**LAPORAN PRAKTIKUM**

**Algoritma Dan Struktur Data**

**Jobsheet - 8 : Queue**



|  |  |
| --- | --- |
| Nama | : Ghoffar Abdul Ja’far |
| NIM | : 41720035 |
| Kelas | : 1E |

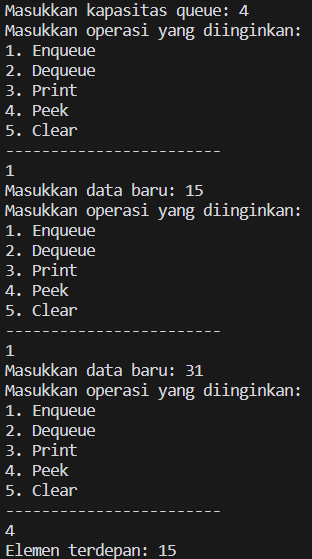
**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**2023/2024**

**Praktikum 1**

Hasil:



**Pernyataan**

1. Pada konstruktor, mengapa nilai awal atribut front dan rear bernilai -1, sementara atribut size bernilai 0?

= Karena front dan rear bernilai -1 yang mana front dan rear tidak menujuk nilai pada queue, maka size bernilai 0.

1. Pada method Enqueue, jelaskan maksud dan kegunaan dari potongan kode berikut!



= Digunakan saat rear baru ditambahkan dan posisi rear sudah sampai index terakhir yaitu max – 1, maka rear baru akan berada pada index 0.

1. Pada method Dequeue, jelaskan maksud dan kegunaan dari potongan kode berikut!



= Digunakan saat front lama dikeluarkan dan posisi front sudah sampai index terakhir yaitu max – 1, maka front baru akan berada pada index 0.

1. Pada method print, mengapa pada proses perulangan variabel i tidak dimulai dari 0 (int i=0), melainkan int i = front?

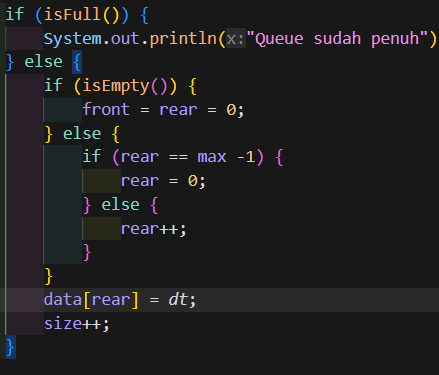
= Karena queue di print dimulai dari front dan front tidak selalu dimulai dari indeks ke 0 maka I perlu diawali dari front.

1. Perhatikan kembali method print, jelaskan maksud dari potongan kode berikut!



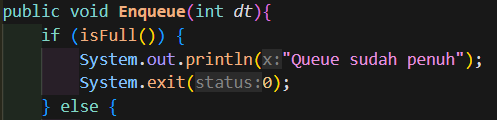
= Digunakan untuk menunjuk front jika front sudah sampai index terakhir maka akan dimodulus sebanyak max agar dapat Kembali ke indeks 0

1. Tunjukkan potongan kode program yang merupakan queue overflow!

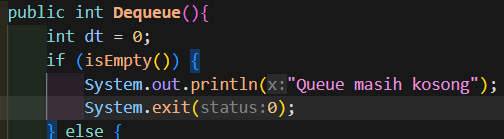


1. Pada saat terjadi queue overflow dan queue underflow, program tersebut tetap dapat berjalan dan hanya menampilkan teks informasi. Lakukan modifikasi program sehingga pada saat terjadi queue overflow dan queue underflow, program dihentikan!

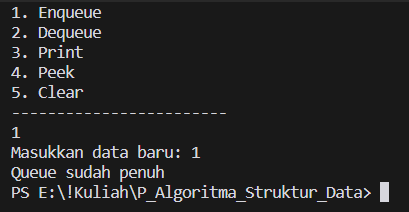
Enqueue:



Dequeue:

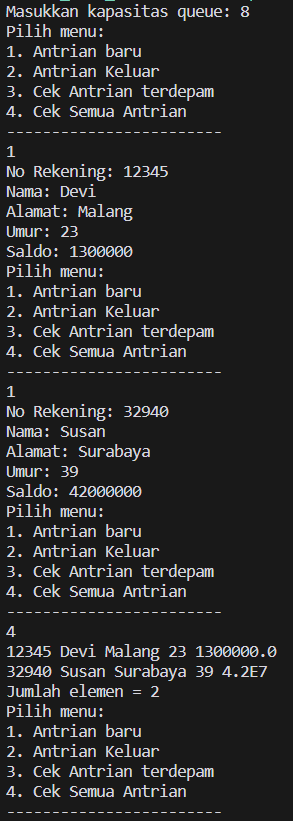


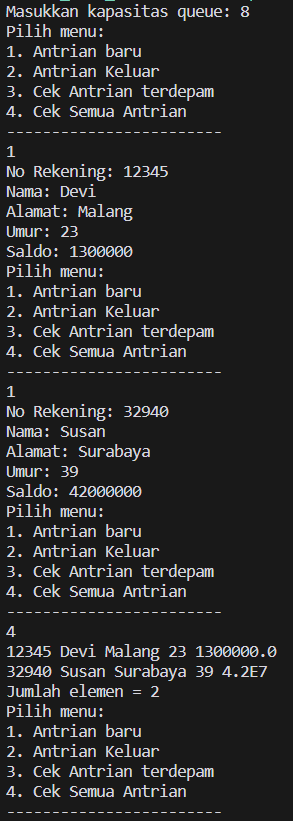
Hasil:



**Praktium 2**

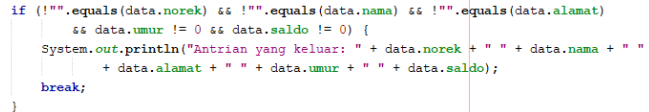
Hasil:





**Pertanyaan**

1. Pada class QueueMain, jelaskan fungsi IF pada potongan kode program berikut!



1. Lakukan modifikasi program dengan menambahkan method baru bernama peekRear pada class Queue yang digunakan untuk mengecek antrian yang berada di posisi belakang! Tambahkan pula daftar menu **5. Cek Antrian paling belakang** pada class **QueueMain** sehingga method **peekRear** dapat dipanggil!