

## Bully Code

```
Bully.java      Ring.java
1 import java.util.Scanner;
2 public class Bully {
3
4     static boolean state[] = new boolean[5];
5     int coordinator;
6     public static void up(int up)
7     {
8         if(state[up-1]==true)
9         {
10             System.out.println("process"+up+"is already up");
11         }
12         else
13         {
14             state[up-1] = true;
15             System.out.println("process "+up+" held election");
16             for(int i=up;i<5;i++)
17             {
18                 System.out.println("election message sent from
19 process "+up+" to process"+(i+1));
20             }
21             for(int i=up+1;i<=5;i++)
22             {
23                 if(state[i-1]==true)
24                 {
25                     System.out.println("alive message send from
26 process "+i+" to process"+up);
27                     break;
28                 }
29             }
30         }
31     }
32 }

42
43 public static void mess(int mess)
44 {
45     if(state[mess-1]==true)
46     {
47         if(state[4]==true)
48         {
49             System.out.println("OK");
50         }
51         else
52         {
53             if(state[4]==false)
54             {
55                 System.out.println("process"+mess+"election");
56                 for(int i=mess;i<5;i++)
57                 {
58                     System.out.println("election send from
59 process"+mess+"to process "+(i+1));
60                 }
61                 for(int i=5;i>=mess;i--)
62                 {
63                     if(state[i-1]==true)
64                     {
65                         System.out.println("Coordinator message send
66 from process"+i+"to all");
67                         break;
68                     }
69                 }
70             }
71         }
72     }
73     else
74     {
75         System.out.println("Prccess"+mess+"is down");
76     }
77 }
```

## Bully Output

```
akashkulkarni@akash-kulkarni: ~/College/SEM 8/LP5/Assn-6
akashkulkarni@akash-kulkarni:~/College/SEM 8/LP5/Assn-6$ javac Bully.java
akashkulkarni@akash-kulkarni:~/College/SEM 8/LP5/Assn-6$ java Bully
5 active process are:
Process up = p1 p2 p3 p4 p5
Process 5 is coordinator
.....
1 up a process.
2.down a process
3 send a message
4.Exit
2
bring down any process.
5
.....
1 up a process.
2.down a process
3 send a message
4.Exit
3
which process will send message
2
process2election
election send from process2to process 3
election send from process2to process 4
election send from process2to process 5
Coordinator message send from process4to all
.....
1 up a process.
2.down a process
3 send a message
4.Exit
```

## Ring Code

```
1 import java.util.Scanner;
2
3 public class Ring {
4
5     int n, inactive_count;
6     int coordinator;
7     boolean[] process_state;
8
9     public Ring(int n) {
10         this.n = n;
11         this.inactive_count = 0;
12         this.process_state = new boolean[n];
13         // State all processes as active
14         for(int i = 0; i < n; i++) {
15             this.process_state[i] = true;
16         }
17         this.coordinator = n - 1;
18         System.out.println("Process " + n + " is set as initial
19         coordinator");
20     }
21     public void deactivate_process(int id) {
22         /*
23          * Input : Process ID
24          * Utility : Deactivate process
25          * Output : None
26          */
27         if(id > n || id < 0) {
28             System.out.println("Invalid ID");
29             return;
30         }
31         if(!process_state[id - 1]) {
32             System.out.println("Process already inactive");
33         } else {
34             process_state[id - 1] = false;
```

```

57 public void election(int id) {
58     /*
59     * Input : Initiator
60     * Utility : Hold election process to select coordinator
61     * Output : Coordinator id
62     */
63     if(this.inactive_count == this.n) {
64         System.out.println("All members inactive...");
65         System.out.println("Aborting election process...");
66         this.coordinator = -1;
67         return;
68     }
69     id = id - 1;
70     int current_coordinator = id;
71     int token = (id + 1) % n;
72     System.out.println("\nElection initiator : " + (id + 1));
73     // Election algorithm
74     while(token != id) {
75         System.out.println("Token at process " + (token + 1));
76         if(this.process_state[token]) {
77             if(token > current_coordinator) {
78                 current_coordinator = token;
79             }
80         }
81         token = (token + 1) % this.n;
82     }
83     System.out.println("Elected coordinator : " + (current_coordinator
+ 1));
84     this.coordinator = current_coordinator;
85 }

```

## Ring Output

```

akashkulkarni@akash-kulkarni: ~/College/SEM 8/LP5/Assn-6
akashkulkarni@akash-kulkarni:~/College/SEM 8/LP5/Assn-6$ java Ring
Enter number of processes:
3
Process 3 is set as initial coordinator
*****Menu*****
1. Deactivate a process
2. Ping coordinator
3. View Ring
4. Election
5. Exit
*****
Enter Choice :
2
Enter process ID for sender
2
Coordinator active
Sending message from process 2 to 3
Coordinator alive

*****Menu*****
1. Deactivate a process
2. Ping coordinator
3. View Ring
4. Election
5. Exit
*****
Enter Choice :
3
Active Ring members
1
2
3

```

```
*****Menu*****
1. Deactivate a process
2. Ping coordinator
3. View Ring
4. Election
5. Exit
*****
Enter Choice :
1
Enter process ID :
2
Process 2 deactivated

*****Menu*****
1. Deactivate a process
2. Ping coordinator
3. View Ring
4. Election
5. Exit
*****
Enter Choice :
3
Active Ring members
1
3

*****Menu*****
1. Deactivate a process
2. Ping coordinator
3. View Ring
4. Election
5. Exit
*****
```