

Pertemuan 9

QUEUE (ANTREAN)



PENGERTIAN QUEUE (ANTREAN)

Struktur Data Antrean (Queue) adalah suatu bentuk khusus dari List Linier dengan operasi pemasukan data hanya diperbolehkan pada salah satu sisi, yang disebut sisi Belakang / ekor (Tail) dan operasi penghapusan hanya diperbolehkan pada sisi lainnya yang disebut sisi Depan / kepala (Head) dari LinkedList.

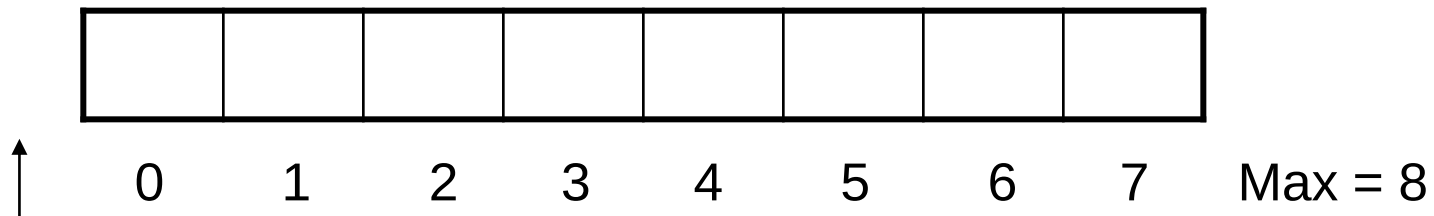
Prinsip Antrean : FIFO (First In First Out)
FCFS (First Come First Serve)

“Yang Tiba lebih awal Maka akan dilayani Terlebih Dahulu”

Deklarasi Queue

```
#define MAX 8
typedef struct{
    int data[MAX];
    int head;
    int tail;
} Queue;

Queue antrian;
```



head = -1

tail = -1

OPERASI QUEUE

- **CREATE**

Untuk menciptakan dan menginisialisasi Queue
Dengan cara membuat Head dan Tail = -1

- **ISEMPTY**

Untuk memeriksa apakah queue kosong

- **ISFULL**

Untuk memeriksa apakah queue sudah penuh

- **ENQUEUE**

Untuk menambahkan item pada posisi paling belakang

- **DEQUEUE**

Untuk menghapus item dari posisi paling depan

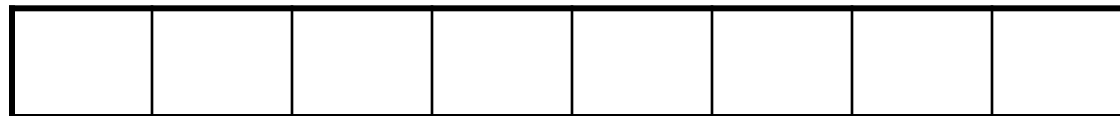
- **CLEAR**

Untuk mengosongkan queue

Fungsi Create

Digunakan untuk membentuk dan menunjukan awal terbentuknya suatu Antrean / Queue

```
Void Create()
{
    antrian.head = antrian.tail = -1
}
```



↑
0 1 2 3 4 5 6 7 Max = 8

head = -1

tail = -1

Antrian pertama kali

Fungsi IsEmpty

- Untuk memeriksa apakah Antrian penuh atau kosong
- Dengan cara memeriksa nilai Tail, jika $\text{Tail} = -1$ maka antrian kosong (empty)
- Head adalah tanda untuk kepala antrian (elemen pertama dalam antrian) yang tidak akan berubah-ubah
- Pergerakan pada Antrian terjadi dengan penambahan elemen Antrian kebelakang, yaitu menggunakan nilai Tail

Fungsi IsEmpty

```
Int IsEmpty()  
{  
    if (antrian.tail == -1)  
        return 1;  
    else  
        return 0;  
}
```



↑ 0 1 2 3 4 5 6 7 Max = 8

head = -1

tail = -1

Antrian kosong

Karena tail = -1

Fungsi IsFull

Untuk mengecek apakah Antrian sudah penuh atau belum

Dengan cara :

- Mengecek nilai Tail
- Jika $\text{tail} = \text{MAX}-1$ berarti antrian sudah penuh (MAX-1 adalah batas elemen array dalam program C++)

Fungsi IsFull

```
Int IsFull()
{
    if (antrian.tail == Max-1)
        return 1;
    else
        return 0;
}
```

5	10	35	20	15	30	40	25
---	----	----	----	----	----	----	----

0 1 2 3 4 5 6 7 Max = 8

↑

head = 0

Antrian penuh karena

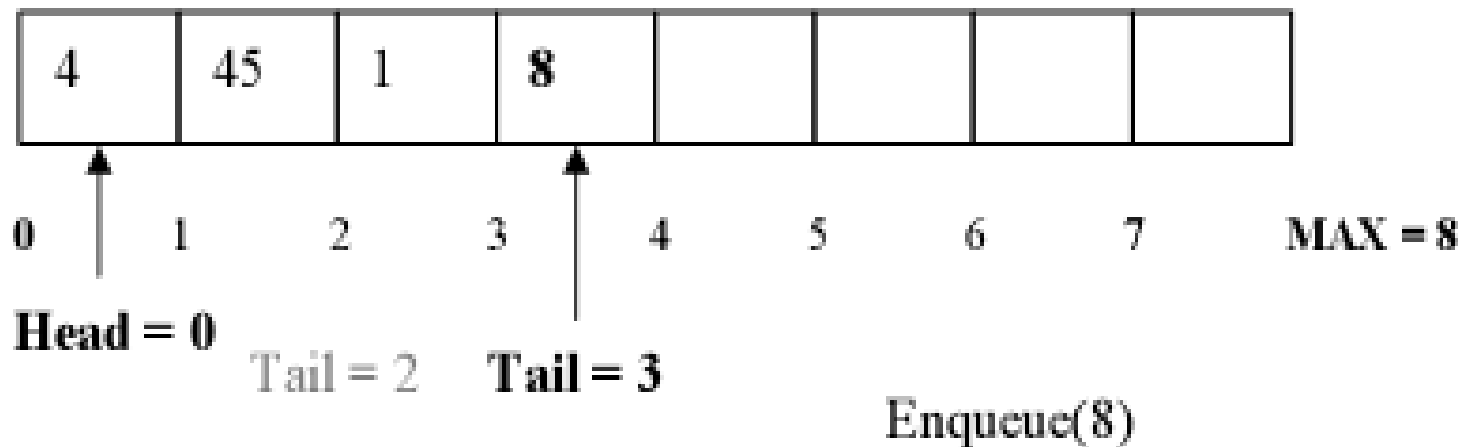
Head = 0

tail = 7

tail = max - 1

Fungsi Enqueue

- Untuk menambahkan elemen ke dalam Antrian, penambahan elemen selalu dilakukan pada elemen paling **belakang**
- Penambahan elemen selalu menggerakkan variabel Tail dengan cara menambahkan Tail terlebih dahulu



Fungsi Enqueue

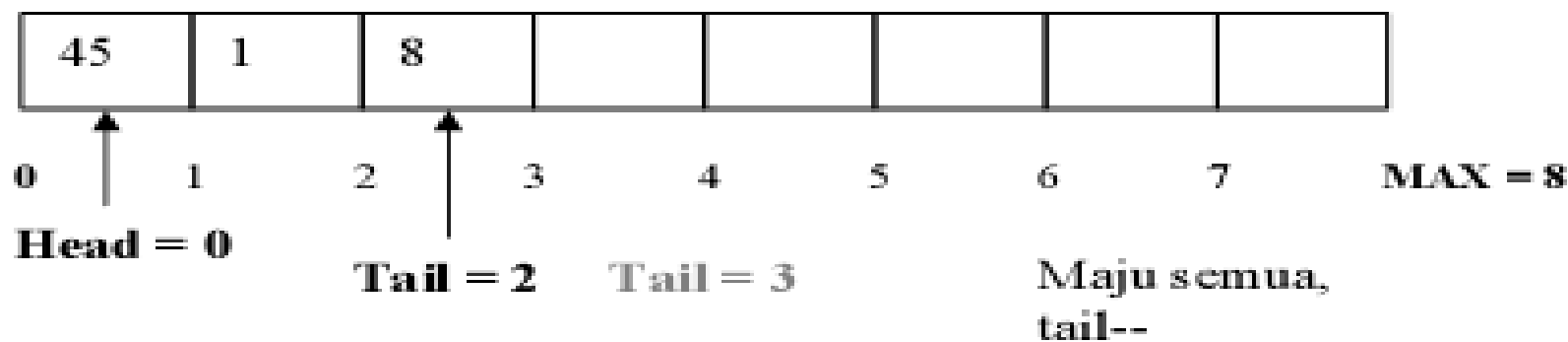
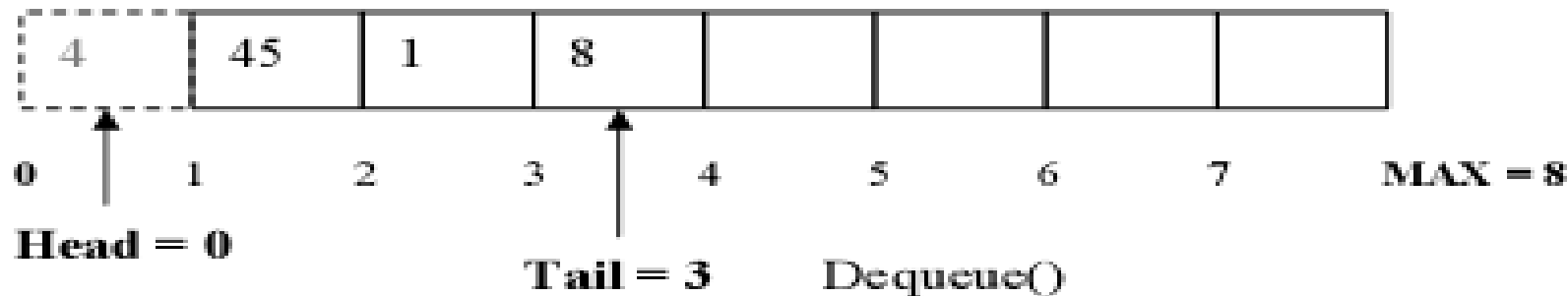
```
void Enqueue(int data){  
    if(IsEmpty()==1){  
        antrian.head=antrian.tail=0;  
        antrian.data[antrian.tail]=data;  
        printf("%d masuk!", antrian.data[antrian.tail]);  
    } else  
    if(IsFull()==0){  
        antrian.tail++;  
        antrian.data[antrian.tail]=data;  
        printf("%d masuk!", antrian.data[antrian.tail]);  
    }  
}
```

Fungsi Dequeue

- Digunakan untuk menghapus elemen terdepan (head) dari Antrian
- Dengan cara : menggeser semua elemen antrian kedepan dan mengurangi Tail dgn 1. Penggeseran dilakukan dengan menggunakan looping

```
int Dequeue() {  
    int i;  
    int e = antrian.data[antrian.head];  
    for(i=antrian.head; i<=antrian.tail-1; i++){  
        antrian.data[i] = antrian.data[i+1];  
    }  
    antrian.tail--;  
    return e;  
}
```

Fungsi Dequeue



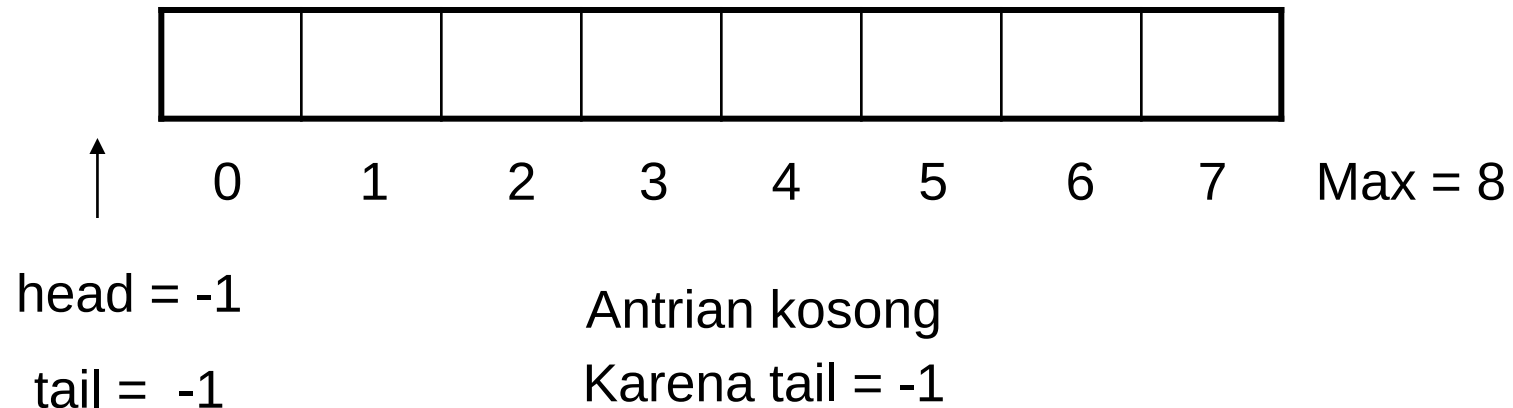
Fungsi Clear

- Untuk menghapus elemen-elemen Antrian dengan cara membuat Tail dan Head = -1
- Penghapusan elemen-elemen Antrian sebenarnya tidak menghapus arraynya, namun hanya mengeset indeks pengaksesan-nya ke nilai -1 sehingga elemen-elemen Antrian tidak lagi terbaca sehingga mengembalikan antrian seperti keadaan semula

Fungsi Clear

```
void Clear() {  
    antrian.head=antrian.tail=-1;  
    printf("data clear");  
}
```

Antrian setelah di lakukan Clear



Latihan I Struktur Data (Pertemuan 9)

Berikan gambaran/ilustrasi dari kasus antrian berikut :

- a) Diketahui suatu Antrian/queue dgn max = 6.
- b) Lakukan Enqueue 4 elemen ke dalam antrian, dimanakah posisi Head dan Tail ?
- c) Kemudian lakukan Dequeue 2 elemen dari antrian. Maka dimana posisi Head dan Tail ?
- d) Dari keadaan diatas, bagaimanakah kondisi IsFull dan IsEmpty nya ?

**Jawaban dibahas dgn
menggunakan contoh program**

Contoh program queue

[klikdisini](#)

Latihan Soal II Struktur Data (Pertemuan 9)

1. Prinsip antrian adalah?
 - a. First In First Out
 - b. First Out First In
 - c. Never In First Out
 - d. Last Out Last In
 - e. Last In First Out

2. Bentuk deklarasi manipulasi linked list harus menggunakan suatu pointer penunjuk ke node pertama (Head) dalam linked list yaitu?
 - a. TNode *head *Node
 - b. Head *TNode
 - c. Head *TNode *head
 - d. TNode *head
 - e. TNode +head

2. Bentuk deklarasi manipulasi linked list harus menggunakan suatu pointer penunjuk ke node pertama (Head) dalam linked list yaitu?

- a. TNode *head *Node
- b. Head *TNode
- c. Head *TNode *head
- d. TNode *head
- e. TNode +head

3. Yang tidak termasuk dalam operasi antrian, adalah ...

- | | |
|------------|------------|
| a. Clear | d. Push |
| b. Enqueue | e. Dequeue |
| c. IsFull | |

3. Yang tidak termasuk dalam operasi antrian, adalah ...
 - a. Clear
 - b. Enqueue
 - c. IsFull
 - d. Push
 - e. Dequeue

4. Menghapus elemen dari antrian dilakukan dari posisi :
 - a. Tengah / Middle
 - b. Depan / Head
 - c. Bawah / bottom
 - d. Belakang / Tail
 - e. Atas / Top

4. Menghapus elemen dari antrian dilakukan dari posisi :
 - a. Tengah / Middle
 - b. Depan / Head
 - c. Bawah / bottom
 - d. Belakang / Tail
 - e. Atas / Top

5. Maksud dari perintah program `antrian.head=antrian.tail=-1;` adalah untuk
 - a. Menambah elemen antrian
 - b. Mengecek kondisi antrian kosong atau tidak
 - c. Mengecek kondisi antrian penuh atau tidak
 - d. Membentuk atau menghapus semua elemen antrian
 - e. Menghapus elemen antrian

5. Maksud dari perintah program `antrian.head=antrian.tail=-1;` adalah untuk
- a. Menambah elemen antrian
 - b. Mengecek kondisi antrian kosong atau tidak
 - c. Mengecek kondisi antrian penuh atau tidak
 - d. Membentuk atau menghapus semua elemen antrian
 - e. Menghapus elemen antrian

1. Prinsip antrian adalah?
- a. First In First Out c. Never In First Out
 - b. First Out First In d. Last Out Last In
 - e. Last In First Out