Laporan Praktikum Struktur Data



Disusun Oleh :

**Ridho Radya Pratama (2311533020)**

Dosen Pengampu : Wahyudi , M.T

Departemen Informatika

# Fakultas Teknologi Informasi

Universitas Andalas

Tahun 2024

1. **Pendahuluan**

Dalam dunia komputasi, struktur data memegang peranan penting dalam pengelolaan dan pengolahan informasi. Salah satu struktur data yang sering digunakan adalah Queue atau antrian. Queue adalah struktur data linear yang mengikuti prinsip First In First Out (FIFO), yang berarti elemen pertama yang dimasukkan ke dalam antrian adalah elemen pertama yang akan keluar. Konsep ini mirip dengan antrian di kehidupan nyata, seperti antrian di kasir supermarket atau antrian kendaraan di lampu lalu lintas.

Queue digunakan dalam berbagai aplikasi dan sistem komputasi, mulai dari manajemen tugas dalam sistem operasi, penjadwalan tugas dalam CPU, hingga pengelolaan buffer dalam transmisi data. Struktur ini juga ditemukan dalam berbagai algoritma, seperti algoritma pencarian Breadth-First Search (BFS) dalam graf dan algoritma penjadwalan dalam jaringan komputer.

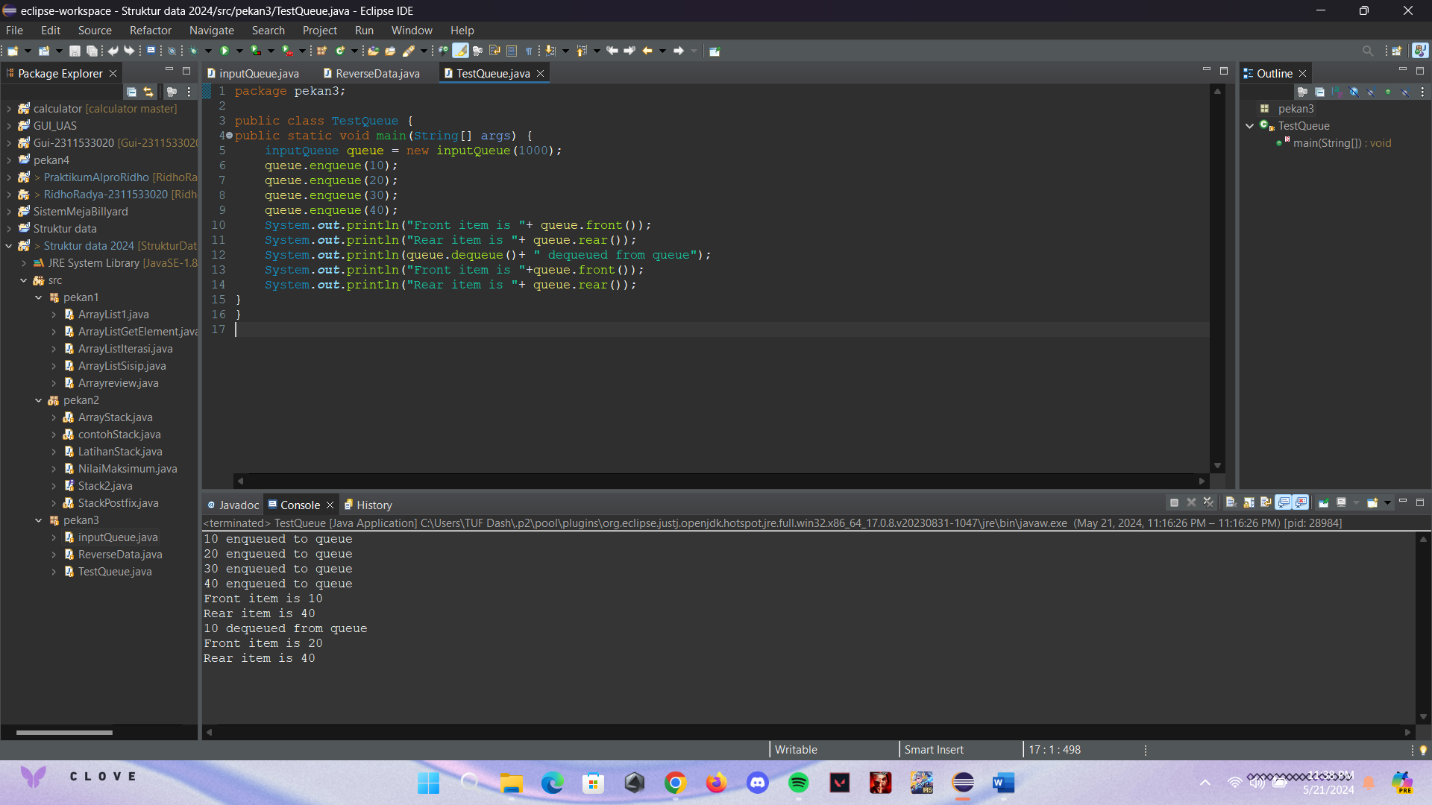
Laporan ini akan membahas secara mendalam tentang konsep, implementasi, dan aplikasi Queue. Pembahasan akan dimulai dengan definisi dan karakteristik dasar Queue, dilanjutkan dengan berbagai metode implementasi, seperti menggunakan array dan linked list. Selain itu, laporan ini akan mengeksplorasi beberapa variasi dari Queue, seperti Circular Queue, Priority Queue, dan Deque (Double-Ended Queue). Setiap jenis Queue memiliki karakteristik dan penggunaan yang spesifik, yang akan dijelaskan secara rinci.

Melalui laporan ini, diharapkan pembaca dapat memahami pentingnya struktur data Queue dalam dunia komputasi, serta mampu mengimplementasikan dan memanfaatkan Queue dalam berbagai konteks pemrograman dan sistem informasi.

1. **LANGKAH-LANGKAH**

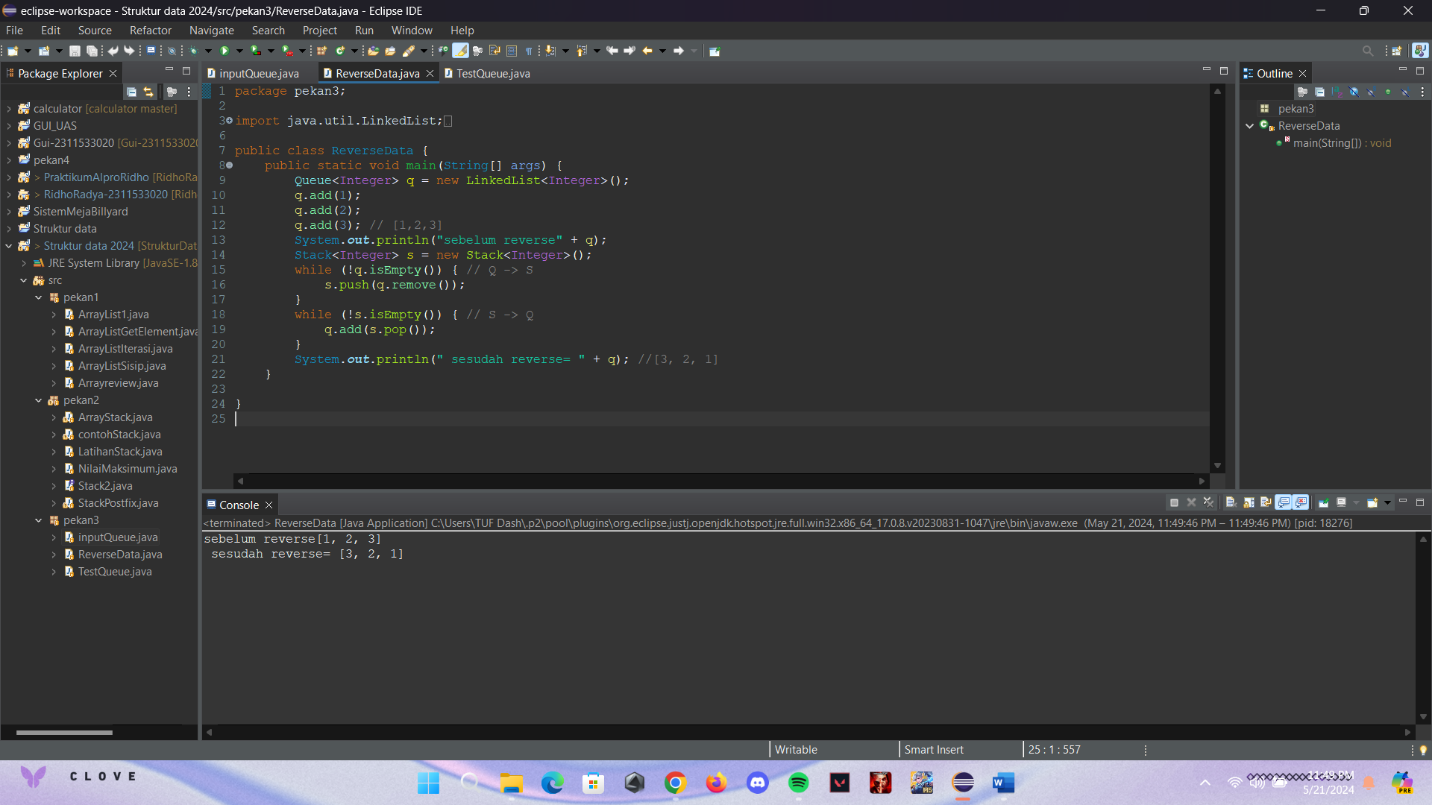
1.Kelas TestQueue

Kelas TestQueue berfungsi untuk menguji fungsionalitas kelas TestQueue. Di dalam kelas ini, kita membuat objek TestQueue dan melakukan operasi enqueue (menambahkan elemen ke antrian) dan dequeue (menghapus elemen dari antrian).



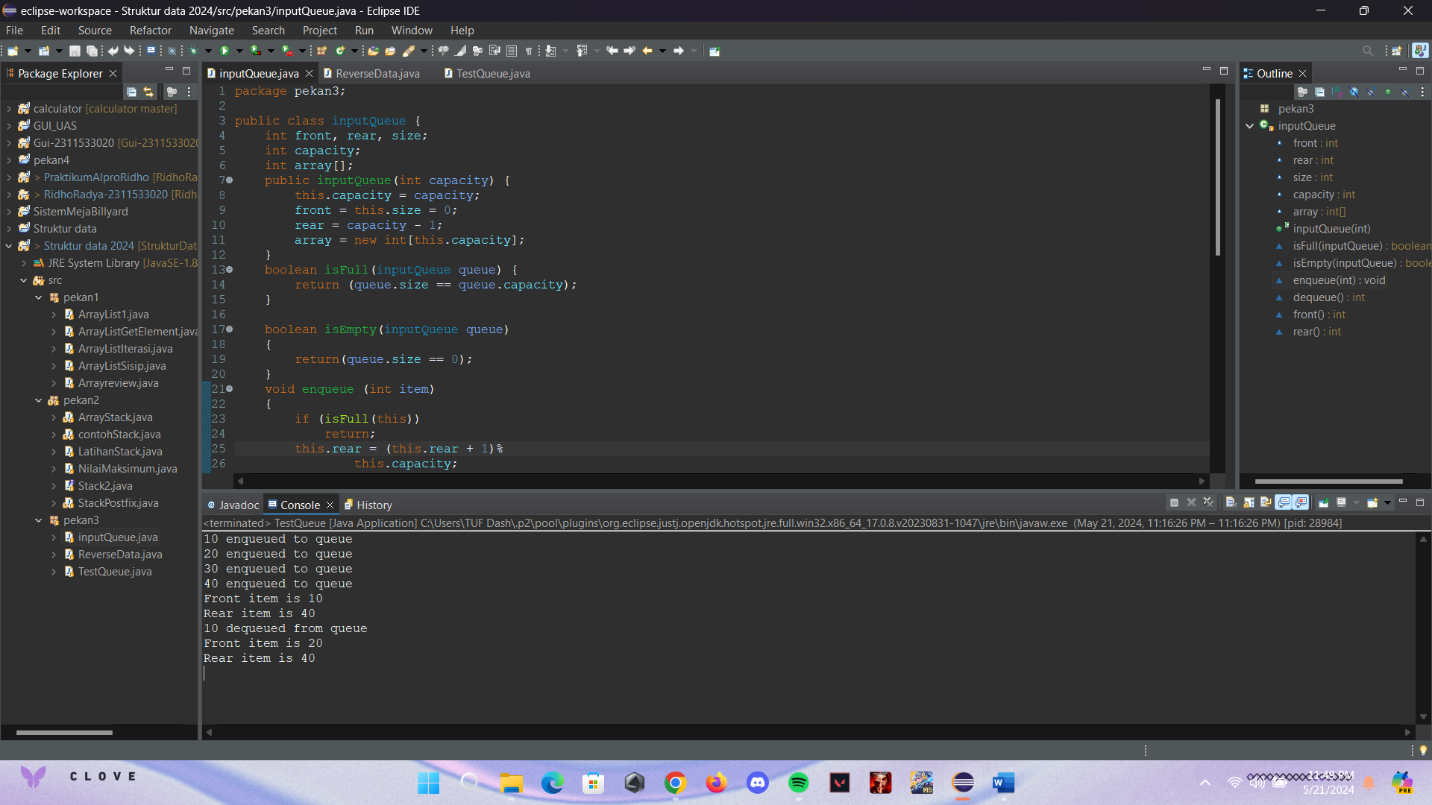
1. Kelas ReverseData

Kelas ini berfungsi untuk membalik urutan elemen dalam sebuah Queue. Proses ini dilakukan dengan memindahkan elemen-elemen dari Queue ke Stack, lalu dari Stack kembali ke Queue.



1. Kelas inputQueue

**Kelas ini mendefinisikan struktur data Queue beserta metode-metodenya untuk menambahkan (enqueue) dan menghapus (dequeue) elemen, serta memeriksa status antrian (penuh atau kosong).**



1. **PENUTUP**

Dalam menutup sebuah laporan tentang Queue, penting untuk merangkum inti dari apa yang telah dibahas dan memberikan kesimpulan yang kuat. Berikut ini sebuah penutup yang mungkin bisa Anda gunakan:

Dengan demikian, laporan ini telah membahas secara komprehensif tentang struktur data Queue dan aplikasinya dalam pemrograman komputer. Dari pengertian dasar, operasi-operasi utama, hingga implementasi dalam bahasa pemrograman Java, kita telah menjelajahi berbagai aspek yang terkait dengan Queue.

Queue sebagai struktur data yang mengikuti konsep FIFO (First In First Out) memiliki peran yang sangat penting dalam pengelolaan data terurut, terutama dalam konteks sistem komputasi dan algoritma. Dari contoh-contoh sederhana hingga aplikasi yang lebih kompleks, Queue memberikan solusi efisien dalam pengaturan antrian tugas atau operasi.

Dalam proses pembelajaran ini, kita juga memahami pentingnya pemilihan struktur data yang tepat sesuai dengan kebutuhan dan konteks aplikasi. Pemahaman yang mendalam tentang Queue juga membantu dalam memperluas wawasan terkait dengan struktur data lainnya seperti Stack, LinkedList, dan sebagainya.

Terakhir, diharapkan bahwa laporan ini dapat menjadi panduan yang bermanfaat bagi pembaca dalam memahami konsep dasar Queue, mengimplementasikannya dalam kode program, serta mengaplikasikannya secara efektif dalam proyek-proyek pengembangan perangkat lunak di masa depan.

Dengan demikian, mari kita terus eksplorasi dan memperdalam pemahaman terkait dengan struktur data dan algoritma, sehingga dapat menghasilkan solusi-solusi yang inovatif dan efisien dalam dunia pemrograman dan teknologi informasi secara keseluruhan.