**POLITEKNIK NEGERI MALANG**

**TEKNOLOGI INFORMASI**

**TEKNIK INFORMATIKA**



**Nama: Abdul Rahman Hanif Darmawan**

**NIM: 244107020232**

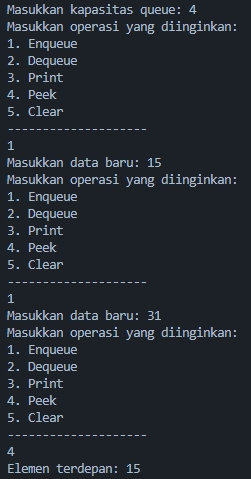
**Kelas: TI-1A**

**Prodi: D4-TEKNIK INFORMATIKA**

# JOBSHEET 10

2.1 Percobaan 1 : Operasi Dasar Queue

2.1.2 Verifikasi Hasil Percobaan



2.1.3 Pertanyaan

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

* Nilai -1 menandakan bahwa queue dalam keadaan kosong sedangkan size=0 menunujukkan bahwa elemen dalam queue adalah nol.

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



* Kode tersebut berfungsi untuk mengimplementasikan queue melingkar jika rear sudah di indeks terakhir array, maka setelahnya akan dilanjutkan kembali ke indeks 0.

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



* Mirip dengan nomor 2, jika front mencapai indeks terakhir (max - 1), maka indeks pengambilan data selanjutnya akan kembali ke 0.

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

* Karena data dalam queue tidak selalu dimulai dari indeks 0 terutama sudah terjadi proses enqueue-dequeue, dan mulai mencetak dari indeks fornt yang menandai elemen pertama dalam queue

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



* Setelah indeks mencapai batas akhir (max - 1), ia akan kembali ke 0, fungsinya agar pencetakan elemen queue bisa menyusuri semua elemen secara sirkuler.

1. Tunjukkan potongan kode program yang merupakan queue overflow!





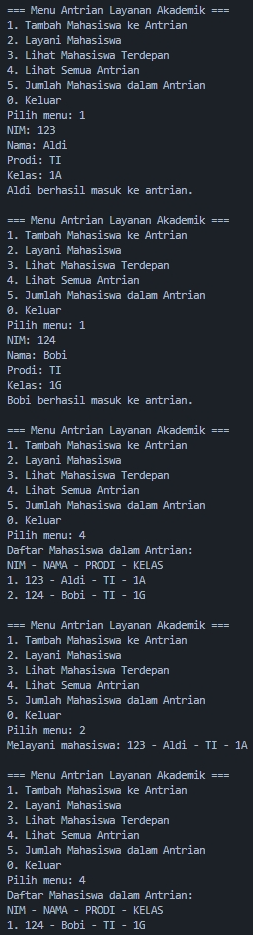
1. Pada saat terjadi queue overflow dan queue underflow, program tersebut tetap dapat berjalan dan hanya menampilkan teks informasi. Lakukan modifikasi program sehingga pada saat terjadi queue overflow dan queue underflow, program dihentikan!

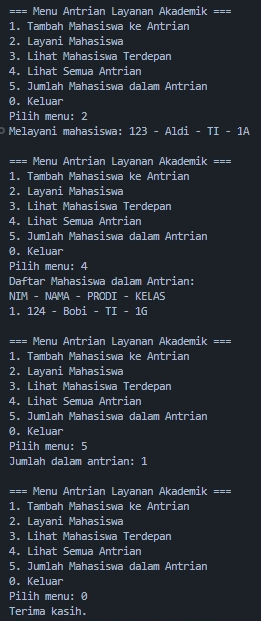




2.2 Percobaan 2 : Antrian Layanan Akademik

2.2.2 Verifikasi Hasil Percobaan





2.2.3 Pertanyaan

1. Lakukan modifikasi program dengan menambahkan method baru bernama LihatAkhir pada class AntrianLayanan yang digunakan untuk mengecek antrian yang berada di posisi belakang. Tambahkan pula daftar menu 6. Cek Antrian paling belakang pada class LayananAkademikSIAKAD sehingga method LihatAkhir dapat dipanggil!









2.3 Tugas

AntrianKRS.java





Mahasiswa.java



MainKRS.java





Output





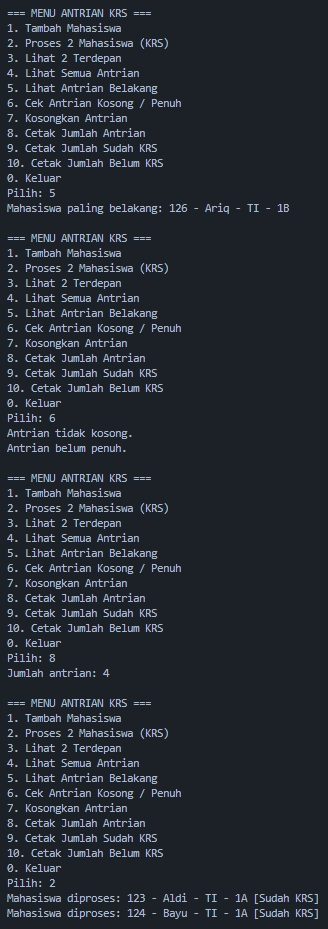




Diagram Class

|  |
| --- |
| Mahasiswa |
| * String nim * String nama * String prodi * String kelas * boolean sudahKRS |
| * tampilkanData() |

|  |
| --- |
| AntrianKRS |
| * Mahasiswa[] data * int front * int rear * int front * int max |
| * isEmpty(), isFull(), tambahAntrian(), prosesKRS() * lihatTerdepan(), lihatDuaTerdepan(), lihatAkhir() * tampilkanSemua(), getJumlahAntrian(), getJumlahSudahKRS(), getJumlahBelumKRS() * kosongkanAntrian() |

Link Github   
<https://github.com/baynobu/ALSD/tree/c5cbfff0e271158a4b2c7a488cbf240a104f98f9/Pertemuan%2011>