**LAPORAN**

**PRAKTIKUM STRUKTUR DATA**

**Pertemuan ke – 6**

****

Disusun Oleh :

Rendra Eka Herlambang

175410028

**LABORATORIUM TERPADU**

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER AKAKOM YOGYAKARTA**

**2018**

**TUMPUKAN/STACK**

1. **TUJUAN**

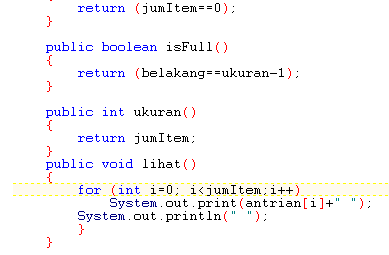
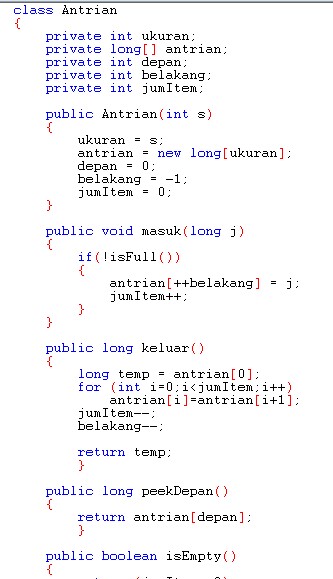
-Mahasiswa dapat membuat program dengan antrian

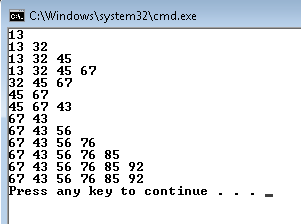
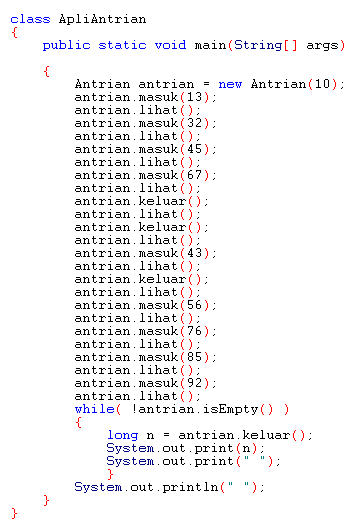
- Mahasiswa mampu mengimplementasikan antrian

- Mahasiswa mampu mengimplemetasikan bentuk data dengan menggunakan java.

-Mahasiswa mampu memodifikasi model tipe data int ke string.

1. **PEMBAHASAN PRAKTIK**





Pembahasan

class Antrian

{

private int ukuran;

private long[] antrian;

private int depan;

private int belakang;

private int jumItem;

//yaitu nama class Antrian,mendekrasikan ukuran,antrian,depan, belakang,jumItem

public Antrian(int s)

{

ukuran = s;

antrian = new long[ukuran];

depan = 0;

belakang = -1;

jumItem = 0;

}

//yaitu bentuk antrian public,dengan method long,nilai depan 0 nilai belakang -1 jumItem=0

public void masuk(long j)

{

if(!isFull())

{

antrian[++belakang] = j;

jumItem++;

}

}

//yaitu mendekrasikan method long j, jika tidak full antrian nambah 1 dan jumItem 1

public long keluar()

{

long temp = antrian[0];

for (int i=0;i<jumItem;i++)

antrian[i]=antrian[i+1];

jumItem--;

belakang--;

//yaitu mendekrasikan long keluar,long temp antrian dari 0,perulangan nilai x<jumlahItem maka nambah 1.nilai i+1,maka jumItem -1 dan belakang pun juga.

return temp;

}

public long peekDepan()

{

return antrian[depan];

}

//return tempt maka antrian depan

public boolean isEmpty()

{

return (jumItem==0);

}

//maka jika kosong maka jumItem =0

public boolean isFull()

{

return (belakang==ukuran-1);

}

//return yaitu nilai belakang jika sama maka ukuran-1

public int ukuran()

{

return jumItem;

}

public void lihat()

{

//yaitu method untuk melihat

for (int i=0; i<jumItem;i++)

System.out.print(antrian[i]+" ");

System.out.println(" ");

}

}

//yaitu perulangan i<jumItem i++

class ApliAntrian

{

public static void main(String[] args)

//yaitu membuat class ApliAntrian yang bersifat public.

{

Antrian antrian = new Antrian(10);

antrian.masuk(13);

antrian.lihat();

antrian.masuk(32);

antrian.lihat();

antrian.masuk(45);

antrian.lihat();

antrian.masuk(67);

antrian.lihat();

antrian.keluar();

antrian.lihat();

antrian.keluar();

antrian.lihat();

antrian.masuk(43);

antrian.lihat();

antrian.keluar();

antrian.lihat();

antrian.masuk(56);

antrian.lihat();

antrian.masuk(76);

antrian.lihat();

antrian.masuk(85);

antrian.lihat();

antrian.masuk(92);

antrian.lihat();

while( !antrian.isEmpty() )

{

//yaitu method antrian,yang masuk 13,32,45,67,43,56,76,85,92 maka data itu di lihat, maka perulangan jika antrian tidak kosong.

long n = antrian.keluar();

System.out.print(n);

System.out.print(" ");

}

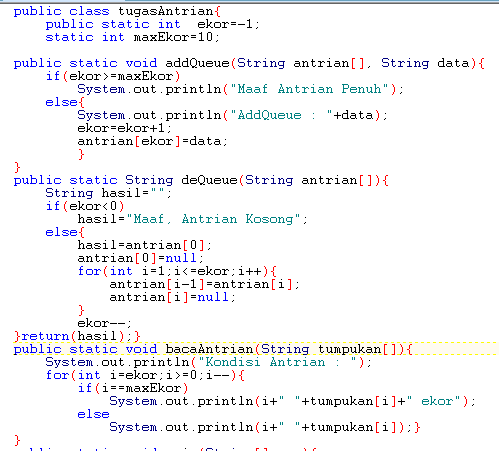
System.out.println(" ");

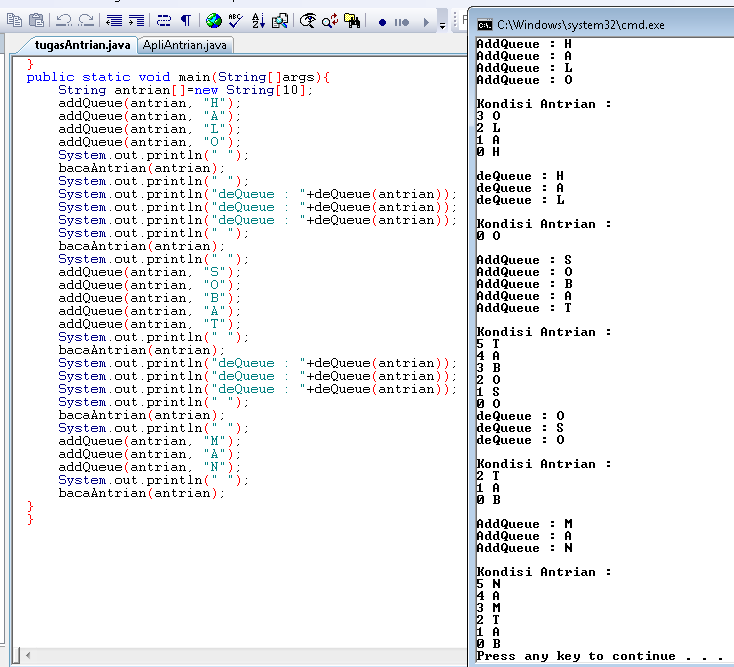
}

}

//yaitu hasil atau outputnya, antrian keluar maka di dekrasikan n.

Latihan





Pembahasan

public class tugasAntrian{

public static int ekor=-1;

static int maxEkor=10;

//yaitu class tugasAntrian. Yang mendekrasikan nilai ekor=-1 dan maxEkor=10

public static void addQueue(String antrian[], String data){

if(ekor>=maxEkor)

System.out.println("Maaf Antrian Penuh");

//yaitu jika ekor>=maxEkor maka antrian penuh.

else{

System.out.println("AddQueue : "+data);

ekor=ekor+1;

antrian[ekor]=data;

}

//dan jika tidak addQueue +data ,maka ekor+1,maka antrian, ekor=data

}

public static String deQueue(String antrian[]){

String hasil="";

if(ekor<0)

hasil="Maaf, Antrian Kosong";

//yaitu jika dequeue string maka jika ekor<0 maka antrian kosong.

else{

hasil=antrian[0];

antrian[0]=null;

for(int i=1;i<=ekor;i++){

antrian[i-1]=antrian[i];

antrian[i]=null;

}

//jika tidak hasil antrian 0,maka perulangan int=1<=ekor maka nambah 1 secara otomatis.

ekor--;

}return(hasil);}

public static void bacaAntrian(String tumpukan[]){

System.out.println("Kondisi Antrian : ");

for(int i=ekor;i>=0;i--){

if(i==maxEkor)

System.out.println(i+" "+tumpukan[i]+" ekor");

//yaitu kondisi perulangan yang mana i>=0 maka berkurang 1 dan jika i==maxEkor

else

System.out.println(i+" "+tumpukan[i]);}

}

//jika tidak maka tumpukan i

public static void main(String[]args){

String antrian[]=new String[10];

addQueue(antrian, "H");

addQueue(antrian, "A");

addQueue(antrian, "L");

addQueue(antrian, "O");

System.out.println(" ");

bacaAntrian(antrian);

//yaitu maka method antrian maka batasnya 10, dan isi antrian HALO dan baca antrian.

System.out.println(" ");

System.out.println("deQueue : "+deQueue(antrian));

System.out.println("deQueue : "+deQueue(antrian));

System.out.println("deQueue : "+deQueue(antrian));

//yaitu keluaran atau outputnya urutan 0-2 maka HAL

System.out.println(" ");

bacaAntrian(antrian);

System.out.println(" ");

addQueue(antrian, "S");

addQueue(antrian, "O");

addQueue(antrian, "B");

addQueue(antrian, "A");

addQueue(antrian, "T");

System.out.println(" ");

bacaAntrian(antrian);

//yaitu method antrian yang berisikan SOBAT maka antrian akan di baca.

System.out.println("deQueue : "+deQueue(antrian));

System.out.println("deQueue : "+deQueue(antrian));

System.out.println("deQueue : "+deQueue(antrian));

//yaitu maka dari dekrasi urutan 0-2 maka OSO

System.out.println(" ");

bacaAntrian(antrian);

System.out.println(" ");

addQueue(antrian, "M");

addQueue(antrian, "A");

addQueue(antrian, "N");

System.out.println(" ");

bacaAntrian(antrian);

//yaitu keluaran MAN pada antrian.

}

}

TUGAS

KESIMPULAN

Kita dapat menyimpulkan bahwa antrian sebagai sebuah jalur tunggu seperti sebuah jalur orang menunggu untuk beli tiket di mana orang yang pertama datang akan di layani terlebih dahulu,dan queue di definisikan sebagai sebuah list di mana semua penambahan elemen di buat di ujung belakang dan penghapusan elemen list di buat di depan.