Berdasarkan
                                =");
        System.out.println("=====
=====");

        playlist.urutkanBerdasarkanDurasi();
        playlist.tampilkanPlaylist();

        System.out.println("=====
=====");
        System.out.println("=
Playlist                                LOL
                                =");
        System.out.println("=
Rilis                                Pengurutan Berdasarkan Tahun
                                =");
        System.out.println("=====
=====");

        playlist.urutkanBerdasarkanRilis();
        playlist.tampilkanPlaylist();

        playlist.cariLagu("Runaway Baby");

        playlist.addLagu("Cinta Pertama dan Terakhir", "Sherina", 2009,
4, 13, "Pop");
        playlist.addLagu("To the Bone", "Pamungkas", 2020, 5, 12, "Indie
Pop");
        playlist.addLagu("Dia", "Anji", 2016, 3, 37, "Pop");
        playlist.addLagu("Pupus", "Dewa 19", 2000, 5, 6, "Rock");
        playlist.addLagu("A Whole New World", "Zayn Malik & Zhavia Ward",
2019, 4, 2, "Soundtrack");
        playlist.addLagu("Rewrite the Stars", "Zac Efron & Zendaya",
2018, 3, 37, "Soundtrack");
        playlist.addLagu("Rewrite the Stars", "James Arthur & Anne-
Marie", 2018, 3, 47, "Pop");
        playlist.addLagu("Say You Won't Let Go", "James Arthur", 2016,
3, 31, "Pop");
        playlist.addLagu("Let Her Go", "Passenger", 2012, 4, 12, "Folk
Rock");
        playlist.addLagu("See You Again", "Wiz Khalifa ft. Charlie Puth",
2015, 3, 49, "Hip Hop");
        playlist.addLagu("Blinding Lights", "The Weeknd", 2019, 3, 22,
"Pop");
        playlist.addLagu("Shape of You", "Ed Sheeran", 2017, 3, 53,
"Pop");
        playlist.addLagu("Stay", "The Kid LAROI & Justin Bieber", 2021,
2, 21, "Pop");
        playlist.addLagu("Someone Like You", "Adele", 2011, 4, 45, "Pop");
        playlist.addLagu("Levitating", "Dua Lipa", 2020, 3, 23, "Pop");
        playlist.addLagu("Rolling in the Deep", "Adele", 2010, 3, 49,
"Pop");
        playlist.addLagu("Bad Guy", "Billie Eilish", 2019, 3, 14, "Pop");
        playlist.addLagu("Memories", "Maroon 5", 2019, 3, 9, "Pop");
        playlist.addLagu("Counting Stars", "OneRepublic", 2013, 4, 17,
"Pop Rock");
        playlist.addLagu("Perfect", "Ed Sheeran", 2017, 4, 23, "Pop");
        playlist.addLagu("Uptown Funk", "Mark Ronson ft. Bruno Mars",
2014, 4, 30, "Funk Pop");
        playlist.addLagu("Havana", "Camila Cabello", 2018, 3, 37, "Pop");
        playlist.addLagu("Cheap Thrills", "Sia", 2016, 3, 31, "Pop");

```

```

        playlist.addLagu("Closer", "The Chainsmokers ft. Halsey", 2016,
4, 5, "EDM");
        playlist.addLagu("Love Story", "Taylor Swift", 2008, 3, 57,
"Country Pop");
        playlist.addLagu("Numb", "Linkin Park", 2003, 3, 7, "Rock");
        playlist.addLagu("Can't Stop the Feeling!", "Justin Timberlake",
2016, 3, 56, "Pop");
        playlist.addLagu("Believer", "Imagine Dragons", 2017, 3, 24,
"Rock");
        playlist.addLagu("Toxic", "Britney Spears", 2003, 3, 19, "Pop");
        playlist.addLagu("Senorita", "Shawn Mendes & Camila Cabello",
2019, 3, 11, "Pop");
        playlist.addLagu("Photograph", "Ed Sheeran", 2014, 4, 18, "Pop");
        playlist.addLagu("Viva La Vida", "Coldplay", 2008, 4, 2,
"Alternative");
        playlist.addLagu("Rockstar", "Post Malone ft. 21 Savage", 2017,
3, 38, "Hip Hop");
        playlist.addLagu("Thinking Out Loud", "Ed Sheeran", 2014, 4, 41,
"Pop");
        playlist.addLagu("Don't Let Me Down", "The Chainsmokers ft.
Daya", 2016, 3, 28, "EDM");
        playlist.addLagu("Shake It Off", "Taylor Swift", 2014, 3, 39,
"Pop");
        playlist.addLagu("Old Town Road", "Lil Nas X", 2019, 2, 37,
"Country Rap");
        playlist.addLagu("Faded", "Alan Walker", 2015, 3, 32, "EDM");
        playlist.addLagu("Dance Monkey", "Tones and I", 2019, 3, 29,
"Pop");
        playlist.addLagu("All of Me", "John Legend", 2013, 4, 29, "Pop");
        playlist.addLagu("Kangen", "Dewa 19", 1992, 5, 25, "Rock");
        playlist.addLagu("Sampai Jadi Debu", "Banda Neira", 2013, 4, 5,
"Indie");
        playlist.addLagu("Sesuatu di Jogja", "Adhitia Sofyan", 2016, 4,
15, "Folk");
        playlist.addLagu("Kasih Tak Sampai", "Padi", 2001, 4, 52, "Rock");
        playlist.addLagu("Akad", "Payung Teduh", 2017, 4, 4, "Pop");
        playlist.addLagu("Pelangi", "Efek Rumah Kaca", 2007, 4, 45,
"Indie");
        playlist.addLagu("Biru", "Efek Rumah Kaca", 2007, 3, 50, "Indie");
        playlist.addLagu("Stasiun Balapan", "Didi Kempot", 1999, 4, 42,
"Campursari");
        playlist.addLagu("Mungkin Nanti", "Peterpan", 2004, 4, 3, "Pop
Rock");
        playlist.addLagu("Cinta Luar Biasa", "Andmesh Kamaleng", 2019,
4, 23, "Pop");
        playlist.addLagu("Separuh Aku", "Noah", 2012, 4, 20, "Pop Rock");
        playlist.addLagu("Adu Rayu", "Yovie & Nuno ft. Tulus, Glenn
Fredly", 2019, 4, 18, "Pop");
        playlist.addLagu("Menunggumu", "Peterpan ft. Chrisye", 2004, 4,
3, "Pop Rock");
        playlist.addLagu("Berharap Tak Berpisah", "Reza Artamevia", 2002,
4, 25, "Pop");
        playlist.addLagu("Kisah Kasih di Sekolah", "Chrisye", 2002, 4,
11, "Pop");
        playlist.addLagu("Jangan Menyerah", "D'Masiv", 2009, 4, 39,
"Pop");
        playlist.addLagu("Bento", "Iwan Fals", 1989, 4, 36, "Rock");
        playlist.addLagu("Risalah Hati", "Dewa 19", 2000, 5, 3, "Rock");
        playlist.addLagu("Juwita Malam", "Slank", 2007, 3, 40, "Pop
Rock");

```

```

        playlist.addLagu("Laskar Pelangi", "Nidji", 2008, 5, 7, "Pop
Rock");
        playlist.addLagu("Meraih Bintang", "Via Vallen", 2018, 4, 8,
"Dangdut");
        playlist.addLagu("Bidadari Tak Bersayap", "Anji", 2018, 4, 23,
"Pop");
        playlist.addLagu("Cinta Pertama dan Terakhir", "Sherina", 2009,
4, 13, "Pop");
        playlist.addLagu("To the Bone", "Pamungkas", 2020, 5, 12, "Indie
Pop");
        playlist.addLagu("Dia", "Anji", 2016, 3, 37, "Pop");
        playlist.addLagu("Pupus", "Dewa 19", 2000, 5, 6, "Rock");
        playlist.addLagu("A Whole New World", "Zayn Malik & Zhavia Ward",
2019, 4, 2, "Soundtrack");
        playlist.addLagu("Rewrite the Stars", "Zac Efron & Zendaya",
2018, 3, 37, "Soundtrack");
        playlist.addLagu("Rewrite the Stars", "James Arthur & Anne-
Marie", 2018, 3, 47, "Pop");
        playlist.addLagu("Say You Won't Let Go", "James Arthur", 2016,
3, 31, "Pop");
        playlist.addLagu("Let Her Go", "Passenger", 2012, 4, 12, "Folk
Rock");
        playlist.addLagu("See You Again", "Wiz Khalifa ft. Charlie Puth",
2015, 3, 49, "Hip Hop");

        playlist.tampilkanPlaylist();
        long startBubbleSort = System.nanoTime();
        playlist.urutkanBerdasarkanDurasi();
        long endBubbleSort = System.nanoTime();
        long startSelectionSort = System.nanoTime();
        playlist.urutkanBerdasarkanRilis();
        long endSelectionSort = System.nanoTime();
        System.out.println("Waktu Bubble Sort : " + (endBubbleSort -
startBubbleSort) + " nanosekon");
        System.out.println("Waktu Selection Sort : " + (endSelectionSort
- startSelectionSort) + " nanosekon");
    }
}

```

Kode di atas berfungsi sebagai inti atau tempat eksekusi utama dari pemrograman. Menambahkan lagu ke playlist, dan menjalankan operasi playlist, seperti menampilkan, mengurutkan berdasarkan durasi, mengurutkan tahun rilis, dan mencari lagu dalam playlist. Menghitung dan mencetak waktu eksekusi untuk bubble sort dan selection sort dengan fungsi `nanotime`. Hasil ini digunakan untuk membandingkan kecepatan kedua algoritma pengurutan yang digunakan dalam program.

3.3.2 Analisis Readme

Dalam percobaan yang dilakukan pada jurnal ini, saya menggunakan dua algoritma sorting, yaitu Bubble Sort dan Selection Sort. Berdasarkan hasil percobaan, terlihat bahwa Selection Sort secara konsisten membutuhkan waktu yang lebih cepat dibandingkan Bubble Sort. Hal ini dapat terjadi karena program yang saya buat memiliki dataset yang besar, di

mana Selection Sort lebih efisien karena jumlah penukaran yang dilakukan lebih sedikit dibandingkan Bubble Sort.

Berikut merupakan hasil percobaan:

- Waktu yang dibutuhkan oleh selection sort: 424600 nanosekon.
- Waktu yang dibutuhkan oleh bubble sort: 58700 nanosekon.

Bubble Sort memiliki kompleksitas waktu ($O(n^2)$) yang disebabkan oleh cara algoritma ini bekerja dengan membandingkan elemen satu per satu dan memindahkan elemen yang lebih besar atau lebih kecil ke posisi yang tepat secara berulang. Pada dataset besar, seperti yang digunakan dalam percobaan ini, banyaknya perbandingan dan penukaran yang terjadi membuat waktu eksekusi menjadi lebih lama. Sementara itu, meskipun Selection Sort memiliki kompleksitas waktu yang sama ($O(n^2)$), algoritma ini lebih efisien karena hanya melakukan satu penukaran di akhir setiap iterasi setelah menemukan elemen terkecil atau terbesar. Dengan lebih sedikit penukaran, Selection Sort dapat menyelesaikan proses penyortiran lebih cepat, terutama pada dataset besar. Perbedaan efisiensi antara kedua algoritma ini sangat terlihat pada dataset yang lebih besar, dan hasil pengujian menunjukkan bahwa Selection Sort lebih efektif daripada Bubble Sort dalam hal kecepatan eksekusi, karena jumlah penukaran yang lebih sedikit membuatnya lebih cepat dalam menangani data dalam jumlah besar.