LAPORAN PRAKTIKUM

STRUKTUR DATA



DISUSUN OLEH :

Fathur Ramadhan Fahmi

2311533012

DOSEN PENGAMPU :

Dr. Wahyudi, S.T.M.T

PROGRAM STUDI INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI

UNIVERSITAS ANDALAS

2024

1. TUJUAN
2. Mahasiswa mampu memahami konsep ShellSort, MergeSort, dan QuickSort
3. Mahasiswa mampu membuat kodingan dengan menggunakan ShellSort, MergeSort, dan QuickSort
4. PEMBAHASAN

Algoritma sorting dapat diterapkan pada semua bahasa pemrograman. Bahasa pemrograman yang akan digunakan pada penelitian ini adalah menggunakan bahasa pemrograman java.

Algoritma Sorting merupakan salah satu tugas mendasar yang dilakukan oleh komputer. Pengurutan data yang dilakukan Algoritma Sorting memegang fungsi penting agar sebuah masalah dapat diselesaikan lebih cepat dan tepat. Proses pengurutan data banyak digunakan pada proses-proses umum seperti mengurutkan data tanggal dari yang terbaru ke yang lama ataupun sebaliknya. Terdapat banyak Algoritma Sortingyang dapat digunakan, diantaranya Shell Sort, Quick Sort dan Merge Sort.

1. ShellSort

ShellSort adalah pengurutan yang dilakukan di tempat. Shell sort bisa di generalisasikan dengan Bubble sort yang mengurutkan dengan pertukaran, atau Insertion sort yang mengurutkan dengan melakukan penambahan. Langkahnya dimulai dengan mengurutkan 2 elemen paling jauh dari semuanya, lalu mengecilkan jarak dari keduanya. Dimulai dengan elemen-elemen yang berjauhan, shell sort bisa menyortir elemen yang berada diluar untuk menuju posisi seharusnya dengan lebih cepat dan simpel daripada Bubble sort.

Shellsort adalah generalisasi dari insertion sort yang bisa menukar tempat elemen yang berjauhan. Initinya adalah untuk mengurutkan elemen dimulai dari mana saja. Disebut juga dengan metoda pertambahan menurun (diminishing increment). Pada langkah pertama, ambil elemen pertama dan kita bandingkan dengan elemen pada jarak terjauh dari elemen pertama tersebut. Kemudian elemen kedua dibandingkan dengan elemen lain dengan jarak yang sama. Demikian seterusnya sampai seluruh elemen terbandingkan.

1. MergeSort

Algoritma Merge Sort mengambil keuntungan dari kemudahan menggabungkan daftar yang sudah diurutkan ke dalam daftar urutan yang baru. Dimulai dengan proses membandingkan setiap dua elemen (misalnya, 1 dengan 2, 3 dengan 4, dan seterusnya) dan menukarnya jika nilai pertama lebih besar. Kemudian menggabungkan masing-masing dua elemen tersebut menjadi empat elemen dan seterusnya (N. Wirth 1985).

Kelebihan Algoritma Merge Sort

* Stabilitas
* Efisiensi pada Data Besar
* Penggunaan Memori
* Kasus Terburuk yang Konsisten
* Pengurutan Linked List

Kekurangan Algoritma Merge Sort

* Penggunaan Memori Tambahan

Algoritma ini memerlukan ruang memori tambahan yang sebanding dengan ukuran data yang akan diurutkan.

* Kompleksitas Penggabungan

Tahap penggabungan (merge) pada Algoritma Merge Sort memerlukan lebih banyak operasi pembandingan dan penyalinan elemen.

* Pengurutan Data Terkini

Merge Sort merupakan algoritma pengurutan non-inplace, artinya ia menghasilkan urutan data baru tanpa mengubah data asli.

* Tidak Optimal untuk Data Kecil

Meskipun Merge Sort memiliki kinerja yang sangat baik pada data dengan ukuran besar, algoritma ini mungkin tidak optimal untuk data yang relatif kecil.

* Tidak Stabil pada Beberapa Implementasi

Walaupun Merge Sort secara teori adalah algoritma pengurutan stabil, namun implementasinya dapat mempengaruhi stabilitasnya.

1. QuickSort

Quick Sort adalah salah satu jenis algoritma pengurutan dengan konsep divide and conquerdengan mengandalkan operasi partisi, untuk mempartisi sebuah elemen array yang dipilih yang disebut sebagai pivot (Cormen, et al. 2009).

Kelebihan Algoritma Quick Sort :

* Kecepatan: Algoritma Quick Sort memiliki kompleksitas waktu rata-rata O(n log n), yang menjadikannya salah satu algoritma pengurutan yang paling cepat. Ia efisien untuk data dengan jumlah elemen besar.
* Efisiensi Penggunaan Memori: Quick Sort adalah algoritma in-place, sehingga mengurutkan data secara langsung pada tempatnya tanpa memerlukan alokasi memori tambahan untuk operasi pengurutan.
* Penggunaan Rekursi yang Efisien: Quick Sort menggunakan rekursi dengan cara yang efisien, dan stack rekursi yang digunakan cenderung lebih kecil dibandingkan dengan beberapa algoritma pengurutan rekursif lainnya.
* Stabilitas: Quick Sort merupakan algoritma pengurutan yang stabil, yang berarti elemen-elemen dengan nilai yang sama akan tetap dalam urutan relatif mereka setelah pengurutan.

Kekurangan Algoritma Quick Sort

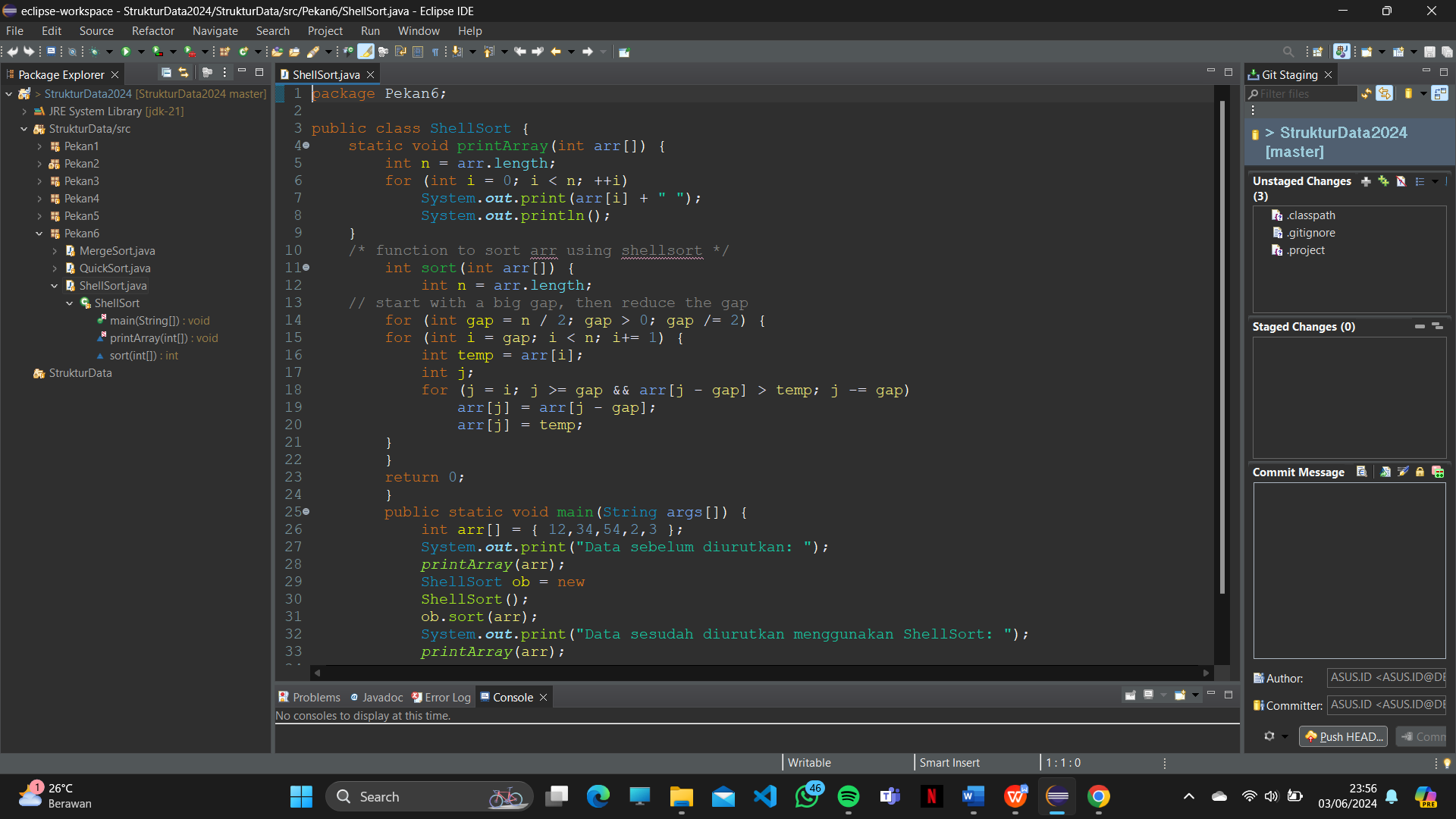
* Kinerja Terburuk: Quick Sort dapat memiliki kinerja terburuk dalam beberapa kasus ketika data hampir terurut atau memiliki banyak elemen yang sama. Dalam situasi ini, kompleksitas waktu bisa mencapai O(n^2).
* Ketergantungan Pivot: Pemilihan pivot yang tidak tepat dapat mempengaruhi performa algoritma. Jika pivot dipilih dengan buruk, ini dapat menyebabkan pengurutan yang tidak seimbang dan mengurangi kecepatan algoritma.
* Tidak Stabil pada Implementasi Non-In-Place: Jika diimplementasikan dalam bentuk non-in-place (menggunakan ruang tambahan untuk pengurutan), Quick Sort tidak lagi menjadi algoritma stabil.

Cara Kerja Algoritma Quick Sort

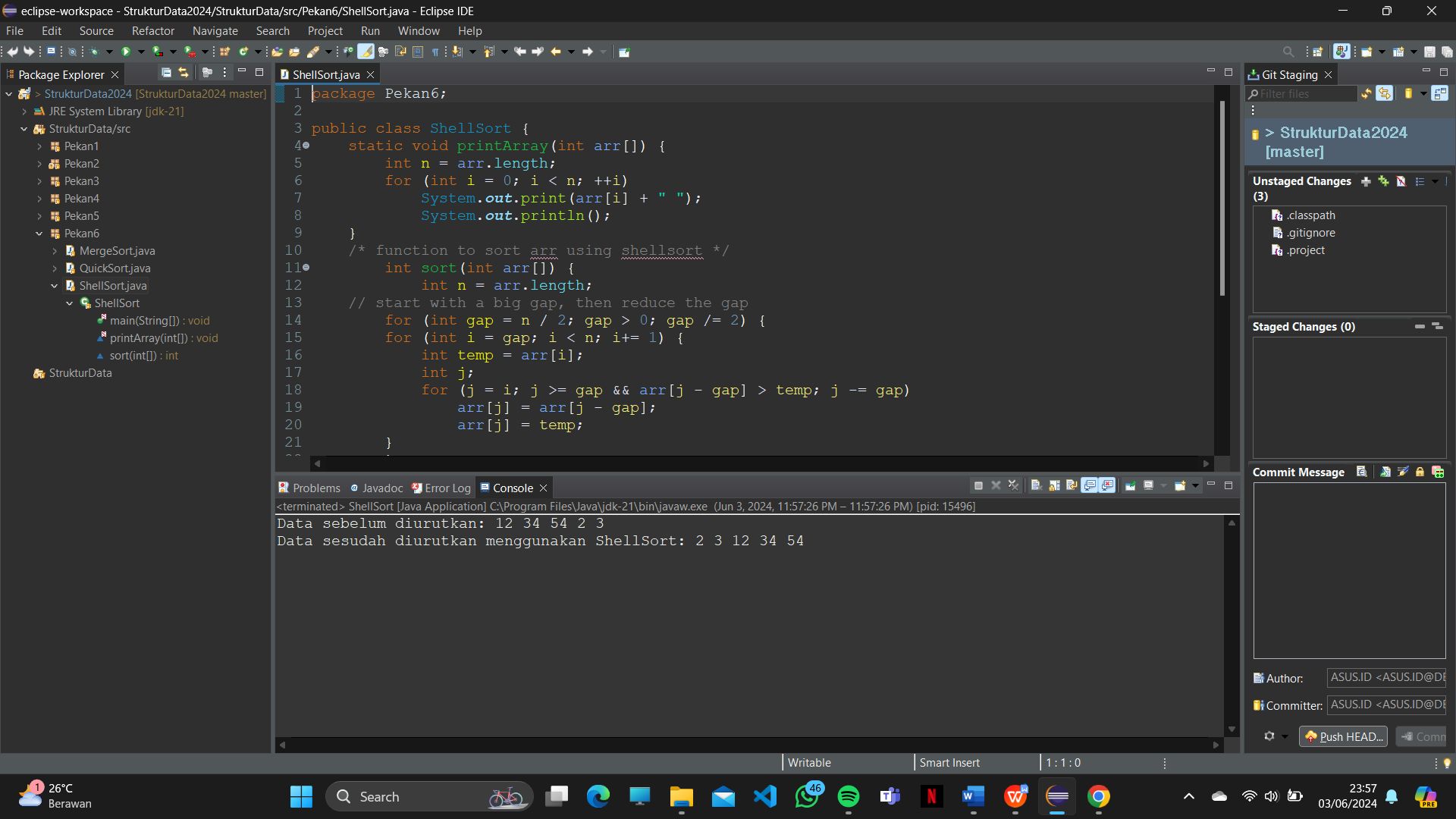
* Pemilihan Pivot: Langkah pertama adalah memilih elemen pivot dari data yang akan diurutkan. Pivot bisa dipilih dari elemen pertama, terakhir, atau elemen tengah.
* Pembagian Data: Data akan dibagi menjadi dua bagian, yaitu elemen yang lebih kecil dari pivot dan elemen yang lebih besar dari pivot. Elemen-elemen ini ditempatkan di sisi kiri dan kanan pivot.
* Rekursi: Proses pembagian dilakukan secara rekursif pada dua bagian data yang lebih kecil hingga mencapai basis rekursif, yaitu ketika hanya ada satu elemen dalam setiap bagian.
* Penggabungan: Setelah seluruh bagian data terurut, mereka digabungkan kembali menjadi satu keseluruhan data yang sudah terurut secara benar.

1. LANGKAH PRAKTIKUM
2. Buka Eclipse IDE
3. Klik New lalu buat Package Pekan6
4. Buat Class baru dan kodingan dibawah berikut ini
5. ShellSort

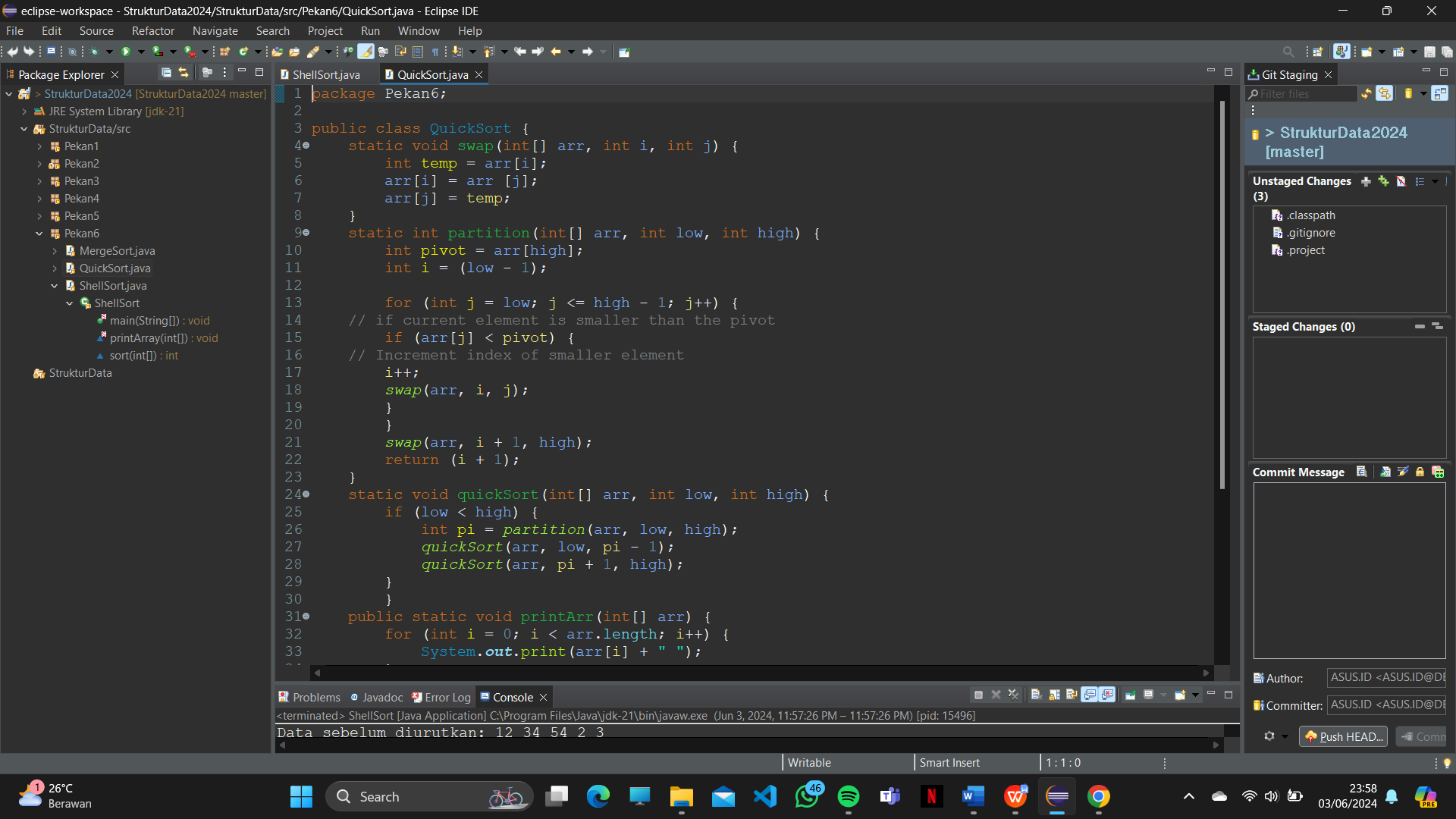
Input :

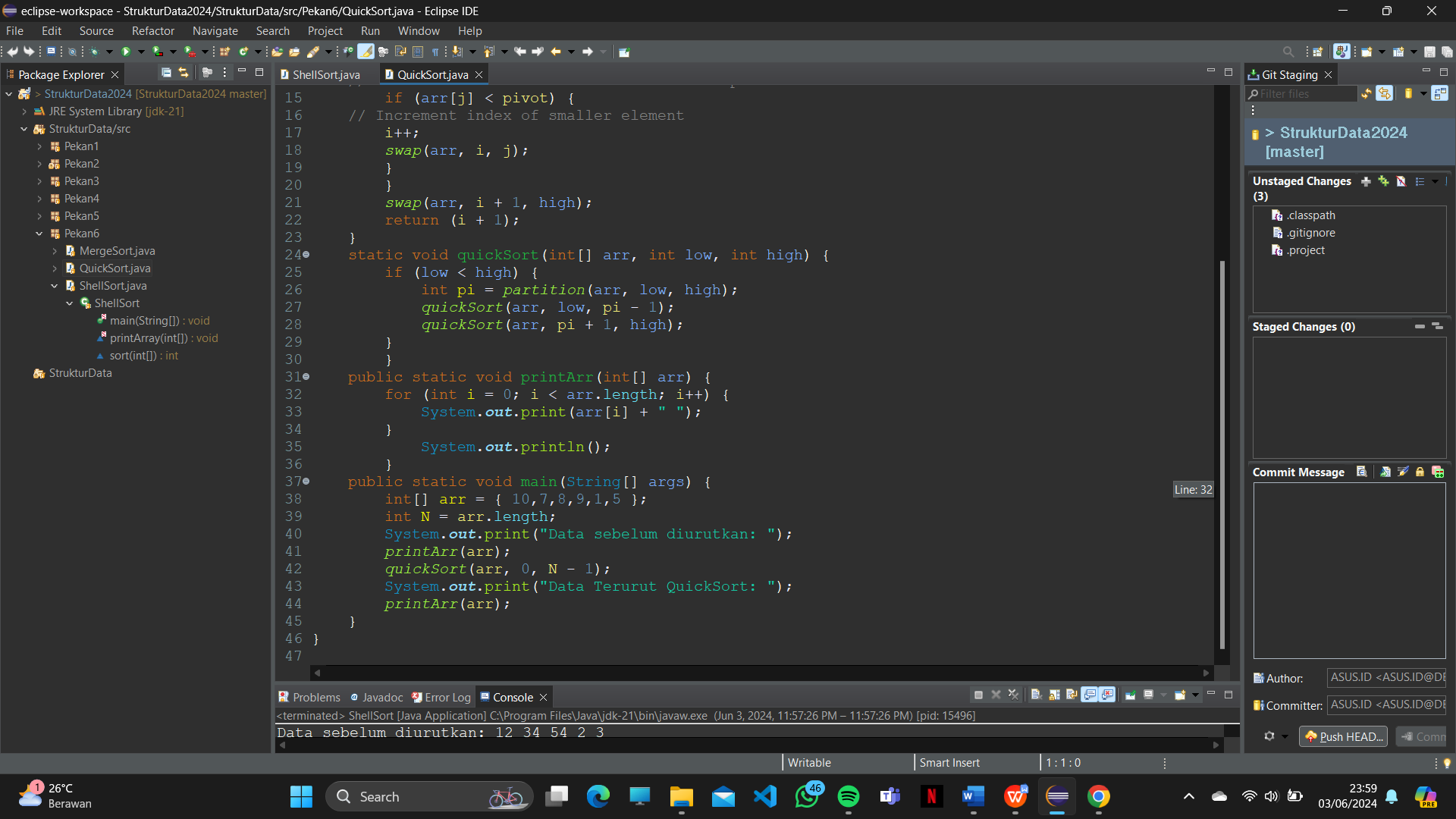


Output :

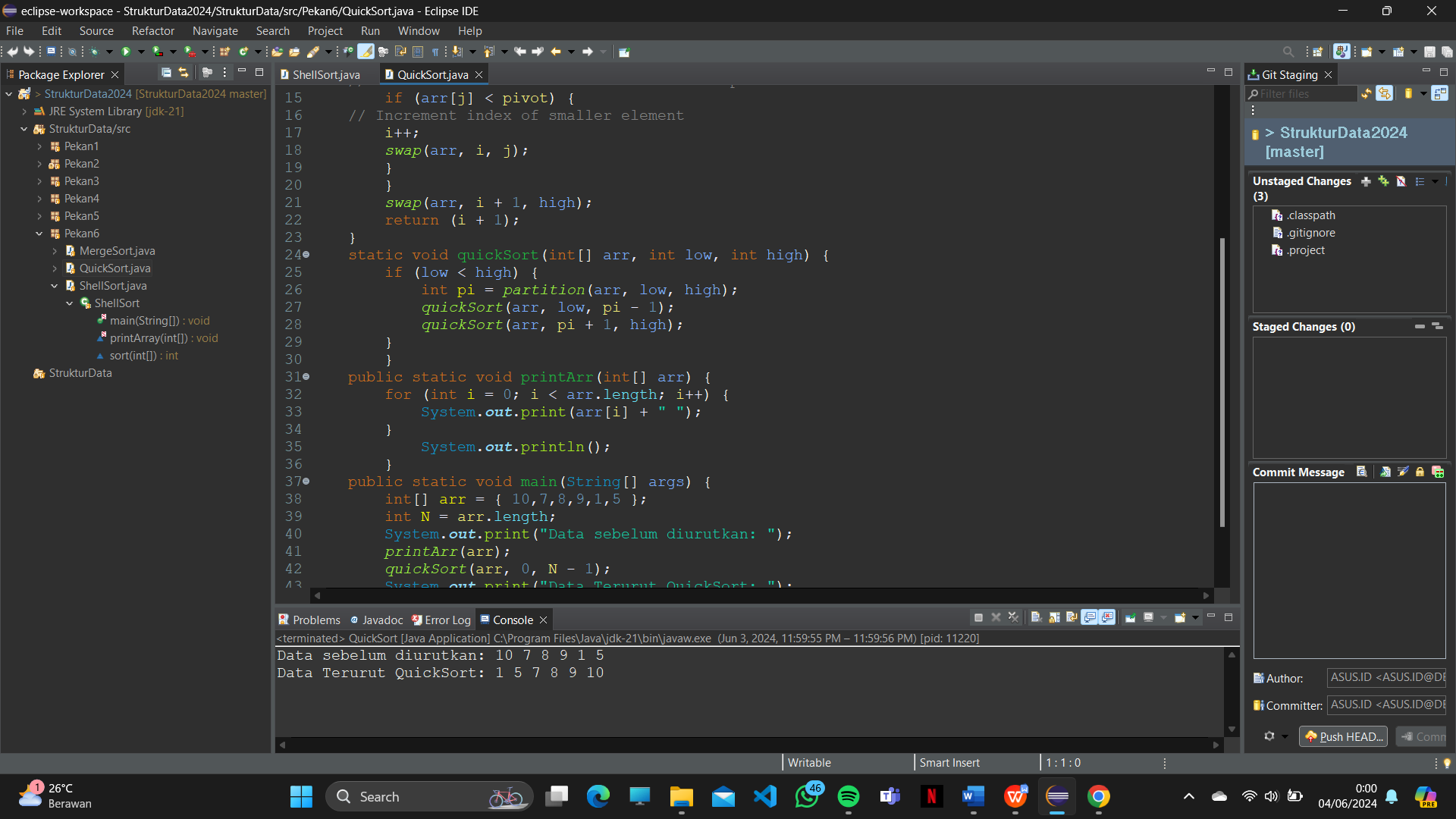


1. QuickSort

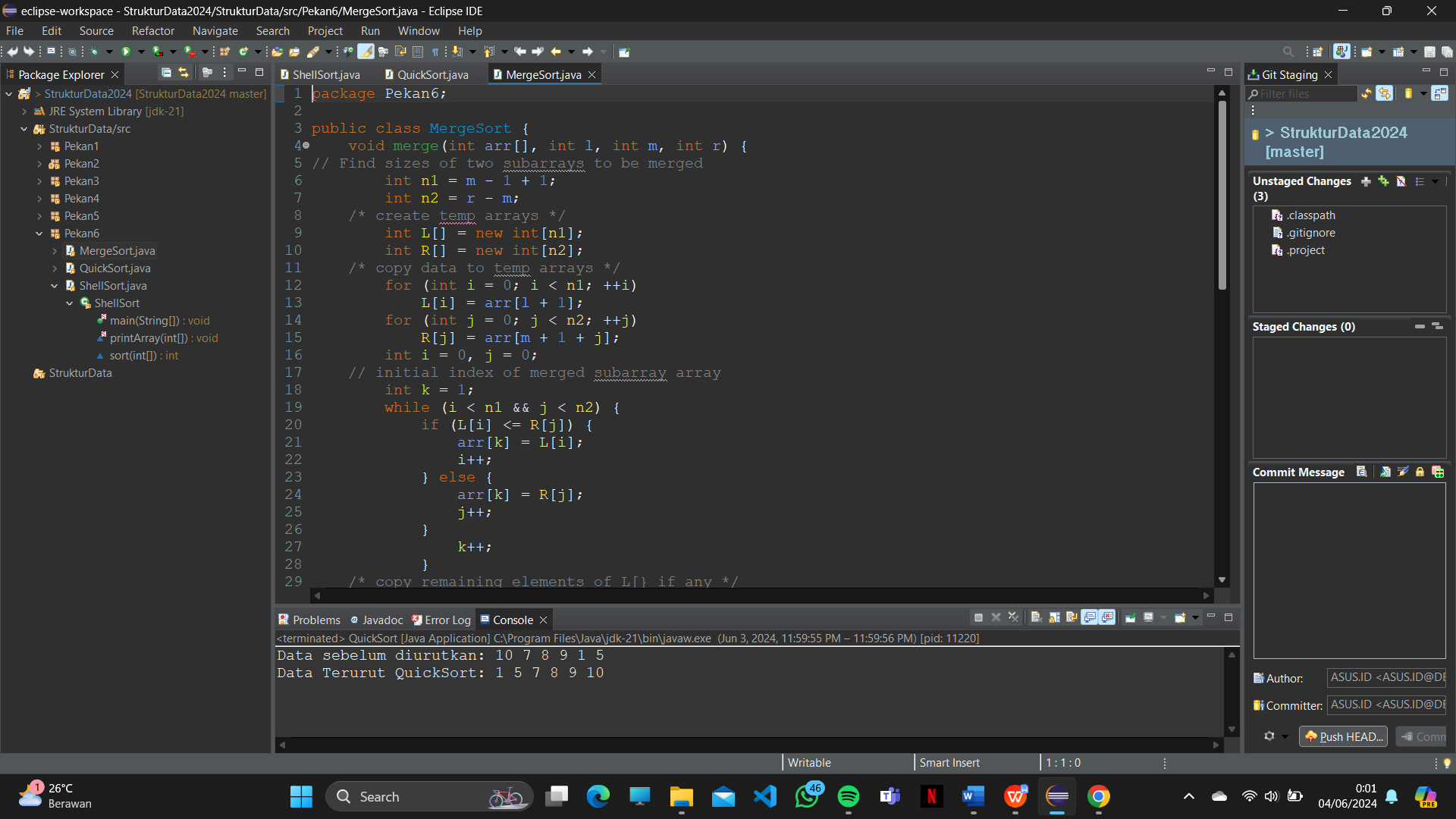


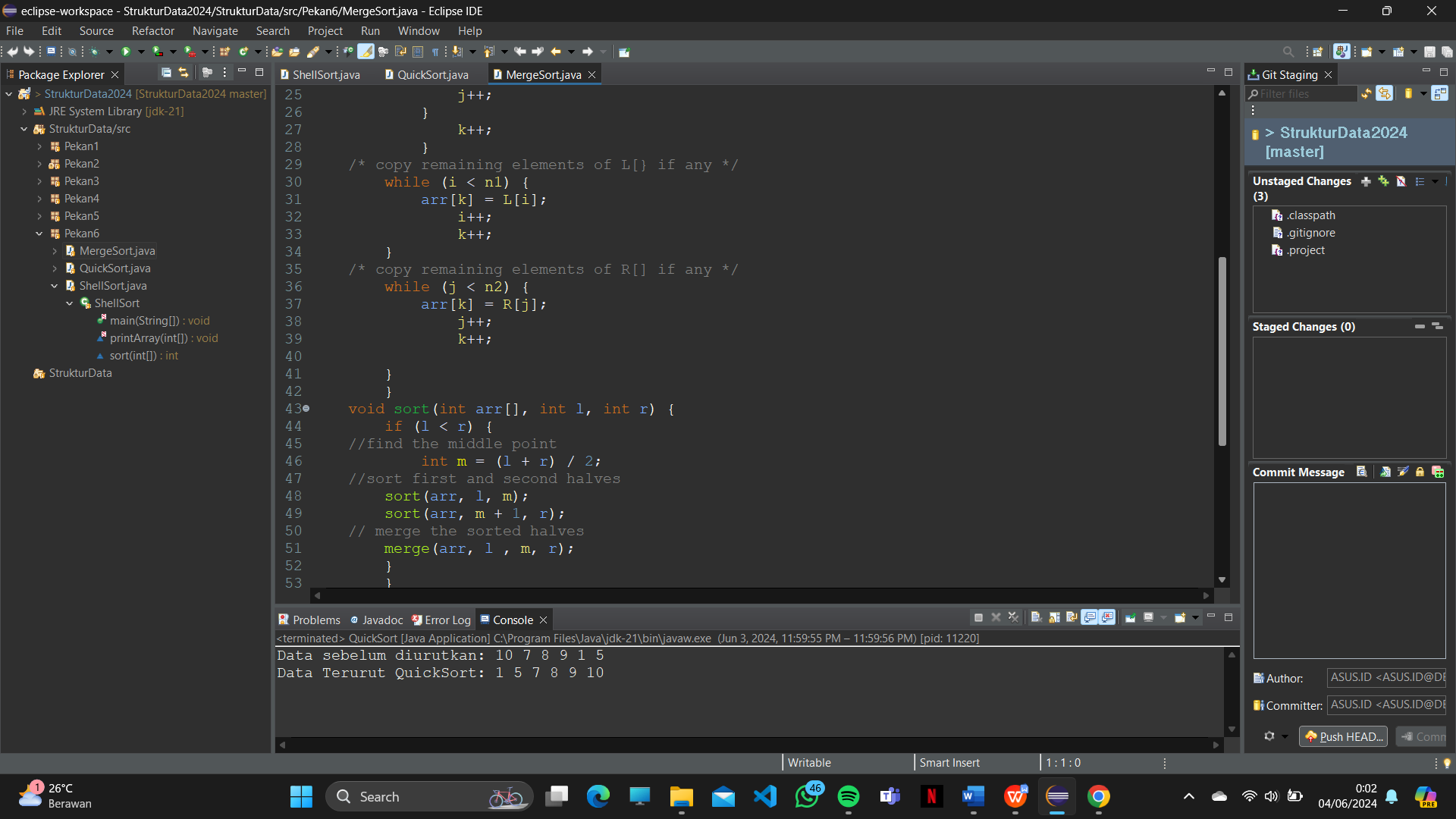


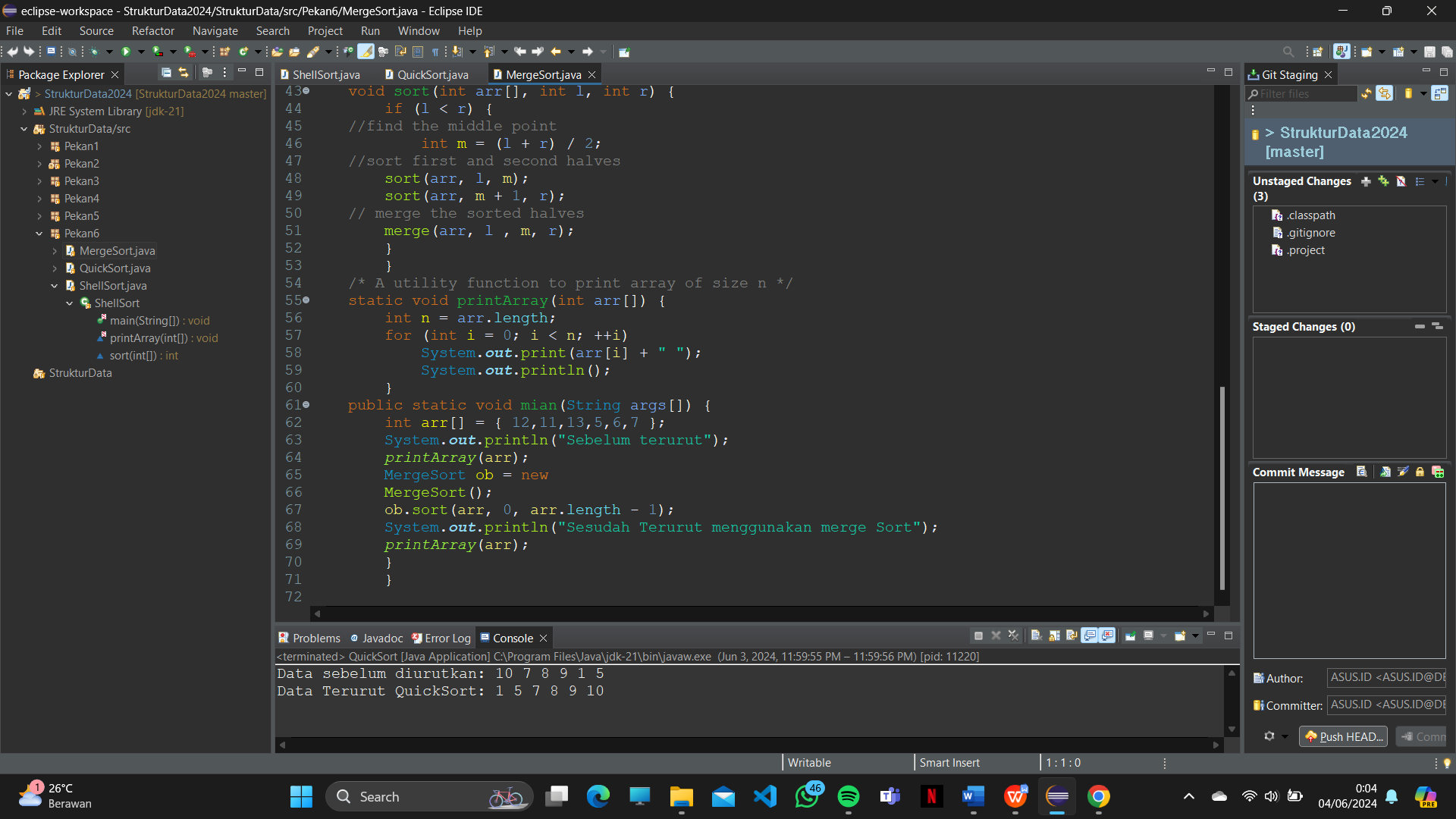
Output :



1. MergeSort







1. KESIMPULAN

Algoritma sorting dapat diterapkan pada semua bahasa pemrograman. Bahasa pemrograman yang akan digunakan pada penelitian ini adalah menggunakan bahasa pemrograman java.

Algoritma Sorting merupakan salah satu tugas mendasar yang dilakukan oleh komputer. Pengurutan data yang dilakukan Algoritma Sorting memegang fungsi penting agar sebuah masalah dapat diselesaikan lebih cepat dan tepat. Proses pengurutan data banyak digunakan pada proses-proses umum seperti mengurutkan data tanggal dari yang terbaru ke yang lama ataupun sebaliknya. Terdapat banyak Algoritma Sortingyang dapat digunakan, diantaranya Shell Sort, Quick Sort dan Merge Sort.