# 2017

# MODUL PRAKTIKUM

# TI014 – STRUKTUR DATA Season 1

Modul praktikum ini berisikan materi praktikum struktur data yang telah dipelajari dalam perkuliahan teori, harapannya mahasiswa dapat memahami dan menerapkan struktur data sederhana. Kemudian mengembangkannya kedalam bentuk modifikasi program

# LUZI DWI OKTAVIANA, S.Kom, M.MSI STMIK AMIKOM PURWOKERTO 3/7/2017

# PERTEMUAN KE-2

# TUJUAN PRAKTIKUM

- Mahasiswa mampu memahami dan menerapkan tentang konsep struktur data sederhana
- Mahasiswa mampu menganalisa bentuk dari struktur data sederhana
- Mahasiswa mampu membuat pemecah masalah kedalam bentuk algoritma dan struktur data
- Mahasiswa mampu mencari penyebab kesalahan program
- Mahasiswa dapat menerapkan materi struktur data primitive (pernyataan if, if-else, if else if, switch case)

### MEMULAI APLIKASI PRAKTIKUM

1. Buka aplikasi Eclipse Juno dengan cara *double click*. Aplikasi ini digunakan untuk menegtikkan kode program Java pada mata kuliah Struktur Data dan mengeksekusi kode program tersebut



Gambar 1.1 Icon / Shortcut Aplikasi Eclipse Juno

2. Setelah Aplikasi terbuka, sortir menu FILE > New > Java Project , seperti terlihat pada Gambar 1.2

st 🌐	ava - Eclipse SDK		-	-		-
File	Edit Source	Refactor Navigate S	earch P	rojec	t Run Window Help	
	New	Alt+Sl	hift+N 🔨	ピ	Java Project	
	Open File			Ċ	Project	
	Close	c	Ctrl+W	₿	Package	ĺ
	Close All	Ctrl+Sh	nift+W	C	Class	k
Q	Save		Ctrl+S	œ	Interface	
	Save As			G	Enum	
Ē.	Save All	Ctrl+S	hift+S	@	Annotation	
العلة	Revert			<b>₽</b> ₽	Source Folder	ſ
				墫	Java Working Set	
	Move				Folder	ľ
	Rename		F2	<b>F</b> ∳	File	i

Gambar 1.2 Membuat Java Project

3. Setelah diklik, akan muncul tampilan seperti Gambar 1.3. pada kolom Project Name diisi dengan format SD\_NamaMhs\_NIM (contoh : SD\_Budi\_16112222), kemudian pada kolom JRE klik *button* Use a Project Specific JRE, setelah itu klik tombol FINISH. Java project ini bertujuan sebagai wadah dalam kita membuat *class program java* agar terkumpul menjadi satu. Dan pembuatan java project ini cukup hanya dilakukan satu kali saja. Keterangan lebih lanjut bias dilihat pada Gambar 1.3

New Java Project					
Create a Java Project					
Create a Java project in the workspace or in an external location.					
Project name: SD_NamaMhs_NIM					
☑ Use default location					
Location: C:\Users\amikom\workspace\SD_NamaMhs_NIM Browse					
JRE					
Use an execution environment JRE:					
Use a project specific JRE:     jre1.8.0_121					
Use default JRE (currently jset 2.0_UL)					
Project layout					
Use project folder as root for sources and class files     Create senarate folders for sources and class files     Configure default					
Working sets					
Working sets:					
3					

Gambar 1.3 Tampilan Pop-Up Message Java Project

4. Langkah selanjutnya adalah pembuatan *class*. Class ini dibuat dengan tujuan kode program yang kita buat bias kita eksekusi. Pemberian nama kelas juga sesuai dengan kode program yang akan kita buat. Cara pembuatan class adalah dengan Sortir menu FILE > New > Class. Keterangan lebih lanjut dapat dilihat pada Gambar 1.4 berikut ini :

🔿 Java - StrukturData/src/rekursif.java - Eclipse SDK							
File	Edit Source	Refactor	Navigate	Search	Projec	t Run Window	Help
	New		Alt+	Shift+N	• 🏄	Java Project	
	Open File				Ľ	Project	
	Close			Ctrl+W	₿	Package	
	Close All		Ctrl+	Shift+W	G	Class	>
P	Save			Ctrl+S	Û	Interface	
	Save As				G	Enum	
na.	Save All		Ctrl	+Shift+S	@	Annotation	
المله	Revert		- Cui	· onne · o	<b>₽</b> ₽	Source Folder	
					8	Java Working Set	
	Move				<b>*</b>	Folder	
	Rename			F2	<b>□</b>	File	
8	Refresh			F5	R	Untitled Text File	
	Convert Line De	limiters To	)			JUnit Test Case	
Đ	Print			Ctrl+P		Other	Ctrl+N

Gambar 1.4 Pembuatan *Class* pada Java

5. Setelah *class* selesai dibuat akan tampil pesan Pop-Up seperti Gambar 1.5. Pada kolom *Source Folder* pastikan nama nya sesuai dengan Nama Java Project yang telah Anda buat sebelumnya. Kemudian isikan nama kelas yang akan dibuat (contoh : Perintah\_IF\_Else) kemudian *checklist* pada bagian : public static void main (String [] args) dan inherited abstract methods, setelah itu klik tombol FINISH

New Java Class	and the second se	
Java Class	efault package is discouraged.	1
Source folder:	SD_BUDI_16112222/src	Browse
Package:	(default)	Browse
Enclosing type:		2 Browse
Name: Modifiers:	Perintah_IF_ELSE          public	
Superclass:	java.lang.Object	Browse
Interfaces:		Add Remove
Which method state	os would you like to create? public static void main(String[] args) Constructors from superclass Inherited abstract methods	
Do you want to add	reomments? (Configure templates and default value <u>here</u> )	
?	Finish	Cancel

Gambar 1.5 Pop-Up Message Class Pada Java

6. Setelah itu lembaran projek java kita siap untul dieksekusi dengan perintah yang akan kita buat. Pada pembuatan class ini bias kita lakukan setiap kali kita akan mengetikkan kode program yang baru, sesuai dengan perintah yang akan kita buat.



Gambar 1.6 WorkSheet yang siap dieksekusi pada Java

7. Setiap kali Anda selesai mengetikkan kode program jangan lupa untuk menyimpan **(Save : Ctrl+S)** terlebih dahulu sebelum dijalankan **(Run : Ctrl + F11).** 

#### 1. Praktikum 1 Perintah IF

```
Program untuk mengambil keputusan
        dengan mengguñakan pernytaan IF
                           . .
import java.util.Scanner;
public class Perintah_IF_Usia {
        /**
         * @param args
         */
        public static void main(String[] args) {
                 // TODO Auto-generated method stub
// Buat objek Scanner
                 Scanner kbd = new Scanner(System.in);
                 // Membaca Data Usia Yang dimasukkan
System.out.print("Berapa usia Anda? ");
                 byte usia = kbd.nextByte();
                 String pesan = "17 Tahun atau lebih adalah saat untuk mengukir prestasi!!";
                 if (usia<17)
                         pesan = "SEMANGAT terus untuk menuju usia 17-mu!!";
                 System.out.println(pesan);
                 kbd.close();
        }
}
```

#### 2. Praktikum 2 Perintah IF ELSE

a. Penentu Predikat Kelulusan Berdasarkan IPK

```
public class Perintah_IF_Else {
    /***
    * @param args
    */
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        double IPK=3.8;
        if (IPK > 3.75)
            System.out.println("Anda Lulus dengan predikat CUMLAUDE");
        else
            System.out.println("Anda Lulus, namun tidak mendapat predikat CUMLAUDE");
    }
}
```

#### b. Penentu Bilangan ganjil dan genap

```
Praktikum Pemakaian IF Else Untuk Menetukan
         suatu bilangan termasuk genap atau ganjil
import java.util.Scanner;
public class Perintah_IF_ELSE_GanGen {
         /**
          * @param args
          * /
         public static void main(String[] args) {
                  // TODO Auto-generated method stub
                  Scanner kbd = new Scanner(System.in);
                  // Membaca Data Bilangan
                  System.out.print("Masukkan sebuah bilangan bulat : ");
                  int bilangan = kbd.nextByte();
                  if (bilangan % 2 == 1){
                           System.out.println("Bilangan : " + bilangan + " tidak habis dibagi 2");
                           Sýstem.out.println("Jadi ťermasuk bilangán GANJIL");
                  else {
                           System.out.println("Bilang : " + bilangan + " habis dibagi 2");
System.out.println("Jadi termasuk bilangan GENAP");
                  }
                  kbd.close();
         }
}
```

### 3. Praktikum 3 Perintah Nested IF

#### a. Praktikum IF Bersarang 1

```
public class Perintah_Nested_IF {
           /**
           * @param args
           ×
          public static void main(String[] args) {
                    // TODO Auto-generated method stub
char nilai='B';
                    if (nilai=='A')
                    System.out.println("Nilai Anda Istimewa");
else if (nilai=='B')
                    System.out.println("Nilai Anda Sangat Memuaskan");
else if (nilai=='C')
                    System.out.println("Nilai Anda Cukup");
else if (nilai=='D')
                              System.out.println("Nilai Anda Kurang");
                    else if (nilai=='E')
                              System.out.println("Anda Tidak LULUS");
                    else
                              System.out.println("MAAF, Huruf Yang Anda Masukan SALAH");
          }
}
       b. Praktikum IF Bersarang 2
import java.util.Scanner;
public class Perintah_Nested_IF_2 {
          /**
           * @param args
          public static void main(String[] args) {
                    // TODO Auto-generated method stub
                               Scanner masuk = new Scanner (System.in);
                               int nil;
char nil_huruf='E';
System.out.print("Masukkan Bilangan : ");
                               nil=masuk.nextInt();
                               if ((nil>=0) && (nil<=20))
nil_huruf='E';</pre>
                               else if ((nil>20) && (nil<=40))
nil_huruf='D';
else if ((nil>40) && (nil<=60))
nil_huruf='C';
                               else if ((nil>60) && (nil<=80))
                               nil_huruf='B';
                               else if ((nil>80) && (nil<=100))
                               nil_huruf='A';
                               else
                               System.out.println("Nilai yang anda masukkan Salah");
System.out.println("Nilai huruf : " + nil_huruf);
```

#### Praktikum 4 Perintah Switch Case

#### a. Program Penentu Arah Mata Angin

```
public class Perintah_Swtich_Case {
         /**
          * @param args
          ×
         public static void main(String[] args) {
    // TODO Auto-generated method stub
    char arah='N';
                   switch (arah){
                   case 'N':
                            System.out.println("Arah Utara");
                            break;
                   case 'W':
                            System.out.println("Arah Barat");
                            break;
                   case 's':
                            System.out.println("Arah Selatan");
                            break;
                   case 'E':
                            System.out.println("Arah Timur");
                            break;
default:
                                      System.out.println("Arah yang anda masukkan salah");
                   }
         }
}
```

#### b. Program Aritmatika

```
-----
         Program Aritmatika Menggunakan Switch Case
         _____
import java.util.Scanner;
public class Perintah_SwC {
         /**
          * @param args
          * /
         public static void main(String[] args) {
                  // TODO Auto-generated method stub
                  float hasil;
                  Scanner data = new Scanner (System.in);
                  int bil1, bil2;
                  String op;
                  char operator;
System.out.print("Masukan Bil 1 : ");
                  bil1=data.nextInt();
System.out.print("Masukkan Bil 2 : ");
                  bil2=data.nextInt();
                  System.out.print("Masukkan Operator :
                                                                  ");
                  op=data.next();
                  operator=op.charAt(0);
                  switch(operator){
case '*' : hasil = bil1 * bil2;break;
case '/' : hasil = (float)bil1/bil2;break;
case '%' : hasil = bil1 % bil2;break;
case '+' : hasil = bil1 + bil2;break;
case '-' : hasil = bil1 - bil2;break;
                  default : hasil=0;
                  System.out.println("Hasil Operasinya Adalah = " + hasil);
                  data.close();
         }
}
```

# TUGAS INDIVIDU 1

- a) Buatlah sebuah program modifikasi yang didalamnya terdiri dari programprogram yang sudah pernah dipraktikkan saat praktikum min 3 pilihan
- b) Ketentuan program nya adalah ketika program pertama kali dijalankan muncul tampilan menu program dengan pilihan nomor setiap butir program sebagai contoh lihat gambar 1.1

// -----// // MENU PROGRAM // Silahkan Masukkan Nomor Program yang akan dijalankan : 1. Program Cek Usia 2. Program Aritmatika 3. Program Ganjil Genap Masukkan Nomor Menu : |

Gambar 1.1 Contoh Menu Pilihan

- c) Ketika nomor tertentu yang dimasukkan maka akan ditampilkan program
- d) Di akhir program yang sedang dijalankan berikan opsi untuk keluar atau kembali ke Menu awal
- e) Buat laporan akhir dilengkapi kode program dan screenshoot tampilan program ketika dijalankan

# PERTEMUAN 3

### TUJUAN PRAKTIKUM

- Mahasiswa mampu memahami dan menerapkan tentang konsep struktur data sederhana
- Mahasiswa mampu menganalisa bentuk dari struktur data sederhana
- Mahasiswa mampu membuat pemecah masalah kedalam bentuk algoritma dan struktur data
- Mahasiswa dapat menerapkan materi struktur data primitive (pernyataan while, do...while, for, break, continue)

## PRAKTIKUM 2

Perulangan While

 a. Program 1 Penggunaan While

public class perulangan\_while {

```
/**
 * @param args
 */
public static void main(String[] args) {
    // TODO Auto-generated method stub
    int i=6;
    while (i>0){
      System.out.println("STUKTUR DATA SANGAT MUDAH");
      i --;
    }
}
```

#### b. Program 2 Penggunaan While

```
public class perulangan_while_2 {
    /**
    * @param args
    */
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        boolean hasil=false;
        int i=0;
        while(!hasil) {
        if (i==5)
            hasil=true;
        System.out.println("While dengan Boolean");
        i++;
      }
   }
}
```

#### 2. Perulangan Do-While

```
public class Ulang_Do {
    /**
    * @param args
    */
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        int jumlah=0;
        int i=0;
        do {
            jumlah = jumlah + i;
            System.out.println("i = "+i+ " dan jumlah = "+ jumlah);
            i++;
            }
        while (i<10);
    }
}</pre>
```

#### 3. Perulangan For

```
public class Abjad {
    public static void main(String[] args) {
        for (char huruf = 'A'; huruf <= 'Z'; huruf++)
            System.out.print(huruf);
        System.out.println(); // Pindah baris
    }
}</pre>
```

#### 4. Perulangan Break

```
// contoh program untuk melihat efek break
// pada pernyataan while
public class TesBreak {
    public static void main(String[] args) {
        int bil = 1;
        while (bil <= 10) {
            if (bil > 5)
                break;
            System.out.println(bil);
            bil++;
        }
    }
}
```

#### 5. Perulangan Continue

```
// Contoh program untuk melihat efek continue
pada pernyataan for
public class TesContinue {
    /**
    * @param args
    */
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        for (int bil = 0; bil < 10; bil++) {
            if (bil == 5)
                continue;
               System.out.println(bil);
            }
        }
    }
}</pre>
```

## TUGAS INDIVIDU 2

- 1. Buatlah sebuah program java untuk membuat sebuah **keluaran** perkalian 0 sampai 10
- 2. Buatlah sebuah program java yang digunakan untuk **memasukkan** data bertipe **double** dan menampilkan hasilnya
- 3. Buatlah sebuah program java untuk membuat kalkulator sederhana

# PERTEMUAN 4

### **TUJUAN PRAKTIKUM**

- 1. Mahasiswa mampu dan dapat menerapkan Array Satu Dimensi dengan Array Dua Dimensi
- 2. Mahsiswa mampu dan dapat menerapkan Duplikasi Array, string, rekusif
- 3. Mahasiswa mampu dan dapat mengidentifikasi Kelas dan Objek dalam Struktur Data
- 4. Mahasiswa mampu dan dapat memodifikasi Array datu dimensi, Array dua dimensi dan rekursif

### LATIHAN PRAKTIKUM

#### 1. Array Satu Dimensi

```
2. Array Dua Dimensi
```

```
a. Program penggunaan Array Dua DImensi
public class DataTes {
    public static void main(String[]_args) {
          int nilai[][] = new int[5][3];
          // Mengisi elemen larik
         nilai[0][0] = 77;
nilai[0][1] = 80;
nilai[0][2] = 92;
         nilai[1][0] = 81;
nilai[1][1] = 80;
         nilai[1][2] = 88;
         nilai[2][0] = 75;
nilai[2][1] = 82;
nilai[2][2] = 76;
         nilai[3][0] = 64;
         nilai[3][1] = 76;
nilai[3][2] = 75;
         nilai[4][0] = 80;
nilai[4][1] = 83;
nilai[4][2] = 72;
          // Menampilkan semua elemen
         for (int i = 0; i < 5; i++) {
               System.out.println("Isi baris " + i + ' ');
              // Pindah baris
               System.out.println();
          }
    }
```

#### 3. Duplikasi Array

```
// Contoh penyalinan array
// menggunakan clone()
                             -----
public class DuplikasiArray {
     public static void main(String[] args) {
    String[] musik = { "Jazz", "Samba", "Dixie", "Fusion" };
           // Salin array musik ke aliranJazz
           String[] aliranJazz = musik.clone();
           System.out.print("musik: ");
           tampilkanArray(musik);
           System.out.print("aliranJazz: ");
           tampilkanArray(aliranJazz);
           // Lakukan perubahan pada aliranJazz
           aliranJazz[1] = "Bepop";
           // Cek isi kedua array
System.out.println("setelah ada perubahan:");
System.out.print("musik: ");
           tampilkanArray(musik);
           System.out.print("aliranJazz: ");
           tampilkanArray(aliranJazz);
     }
     public static void tampilkanArray(String[] x) {
   System.out.print("{" + x[0]);
   for (int i = 1; i < x.length; i++)
        System.out.print(", " + x[i]);</pre>
           System.out.println('}');
     }
}
```

```
4. String
public class ContohString {
     public static void main(String[] args) {
          String tes = "Semangat Terus!";
          System.out.println(tes);
          System.out.println("length() :
System.out.println("toUpperCase():
System.out.println("toLowerCase():
System.out.println("replace() :
                                                          + tes.length());
                                                        ...
                                                           + tes.toUpperCase());
                                                       ....
                                                          + tes.toLowerCase());
                                   tes.replace("Terus", ", bung"));
                                                     : " + tes.endswith("us!"));
          System.out.println("endwith
                                                     : " + tes.substring(0, 8));
          System.out.println("substr()
     }
}
          5. Rekursif
public class contoh_rekursif_dua {
```

```
/**
 * @param args
 */
public static void main(String[] args) {
    // TODO Auto-generated method stub
        int a=4;
        System.out.println("Nilai Faktorial dari "+a+" adalah : "+factorial(a));
}
static int factorial (int n) {
    if (n==0)
        return 1;
    else
        return n*factorial(n-1);
}
```

## TUGAS KELOMPOK 1

- 1. Buatlah sebuah program yang digunakan untuk **memasukkan** data bertipe string dan menampilkannya
- Buatlah sebuah program yang berfungsi untuk menghitung nilai pangkat. Anda bisa menggunakan metode perulangan for, while, atau rekursi. (Petunjuk : method yang dibuat harus memiliki 2 parameter input)

- 3. Bagaimana cara mendeklarasikan *Array* yang akan dipakai untuk menampung lima nama bunga : yaitu mawar, melati, kenanga, arumdalu dan anthurium ? bagaimana pula pernyataan untuk mengisikan nama-nama bunga tersebut ? (Buatkan program array tersebut dalam java)
- 4. Suatu array dengan data seperti berikut :



- a. Buatkan program untuk menghitung nilai rata-rata dari kelima elemen tersebut
- b. Buatlah pula program untuk memperoleh nilai terkecil
- Buatlah sebuah program dengan strungtur data String untuk mengidentifikasi kata yang dimasukkan menggunakan method CharArt dan IndexOf

# PERTEMUAN 5

### TUJUAN PRAKTIKUM

- 1. Mahasiswa mampu menjelaskan pengertian struktur data tumpukan
- 2. Mahasiswa mampu nenerapkan Operasi dasar pada tumpukan dalam program java sederhana
- 3. Mahasiswa mampu menerapkan tumpukan (stack) dengan Array

#### LATIHAN PRAKTIKUM

1. Implementasi Tumpukan menggunakan metode Stack

```
Contoh tumpukan
          menggunakan kelas Stack
                                                 _____
import java.util.Stack;
public class ContohTumpukan_1 {
       public static void main(String[] args) {
              // Buat objek
              Stack<String> tumpukan = new Stack<String>();
              // Masukkan 9 nama
             // Masukkan 9 nama
tumpukan.push("Aminudin");
tumpukan.push("Zaskia");
tumpukan.push("Rina Melati");
tumpukan.push("Farhan");
tumpukan.push("Agnes Monica");
tumpukan.push("Dian Susilo");
tumpukan.push("Karman Julia");
tumpukan.push("Kartika Putri");
             // Kosongkan isi tumpukan dan tampilkan
System.out.println("Isi tumpukan:");
while (! tumpukan.empty()) {
                     String nama = tumpukan.pop();
                     System.out.println(nama);
              }
       }
```

2. Implementasi tumpukan dengan Array

```
-----
    Contoh tumpukan
      menggunakan array
                                      _____
public class ContohTumpukan {
      public static void main(String[] args) {
    Tumpukan tumpukan = new Tumpukan(); // Buat objek
           // Masukkan 9 nama
tumpukan.push("Aminudin");
tumpukan.push("Zaskia");
tumpukan.push("Farhan");
tumpukan.push("Farhan");
tumpukan.push("Dian Susilo");
tumpukan.push("Larman Julia");
tumpukan.push("Dani Adipta");
tumpukan.push("Kartika Putri");
           // Kosongkan isi tumpukan dan tampilkan
           System.out.println("Isi tumpukan:");
           while (! tumpukan.empty()) {
                 String nama = tumpukan.pop();
                 System.out.println(nama);
           }
            // Lakukan pop() sewaktu tumpukan kosong
           System.out.println("Mencoba pop() pada tumpukan kosong:");
           System.out.println(tumpukan.pop());
      }
}
class Tumpukan {
     final int MAKSDATA = 8;
     private String[] tumpukan = new String[MAKSDATA];
     private int puncak;
      // Konstruktor
     public Tumpukan() {
           puncak = -1;
      }
     // Memasukkan data ke tumpukan
     Nilai balik berupa false
              kalau tumpukan sudah penuh
     public boolean push(String data) {
    // Periksa kalau tumpukan penuh
    if (puncak == (MAKSDATA - 1)) {
        System.out.print("Tumpukan penuh! ");
        System.out.println(data + " tidak dimasukkan.");
    };
               return false;
           }
```

# TUGAS INDIVIDU 3

1. Suatu tumpukan pada keadaan awal dalam keadaan kosong. Kemudian, terdapat deretan operasi seperti berikut :

Push('A') Push('B') Push('C') Pop Push('E') Push('F') Pop

}

Gambarkan output program keadaan tersebut diatas sesudah setiap operasi telah dilaksanakan

2. Terdapat dua keadaaan tumpukan seperti berikut :



Bagaimana cara agar tumpukan keadaan 1 berubah seperti keadaan 2?

3. Buatlah sebuah program untuk mewujudkan tumpukan dengan elemen berupa bilangan bulat dengan memanfaatkan kelas STACK

# PERTEMUAN 6

### TUJUAN PRAKTIKUM

- 1. Mahasiswa mampu menerapkan antrean dengan array pada program java sederhana
- 2. Mahasiswa mampu menerapkan antrean dengan ArrayDeque pada program java sederhana
- 3. Mahasiswa mampu memodifikasi penerapan antrean dengan Array dan ArrayDeque pada program java sederhana

### LATIHAN PRAKTIKUM

1. Antrean (Queue) menggunakan Array

```
// Konstruktor
public Antrean(int ukuran) {
    depan = 0;
    belakang = 0;
    maksElemen = ukuran;
    data = new String[ukuran];
}
// Memasukkan data ke antrean
11
      Nilai balik tidak ada
public void insert(String x) {
    int posisiBelakang;
    // Geser belakang ke posisi berikutnya
if (belakang == maksElemen - 1)
         posisiBelakang = 0;
    else
         posisiBelakang = belakang + 1;
    // Cek belakang apa sama dengan Depan
    if (posisiBelakang == depan) {
         System.out.println("Antrian penuh.");
System.out.println(x + " tidak dimasukkan.");
    }
    else {
         belakang = posisiBelakang;
         // Masukkan data
         data[be]akang] = x;
    }
}
public String remove() {
    if (empty()) {
         System.out.println("Antrian kosong");
return "";
    }
    if (depan == maksElemen - 1)
         depan = 0;
    else
         depan = depan + 1;
    return data[depan];
}
public boolean empty() {
    if (depan == belakang)
         return true;
    else
         return false;
}
```

2. Penerapan Antrean (Queue) menggunakan ArrayDeque

```
// Contoh Antrean Menggunakan kelas ArrayDeque
import java.util.ArrayDeque;
public class ContohAntrean2 {
    public static void main(String[] args) {
        ArrayDeque<String> daftar = new ArrayDeque<String>();
        // Masukkan 9 nama
        daftar.add("Aminudin");
        daftar.add("Zaskia");
        daftar.add("Farhan");
        daftar.add("Farhan");
        daftar.add("Karman Julia");
        daftar.add("Karman Julia");
        daftar.add("Kartika Putri");
        // Kosongkan isi antrean dan tampilkan
        System.out.println("Isi Antrean:");
        while (!daftar.isEmpty()) {
            String nama = daftar.remove();
            System.out.println(nama);
            }
        }
    }
}
```

### TUGAS INDIVIDU 4

- 1. Kenapa antrean dikatakan mempunyai sifat LIFO ? jelaskan !
- 2. Gambakan keadaaan antrean untuk setiap pemanggilan operasi berikut (dari keadaan awal hingga akhir)

Insert('A') Insert('B') Insert('C') remove Insert('X') remove Insert('Y') Insert('Z') remove

}

3. Berdasarkan contoh yang telah diberikan dalam perkuliahan praktikum, sebuah array berjumlah 20 hanya dapat menampung data 19 buah. Mengapa bias demikian ? jelaskan !

- 4. Buatlah sebuah antrean yang memanfaatkan ArrayDeque untuk memproses antrean bilangan pecahan
- 5. Buatlah sebuah program yang memanfaatkan ArrayDeque untuk memproses antrean mobil yang berisi informasi nomor polisi dan tipe mobil.

# PERTEMUAN KE-7

# RESPONSI

Tujuan diadakannya responsi adalah :

- 1. Mengukur seberapa jauh mahasiswa mampu dan memahami materi yang sudah disampaikan
- 2. Mengukur seberapa dalam mahasiswa memahami penerapan materi perkuliahan yang sudah diberikan kedalam bentuk pemrograman java sederhana
- 3. Mengidentifikasi tingkat kesulitan yang dihadapi oleh mahasiswa
- 4. Menambah poin atau nilai untuk kelengkapan penilaian akhir semester

### **MATERI RESPONSI**

- 1. Array
- 2. Stack
- 3. Queue