

## Tugas 4

### 1. linkedlist.java

```
01. public class LinkedList {
02.
03.     LinkedListNode head;
04.     LinkedListNode tail;
05.
06.     LinkedList() {
07.         this.head = null;
08.         this.tail = null;
09.     }
10.
11.     /* First set a Node named current into head
12.     * while current is not null, print current.data, set current = current.next
13.     * print end of line
14.     */
15.     void print() {
16.         LinkedListNode current = this.head;
17.         int i = 1;
18.         while (current != null) {
19.
20.             System.out.println(i+" "+current.data);
21.             current = current.next;
22.             i = i+1;
23.         }
24.         if(this.head == null){
25.             System.out.println("kosong");
26.         }else{
27.             System.out.println("");
28.         }
29.     }
30.
31.
32.     public int size() {
33.         int r = 0;
34.         LinkedListNode current = this.head;
35.         while (current != null) {
36.             current = current.next;
37.             r = r+1;
38.         }
39.         return r;
40.     }
41.
42.     /* if LinkedList is empty, set new_node as head and tail
43.     * if LinkedList is not empty, set tail.next into new_node, set
44.     * new_node.prev into tail, and make new_node a new tail
45.     */
46.     void push(LinkedListNode new_node) {
47.         if (this.head == null && this.tail == null) {
48.             head = new_node;
49.             tail = new_node;
50.         } else {
51.             tail.next = new_node;
52.             new_node.prev = tail;
53.             tail = new_node;
54.         }
55.     }
}
```

```

56.
57.
58.  /* declare a node named taken
59.  * if LinkedList is empty, set taken into null
60.  * if linkedList consists only one node, set taken = head, set both head
61.  and tail into null
62.  * if linkedList consists of two or more nodes, set taken = head,
63.  set head.next into new head, cut off all the link between taken and head
64.  * return taken
65.  */
66.  LinkedListNode qpop() {
67.      LinkedListNode taken = null;
68.      if (this.head == null && this.tail == null) {
69.          taken = null;
70.      } else if (this.head == this.tail) {
71.          taken = head;
72.          head = null;
73.          tail = null;
74.      } else {
75.          taken = head;
76.          //tail.prev.next = null;
77.          head = head.next;
78.      }
79.      return taken;
80.  }
81.
82.  /* declare a node named taken
83.  * if LinkedList is empty, set taken into null
84.  * if linkedList consists only one node, set taken = tail, set both head
85.  and tail into null
86.  * if linkedList consists of two or more nodes, set taken = tail,
87.  set tail.prev into new tail, cut off all the link between taken and tail
88.  * return taken
89.  */
90.  LinkedListNode spop() {
91.      LinkedListNode taken = null;
92.      if (this.head == null && this.tail == null) {
93.          taken = null;
94.      } else if (this.head == this.tail) {
95.          taken = tail;
96.          head = null;
97.          tail = null;
98.      } else {
99.          taken = tail;
100.          tail.prev.next = null;
101.          tail = tail.prev;
102.      }
103.      return taken;
104.  }

```

## 2.linkedlistnode.java

```

01.  public class LinkedListNode {
02.      LinkedListNode next;
03.      LinkedListNode prev;
04.      String data;
05.
06.      /*
07.      * Constructor set this.data into new_data
08.      * set this.prev into null
09.      * set this.next into */
10.      LinkedListNode(String new_data) {
11.          this.data = new_data;
12.          this.prev = null;
13.          this.next = null;
14.      }
15.
16.      /* set this.prev into other
17.      * if other is not null, set */
18.      void set_prev(LinkedListNode other) {
19.          this.prev = other;
20.          if (other != null) {
21.              other.next = this;
22.          }
23.      }
24.
25.      /* set this.next into other
26.      * if other is not null, set other.prev into this
27.      */
28.      void set_next(LinkedListNode other) {
29.          this.next = other;
30.          if (other != null) {
31.              other.prev = this;
32.          }
33.      }
34.  }

```

### 3. tugaskuh.java

```
01. import java.util.Scanner;
02.
03. public class tugaskuh {
04.     public static void main(String[] args) {
05.         Scanner yolo = new Scanner(System.in);
06.         Scanner yalo = new Scanner(System.in);
07.         LinkedList z = new LinkedList();
08.         System.out.println();
09.         int x = 0;
10.         do {
11.             System.out.println("=====");
12.             System.out.println("    antrian warnet");
13.             System.out.println("=====");
14.             System.out.println(
15.                 "Selamat datang di aplikasi sistem antian warnet, \npilih salah satu menu untuk melakuk
16.             System.out.println("=====");
17.             System.out.println(
18.                 "Menu :\n0.Lihat antrian \n1.Tambah nama di antrian \n2.hapus nama dari antrian \n3.kel
19.             System.out.println("=====");
20.             System.out.print("pilihan anda : ");
21.             String masuk = yolo.next();
22.             if (masuk.equals("0")) {
23.                 System.out.println("Nama dalam Antrian :");
24.                 z.print();
25.                 System.out.println("");
26.                 System.out.print("banyak yang mengantri : ");
27.                 System.out.print(z.size());
28.                 System.out.print("\n\n");
29.                 x = x + 1;
30.             } else if (masuk.equals("1")) {
31.                 System.out.print("nama yang akan masuk antrian : ");
32.                 String nama = yalo.next();
33.                 z.push(new LinkedListNode(nama));
34.                 x = x + 1;
35.             } else if (masuk.equals("2")) {
36.                 if (z.size() >= 1) {
37.                     z.print();
38.                     System.out.println("nama dalam antrian yang dihapus :");
39.                     System.out.println(z.pop().data);
40.                 } else if (z.size() == 0) {
41.                     System.out.println("Data kosong, tidak ada yang dihapus!!\nkembali ke menu utama");
42.                 }
43.                 x = x + 1;
44.             } else if (masuk.equals("3")) {
45.                 x = 0;
46.             } else {
47.                 System.out.println("\n\n!!!!!---ERROR---!!!!");
48.                 System.out.println("inputan anda : " + masuk);
49.                 System.out.println("ERROR : undefined input code *001*");
50.                 System.out.println("!!!!!---ERROR---!!!!");
51.                 x=0;
52.             }
53.         } while (x != 0);
54.         System.out.println("Anda Telah keluar dari sistem, Terimakasih!!");
55.     }
56. }
```

output :

output menu utama

```
budosen@budosen-pc:/mnt/b2c7efbf-ef52-437d-8ca7-e46ea581cbb
pertemuan 4/tugas$ javac *.java && java tugaskuh

=====
    antrian warnet
=====
Selamat datang di aplikasi sistem antian warnet,
pilih salah satu menu untuk melakukan proses isi antrian
=====
Menu :
0.Lihat antrian
1.Tambah nama di antrian
2.hapus nama dari antrian
3.keluar dari menu
=====
pilihan anda : █
```

output lihat antrian

```
=====
    antrian warnet
=====
Selamat datang di aplikasi sistem antian warnet,
pilih salah satu menu untuk melakukan proses isi antrian
=====
Menu :
0.Lihat antrian
1.Tambah nama di antrian
2.hapus nama dari antrian
3.keluar dari menu
=====
pilihan anda : 0
Nama dalam Antrian :
kosong

banyak yang mengantri : 0
```

output lihat antrian dan tambahkan antrian

```
=====
    antrian warnet
=====
Selamat datang di aplikasi sistem antian warnet,
pilih salah satu menu untuk melakukan proses isi antrian
=====
Menu :
0.Lihat antrian
1.Tambah nama di antrian
2.hapus nama dari antrian
3.keluar dari menu
=====
pilihan anda : 1
nama yang akan masuk antrian : Panji
=====
    antrian warnet
=====
Selamat datang di aplikasi sistem antian warnet,
pilih salah satu menu untuk melakukan proses isi antrian
=====
Menu :
0.Lihat antrian
1.Tambah nama di antrian
2.hapus nama dari antrian
3.keluar dari menu
=====
pilihan anda : 0
Nama dalam Antrian :
1. Panji

banyak yang mengantri : 1
```

Menghapus nama dari antrian

```
=====
pilihan anda : 1
nama yang akan masuk antrian : budosen
=====
    antrian warnet
=====
Selamat datang di aplikasi sistem antian warnet,
pilih salah satu menu untuk melakukan proses isi antrian
=====
Menu :
0.Lihat antrian
1.Tambah nama di antrian
2.hapus nama dari antrian
3.keluar dari menu
=====
pilihan anda : 0
Nama dalam Antrian :
1. Panji
2. budosen

banyak yang mengantri : 2
```

```
=====
    antrian warnet
=====
Selamat datang di aplikasi sistem antian warnet,
pilih salah satu menu untuk melakukan proses isi antrian
=====
Menu :
0.Lihat antrian
1.Tambah nama di antrian
2.hapus nama dari antrian
3.keluar dari menu
=====
pilihan anda : 2
1. Panji
2. budosen

nama dalam antrian yang dihapus :
Panji
```

```

=====
    antrian warnet
=====
Selamat datang di aplikasi sistem antian warnet,
pilih salah satu menu untuk melakukan proses isi antrian
=====
Menu :
0.Lihat antrian
1.Tambah nama di antrian
2.hapus nama dari antrian
3.keluar dari menu
=====
pilihan anda : 0
Nama dalam Antrian :
1. budosen

banyak yang mengantri : 1

```

keluar dari menu

```

=====
    antrian warnet
=====
Selamat datang di aplikasi sistem antian warnet,
pilih salah satu menu untuk melakukan proses isi antrian
=====
Menu :
0.Lihat antrian
1.Tambah nama di antrian
2.hapus nama dari antrian
3.keluar dari menu
=====
pilihan anda : 3
Anda Telah keluar dari sistem, Terimakasih!!

```

error handling sederhana

```

=====
    antrian warnet
=====
Selamat datang di aplikasi sistem antian warnet,
pilih salah satu menu untuk melakukan proses isi antrian
=====
Menu :
0.Lihat antrian
1.Tambah nama di antrian
2.hapus nama dari antrian
3.keluar dari menu
=====
pilihan anda : fhahfksdfnewdsadfweq

!!!!!---ERROR---!!!!
inputan anda : fhahfksdfnewdsadfweq
ERROR : undefined input code *001*
!!!!!---ERROR---!!!!
Anda Telah keluar dari sistem, Terimakasih!!

```