

QUEUE

OLEH : ANDRI HERYANDI, M.T.



05

CIRCULAR QUEUE

OLEH : ANDRI HERYANDI, M.T.



DEFINISI CIRCULAR QUEUE

01158 - ALGORITMA DAN STRUKTUR DATA 2

- Circular queue biasanya digunakan pada queue yang menggunakan Array sebagai penyimpan elemen queue-nya.
- Circular queue digunakan untuk “menambal” kelemahan/kekurangan dari queue biasa yang diimplementasikan dengan array.
- Kelemahan dari queue biasa yang “ditambal” oleh circular queue adalah ketika adanya pengambilan data (dequeue). Sebagai contoh : ketika sebuah queue mempunyai elemen sebanyak 100, dan kemudian dilakukan dequeue (ambil pada posisi pertama), maka 99 data berikutnya akan bergeser. Selanjutnya ketika ada dequeue lagi, maka akan terjadi 98 pergeseran data. Tentunya pergeseran data ini membutuhkan beban dan waktu komputasi. Nah, pergeseran ini lah yang akan coba dihilangkan dengan circular queue.



ILUSTRASI CIRCULAR QUEUE

01158 - ALGORITMA DAN STRUKTUR DATA 2

1. Kondisi Awal

Index	0	1	2	3	4(max -1)	Front = -1 Queue masih Kosong.
Isi						Rear = -1

2. Enqueue(7)

Index	0	1	2	3	4(max -1)	Front = 0 Front dan Rear menjadi 0.
Isi	7					Rear = 0

3. Enqueue(9)

Index	0	1	2	3	4(max -1)	Front = 0 Hanya Rear yang ditambah 1
Isi	7	9				Rear = 1



ILUSTRASI CIRCULAR QUEUE

01158 - ALGORITMA DAN STRUKTUR DATA 2

4. Enqueue(15)

Index	0	1	2	3	4(max -1)
Isi	7	9	15		

Front = 0

Hanya Rear yang ditambah 1

Rear = 2

5. Isi = Dequeue()

Index	0	1	2	3	4(max -1)
Isi		9	15		

Front = 1

Elemen Front diambil (Isi bernilai 7), kemudian Front ditambah 1. Data tidak mengalami pergeseran. Hanya nilai Front yang pindah.

Rear = 2

6. Enqueue(3)

Index	0	1	2	3	4(max -1)
Isi		9	15	3	

Front = 1

Hanya Rear yang ditambah 1.

Rear = 3



ILUSTRASI CIRCULAR QUEUE

01158 - ALGORITMA DAN STRUKTUR DATA 2

7. Enqueue(17)

Index	0	1	2	3	4(max -1)
Isi		9	15	3	17

Front = 1

Hanya Rear yang ditambah 1.

Rear = 4

8. Enqueue(25)

Index	0	1	2	3	4(max -1)
Isi	25	9	15	3	17

Front = 1

Rear = 0

Walau pun Rear sudah sama dengan Maximum-1, tapi Enqueue masih boleh karena masih ada slot kosong di depan. Karena Rear = max-1 maka Rear pindah ke 0. Elemen baru disimpa di Rear yang baru.

9. Enqueue(3)

Index	0	1	2	3	4(max -1)
Isi	25	9	15	3	17

Front = 1

Rear = 0

Error, karena queue sudah penuh.



ILUSTRASI CIRCULAR QUEUE

01158 - ALGORITMA DAN STRUKTUR DATA 2

10. Isi = Dequeue()

Index	0	1	2	3	4(max -1)
Isi	25		15	3	17

Front = 2

Rear = 0

Elemen Front diambil (Isi bernilai 9), kemudian Front ditambah 1.

11. Isi = Dequeue()

Index	0	1	2	3	4(max -1)
Isi	25			3	17

Front = 3

Rear = 0

Elemen Front diambil (Isi bernilai 15), kemudian Front ditambah 1.

12. Enqueue(70)

Index	0	1	2	3	4(max -1)
Isi	25	70		3	17

Front = 3

Rear = 1

Hanya Rear ditambah 1



ILUSTRASI CIRCULAR QUEUE

01158 - ALGORITMA DAN STRUKTUR DATA 2

13. Isi = Dequeue()

Index	0	1	2	3	4(max -1)
Isi	25	70			17

Front = 4

Rear = 1

Elemen Front diambil (Isi bernilai 3), kemudian Front ditambah 1.

14. Isi = Dequeue()

Index	0	1	2	3	4(max -1)
Isi	25	70			

Front = 0

Rear = 1

Elemen Front diambil (Isi bernilai 17). Karena Front sudah sama dengan Max-1 maka Front tidak boleh ditambah 1, tapi Front diset menjadi 0.

15. Isi = Dequeue()

Index	0	1	2	3	4(max -1)
Isi		70			

Front = 1

Rear = 1

Elemen Front diambil (Isi bernilai 25), kemudian Front ditambah 1.



ILUSTRASI CIRCULAR QUEUE

01158 - ALGORITMA DAN STRUKTUR DATA 2

13. Isi = Dequeue()

Index	0	1	2	3	4(max -1)
Isi					

Front = 0

Rear = 0

Elemen Front diambil (Isi bernilai 70). Karena elemen hanya tinggal 1 maka nilai Front dan Rear diset dengan nilai -1 (menyatakan bahwa queue kosong)



KONVERSI

QUEUE ARRAY BIASA MENJADI CIRCULAR QUEUE

01158 - ALGORITMA DAN STRUKTUR DATA 2

- Beberapa operasi queue array biasa dapat digunakan dalam circular queue.
 - Operasi **isEmpty** : [sama dengan queue biasa]
 - Operasi **isFull**, operasi ini harus ada perombakan karena pada circular queue, kondisi queue penuh tidak hanya ketika Rear sama dengan Maximum elemen. Perhatikan 2 gambar di bawah ini yang sama-sama menyatakan bahwa queue penuh.

Front : 0, Rear : 4, Max=5

index	0	1	2	3	4
Isi	5	8	2	9	10

Front : 3, Rear : 2, Max=5

index	0	1	2	3	4
Isi	6	5	4	2	7

- Operasi **count**, operasi harus ada perombakan karena nilai Rear tidak selalu lebih besar sama dengan nilai Front. Ada kemungkinan Rear berada di depan Front. Perhatikan gambar di bawah ini.

Front : 0, Rear : 3, Max=5

index	0	1	2	3	4
Isi	5	8	2	9	

Harus menghasilkan 4

Front : 3, Rear : 0, Max=5

index	0	1	2	3	4
Isi	6			2	7

Harus menghasilkan 3



KONVERSI

QUEUE ARRAY BIASA MENJADI CIRCULAR QUEUE

01158 - ALGORITMA DAN STRUKTUR DATA 2

- Operasi **initialize** : [sama dengan queue biasa]
- Operasi **display**, harus ada perombakan karena nilai Rear tidak selalu lebih besar Front, terkadang Front lebih besar dari Rear.

Front : 0, Rear : 3, Max=5

index	0	1	2	3	4
Isi	5	8	2	9	

Harus menampilkan : 5 8 2 9

- Operasi **peek** : [sama saja dengan queue biasa]
- Operasi **empty** : [sama saja dengan queue biasa]

Front : 3, Rear : 0, Max=5

index	0	1	2	3	4
Isi	6			2	7

Harus menampilkan : 2 7 6



KONVERSI

QUEUE ARRAY BIASA MENJADI CIRCULAR QUEUE

01158 - ALGORITMA DAN STRUKTUR DATA 2

- Operasi **enqueue** harus ada perombakan karena nilai Rear tidak selalu bertambah, terkadang nilai Rear dipindahkan ke index 0.

Ketika nilai Rear belum mencapai Maximum-1

Front : 0, Rear : 2, Max=5

index	0	1	2	3	4
Isi	5	8	2		

Kondisi Awal

Enqueue(10)

Front : 0, Rear : 3, Max=5

index	0	1	2	3	4
Isi	5	8	2	10	

Nilai Rear ditambah 1 karena nilai Rear tidak sama dengan Maximum-1

Ketika nilai Rear mencapai Maximum-1

Front : 3, Rear : 4, Max=5

index	0	1	2	3	4
Isi				2	7

Kondisi Awal

Enqueue(10)

Front : 3, Rear : 0, Max=5

index	0	1	2	3	4
Isi	10			2	7

Nilai Rear diset menjadi 0 karena Rear sudah mencapai Maksimum-1.



KONVERSI

QUEUE ARRAY BIASA MENJADI CIRCULAR QUEUE

01158 - ALGORITMA DAN STRUKTUR DATA 2

- Operasi **dequeue** harus ada perombakan karena nilai Front tidak selalu ada di indeks ke-0 tetapi bisa berubah (bertambah 1).

Ketika nilai Front belum mencapai Maximum-1

Front : 0, Rear : 2, Max=5

index	0	1	2	3	4
Isi	5	8	2		

Kondisi Awal

Dequeue()

Front : 1, Rear : 2, Max=5

index	0	1	2	3	4
Isi		8	2		

Nilai Front bertambah 1

Ketika nilai Front mencapai Maximum-1

Front : 4, Rear : 1, Max=5

index	0	1	2	3	4
Isi	5	9			7

Kondisi Awal

Dequeue()

Front : 0, Rear : 1, Max=5

index	0	1	2	3	4
Isi	5	9			

Nilai Front diset menjadi 0 karena Front sudah mencapai Maksimum-1.



LATIHAN

01158 - ALGORITMA DAN STRUKTUR DATA 2

- **Buatlah sebuah sistem antrian untuk pemanggilan nasabah di Bank.**
- **Ketentuan :**
 - Semua nasabah dianggap sama (tidak ada yang prioritas).
 - Bank memiliki 2 loket (Loket 1 dan Loket 2).
 - Menu di aplikasi :
 1. Tambah Antrian
 2. Loket 1 Memanggil
 3. Loket 2 Memanggil
 4. Keluar dari Aplikasi
 - Layar aplikasi akan menampilkan
 - No Antrian yang sedang dilayani oleh masing-masing Loket.
 - No Antrian pemanggilan selanjutnya



LATIHAN

01158 - ALGORITMA DAN STRUKTUR DATA 2

Tampilan Aplikasi

SISTEM ANTRIAN S.A.T. BANK

Loket 1

X

Loket 2

Y

No Berikutnya

Z

Menu Pilihan :

1. Tambah Antrian
2. Loket 1 Memanggil
3. Loket 2 Memanggil
4. Keluar

Aturan :

1. Jika dipilih Menu 1, maka antrian bertambah (enqueue). Update nomor antrian berikutnya.
2. Jika dipilih Menu 2, maka ambil antrian terdepan (dequeue) oleh Loket 1. Tampilkan nomor antrian di Loket 1 dan update nomor antrian berikutnya.
3. Jika dipilih Menu 3, maka ambil antrian terdepan (dequeue) oleh Loket 2. Tampilkan nomor antrian di Loket 2 dan update nomor antrian berikutnya.
4. Jika memilih Menu 4 maka keluar dari aplikasi.



LATIHAN (NEXT DEVELOPMENT)

01158 - ALGORITMA DAN STRUKTUR DATA 2

- Setelah anda mengerjakan latihan sebelumnya (antrian di Bank), sekarang saatnya agar aplikasi anda bisa mengeluarkan suara pemanggilan antrian.
- Contoh Kasus : Kalau ada Loker Nomor 2 melakukan panggilan Nomor Antrian 035 maka akan membunyikan speaker komputer dengan kata-kata “Nomor Antrian Nol Tiga Lima Ke Loker Dua”.
- Persiapan file suara.
 - Siapkan file wav untuk suara angka “0”, “1”, “2”, “3” s.d “9”. Disimpan dengan nama file sesuai yang angka yang dibunyikan 0.wav, 1.wav, s.d 9.wav. Boleh merekam atau mencari di internet.
 - Siapkan file wav untuk suara “Nomor Antrian” (nama file “nomor-antrian.wav”) dan “Ke Loker” dengan nama file “ke-loker.wav”. Boleh merekam atau mencari di internet.
- Pemanggilan ketika “Loker Nomor 2 melakukan panggilan Nomor Antrian 035” maka perangkat lunak akan melakukan :
 - Playsound nomor-antrian.wav
 - Playsound 0.wav
 - Playsound 3.wav
 - Playsound 5.wav
 - Playsound ke-loker.wav
 - Playsound 2.wav

Untuk membunyikan suara, silahkan explore “Playsound with Python”.



PENUTUP

01158 - ALGORITMA DAN STRUKTUR DATA 2

SEKIAN

Ada pertanyaan?

Silahkan tanyakan melalui Group Whatsapp, email, LMS atau comment video Youtube.



FORUM DISKUSI

01158 - ALGORITMA DAN STRUKTUR DATA 2



LMS UNIKOM

<https://lms.unikom.ac.id>



**Group Whatsapp
Perkuliahahan**



Youtube Comments

<https://unikom.id/YTCStrukturData>



Oleh : Andri Heryandi, M.T.