

Worksheet pertemuan 8 – 2
Algoritma dan Struktur Data
Queue Dinamis

NIM : 23523195

Nama : Bintang Rizqi Anugrah

A. Membuat Folder Untuk Menyimpan Hasil Praktikum

1. Siapkan folder kosong dengan nama menggunakan NIM masing-masing. Jika folder NIM pada pertemuan sebelumnya mau dimanfaatkan, jangan lupa pindahkan dulu isinya ke folder lain sebagai arsip.
2. Folder ini akan dijadikan tempat untuk menyimpan semua pdf dari worksheet ini beserta fail praktikum lainnya.

B. Membuat class Queue

1. Gunakan fail ***LinkedList.java*** yang sudah Anda buat pada pertemuan sebelumnya dan letakkan dalam satu *package* dengan fail ***Queue.java*** yang akan dibuat.
2. Silakan buat sebuah class dengan nama **Queue**
3. Kemudian salin tempel kode pogram di bawah ini

```
package queuedinamis; // ingat kembali materi FPA tentang package!

/**
 *
 * @param <E>
 */
public class Queue<E> extends LinkedList {

    public Queue() {

    }

    public Object peek(){
        return getFirst();
    }

}
```

C. Membuat dan Menjalankan Main Method

1. Silakan buat sebuah main method class dengan nama **TestQueueDinamis**
2. Kemudian salin tempel kode pogram di bawah ini

```
package queuedinamis;

public class TestQueueDinamis {
    public static void main(String[] args) {
```

```

        Queue<String> mhs = new Queue<>();

        mhs.add("Agus");
        mhs.add("Sinta");
        mhs.add("Bambang");

        System.out.println("Elemen pertama: "+mhs.peek());

        mhs.remove();
        System.out.println("Elemen pertama: "+mhs.peek());

        mhs.clear();
        System.out.println("Elemen pertama: "+mhs.peek());
    }
}

```

3. Jalankan *main method* **TestQueueDinamis** dan hasil tangkapan layar keluaran dari program silakan letakkan di bawah ini

```

PS C:\Users\ROG STRIX\Documents\ASD\Week_6.2\Week_6.2>
s\ASD\Week_6.2\Week_6.2\bin' 'queuedinamis.TestQueueDina
Elemen pertama: Agus
Elemen pertama: Sinta
Elemen pertama: null
PS C:\Users\ROG STRIX\Documents\ASD\Week_6.2\Week_6.2>

```

D. Membuat Class QueueCircular

1. Silakan buat sebuah class dengan nama **QueueCircular**
2. Kemudian salin tempel kode pogram di bawah ini!

```

package queuecircular;

public class QueueCircular {
    private Object [] queueCirc;
    private int head;
    private int tail;
    private int maxsize;
}

```

```

public QueueCircular(int size){
    maxsize = size;
    queueCirc = new Object[maxsize];
    head = 0;
    tail = queueCirc.length - 1;
}

public boolean isEmpty(){
    return ((tail+1) % queueCirc.length == head);
}

public boolean isFull(){
    return ((tail+2) % queueCirc.length == head);
}

public void enqueue(Object x){
    if (!isFull()) {
        tail = ((tail+1) % queueCirc.length);
        queueCirc[tail] = x;
        System.out.println("Enqueue -> "+x);
    }else{
        System.out.println("Queue is full");
    }
}

public Object dequeue(){
    Object temp = null;
    if( !isEmpty() ) {
        temp = queueCirc[head];
        head = ((head+1) % queueCirc.length);
    }
    return temp;
}
}

```

E. Membuat Main Method TestQueueCircular

1. Silakan buat sebuah main method class dengan nama **TestQueueCircular**
2. Kemudian salin tempel kode program di bawah ini

```

package queuircular;

public class TesQueueCircular {
    public static void main(String[] args) {
        QueueCircular q = new QueueCircular(10);
    }
}

```

```

        int i;
        Integer j;

        for(i=0;i<10;i++){
            j = (int) (Math.random()*100);
            q.enqueue(j);
        }
    }
}

```

3. Jalankan fail **TestQueueCircular.java** dan hasil tangkapan layar keluaran dari program silakan letakkan di bawah ini

```

PS C:\Users\ROG STRIX\Documents\ASD\Week_8.2\queuecircular>
Documents\ASD\Week_8.2\queuecircular\bin' 'queuecircular.TesQueu
Enqueue -> 21
Enqueue -> 11
Enqueue -> 84
Enqueue -> 18
Enqueue -> 86
Enqueue -> 12
Enqueue -> 33
Enqueue -> 28
Enqueue -> 34
Queue is full
PS C:\Users\ROG STRIX\Documents\ASD\Week_8.2\queuecircular>

```

4. Buat sebuah *method* tambahan di dalam *class* **QueueCircular** yang digunakan untuk menampilkan isi dari queue mulai dari elemen pertama yang dimasukkan hingga elemen terakhir. Salin tempel kode program *method* yang Anda buat di bagian di bawah ini!

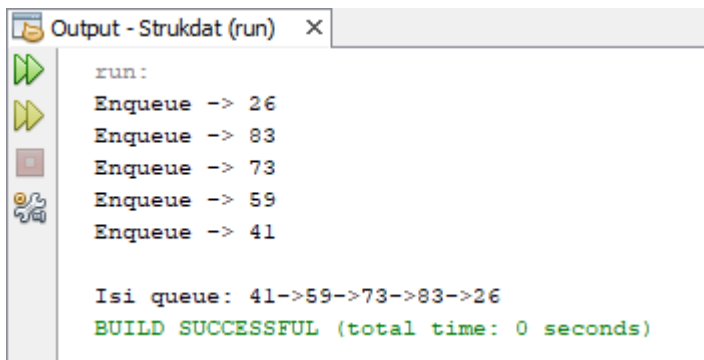
```

public void printQueue() {
    if (isEmpty()) {
        System.out.println("Queue is empty");
    } else {
        int i = head;
        while (i != tail){
            System.out.println(queueCirc[i] + "->");
        }
    }
}

```

```
        i = (i + 1) % queueCirc.length;
    }
    System.out.println(queueCirc[i]);
    System.out.println();
}
}
```

*Contoh tampilan isi Queue



The screenshot shows an IDE output window titled "Output - Strukdat (run)". It displays the output of a Java program. The output starts with "run:" followed by five "Enqueue ->" messages with values 26, 83, 73, 59, and 41. Below these, it shows "Isi queue: 41->59->73->83->26", indicating a circular queue where the oldest element (26) has been removed. The final line is "BUILD SUCCESSFUL (total time: 0 seconds)".

```
run:
Enqueue -> 26
Enqueue -> 83
Enqueue -> 73
Enqueue -> 59
Enqueue -> 41

Isi queue: 41->59->73->83->26
BUILD SUCCESSFUL (total time: 0 seconds)
```

5. Jalankan fail ***TestQueueCircular.java*** sekali lagi dan hasil tangkapan layar keluaran dari program silakan letakkan di bawah ini

```
PS C:\Users\ROG  
IX\Documents\ASD  
Enqueue -> 98  
Enqueue -> 9  
Enqueue -> 17  
Enqueue -> 58  
Enqueue -> 13  
Enqueue -> 39  
Enqueue -> 10  
Enqueue -> 41  
Enqueue -> 73  
Queue is full  
98->  
9->  
17->  
58->  
13->  
39->  
10->  
41->  
73
```

***Catatan**

- Jangan lupa simpan juga fail worksheet ini (yang sudah diisi) sebagai fail pdf di folder NIM anda.
- Sertakan juga fail-fail di bawah ini di dalam folder yang Anda gunakan
 - `LinkedList.java`
 - `Queue.java`
 - `TestQueueDinamis.java`
 - `QueueCircular.java`

- `TestQueueCircular.java`
- **Kompres folder ini sebagai fail ZIP kemudian kumpulkan di classroom atau ruang pengumpulan lain di kelas masing-masing.**