**LAPORAN HASIL PRAKTIKUM ALGORITMA DAN STRUKTUR DATA**

**JOBSHEET 10**



MUHAMMAD AMMAR HAFIZH

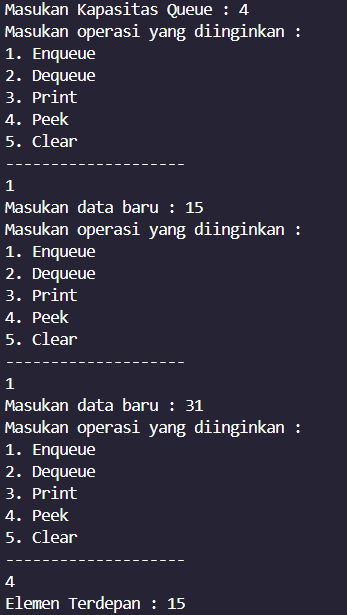
(2341720074)

D-IV TEKNIK INFORMATIKA – 1E

**Jurusan Teknologi Informasi**

**Politeknik Negeri Malang**

**Hasil Praktikum Percobaan 1**

****

**Pertanyaan Percobaan 1**

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

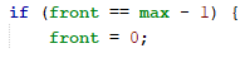
* Front Rear bernilai -1 agar tidak menunjuk atau meposisikan diri ke dalam queue atau antrian yang belum ada isinya dan size berawal dari 0 karena elemen masih kosong belum diisi apapun dan jika sudah terisi maka size akan mengikuti panjangnya antrian dan akan berhenti enqueue saat size sama dengan max.

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



* Untuk mengubah rear pada posisi terdepan dan front posisi terbelakang karena kolom terbelakang atau rear sudah terisi dan kolom depan masih kosong dan nilai yang dimasukan akan masuk pada kolom depan dan dijadikan rear.

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



* Ketika semua antrian sudah dikeluarkan dan sisa baris terbelakang yang ingin dikeluarkan maka front akan direset menjadi paling depan lagi.

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

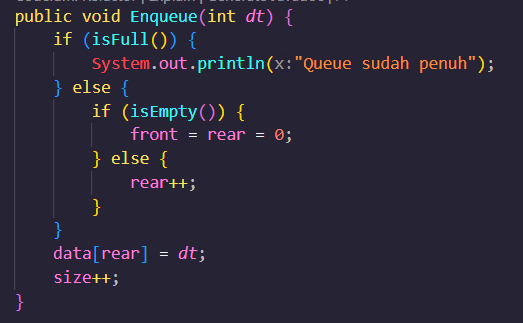
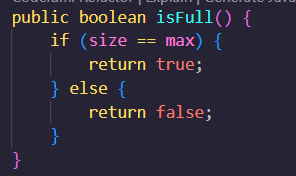
* Karena blm tentu antrian terdepan atau front bernilai indeks 0

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



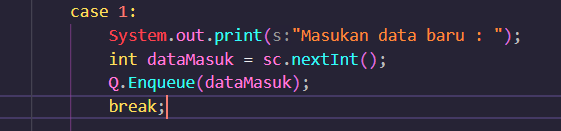
* Untuk melakukan looping dan jika sudah max antrian makan akan di modulus sehingga I menjadi 0 dan looping diberhentikan

6. Tunjukkan potongan kode program yang merupakan queue overflow!

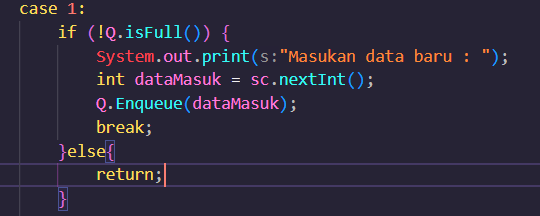


7. Pada saat terjadi queue overflow dan queue underflow, program tersebut tetap dapat berjalan dan hanya menampilkan teks informasi. Lakukan modifikasi program sehingga pada saat terjadi queue overflow dan queue underflow, program dihentikan

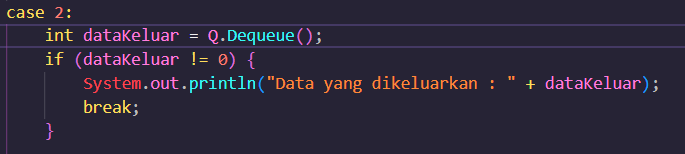
* **Before Overflow**

****

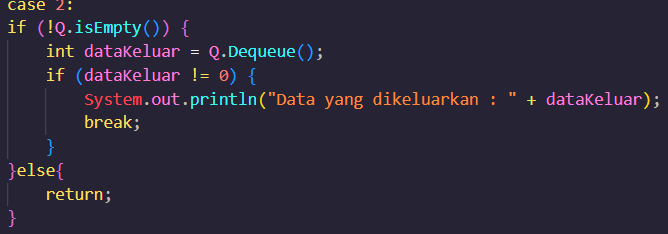
* **After Overflow**

****

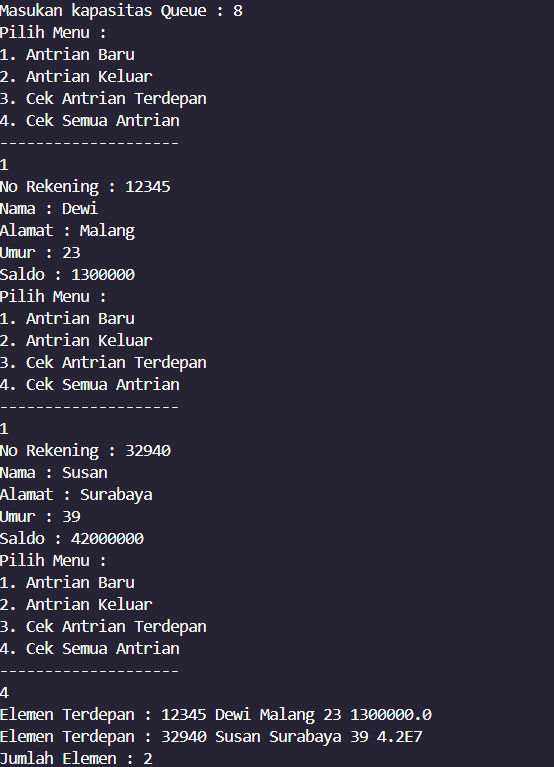
* **Before Underflow**

****

* **After Underflow**

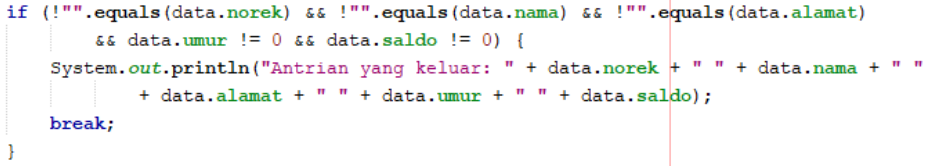
****

**Hasil Praktikum Percobaan 2**

****

**Pertanyaan Percobaan 2**

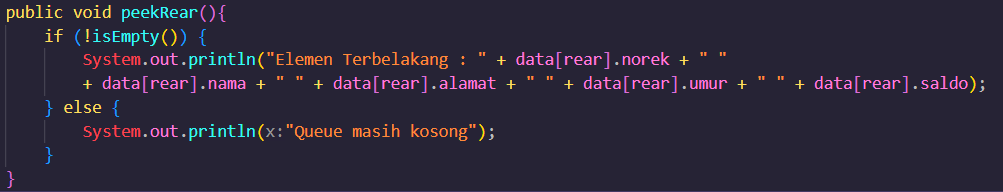
1. Pada class QueueMain, jelaskan fungsi IF pada potongan kode program berikut!



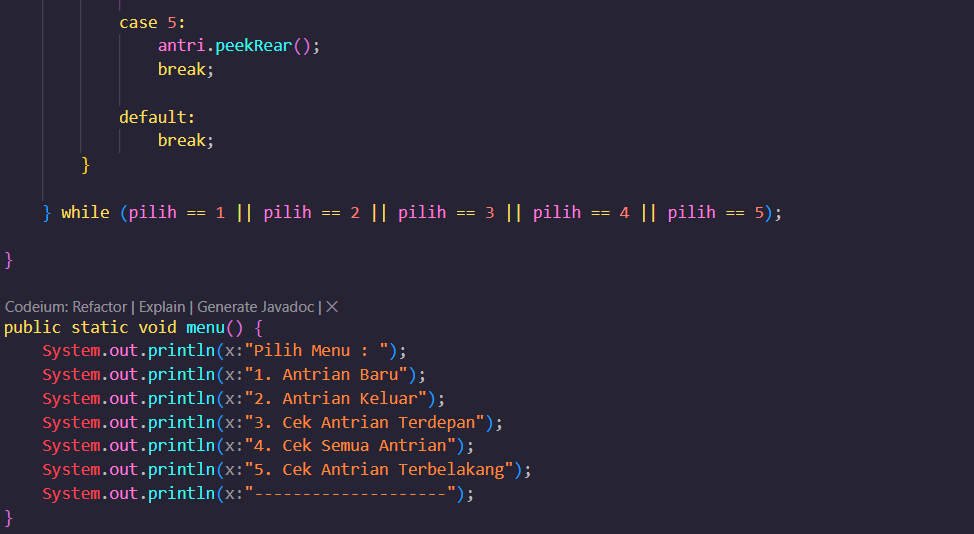
* Code tersebut mencegah terjadinya underflow

2. 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!

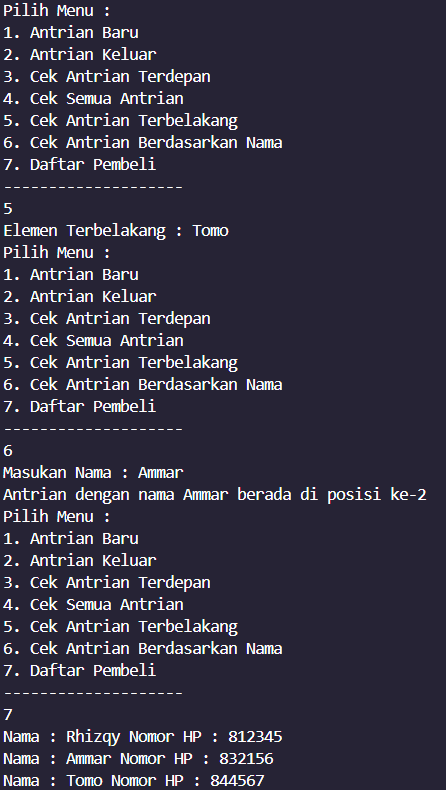
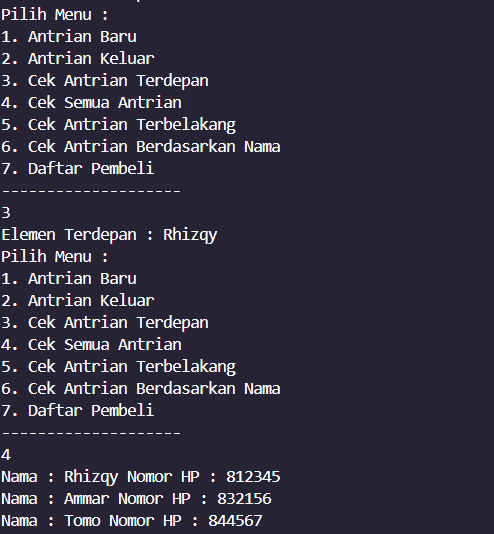
* **Method**

****

* **Main**

****

**Hasil Tugas**

****