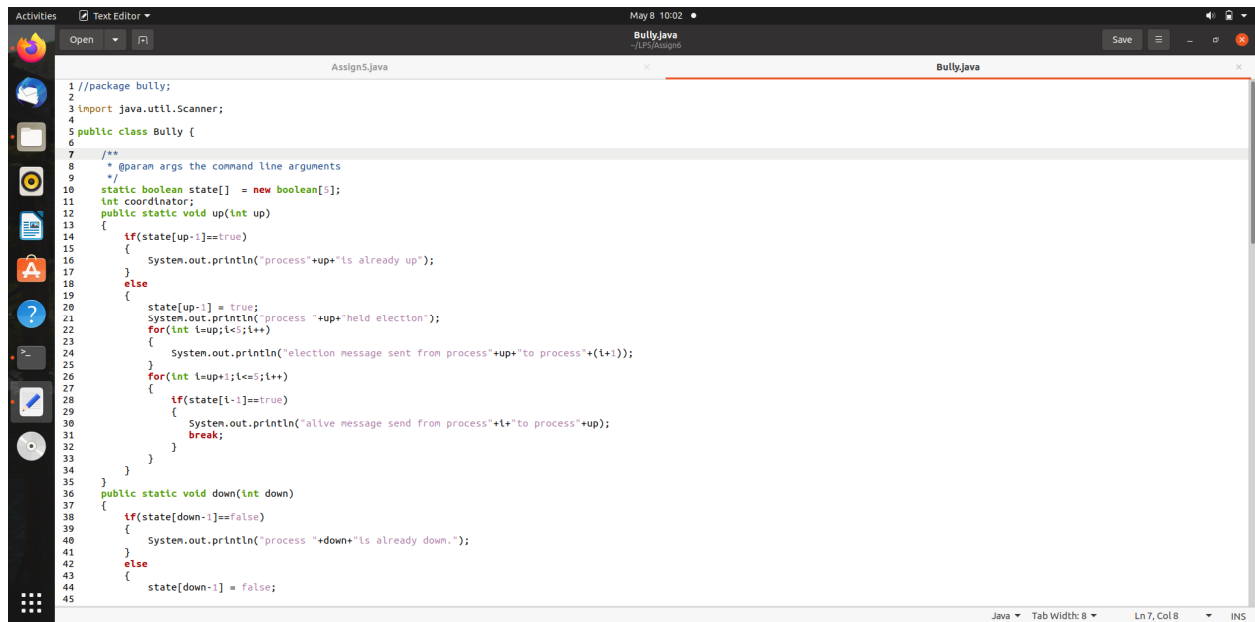


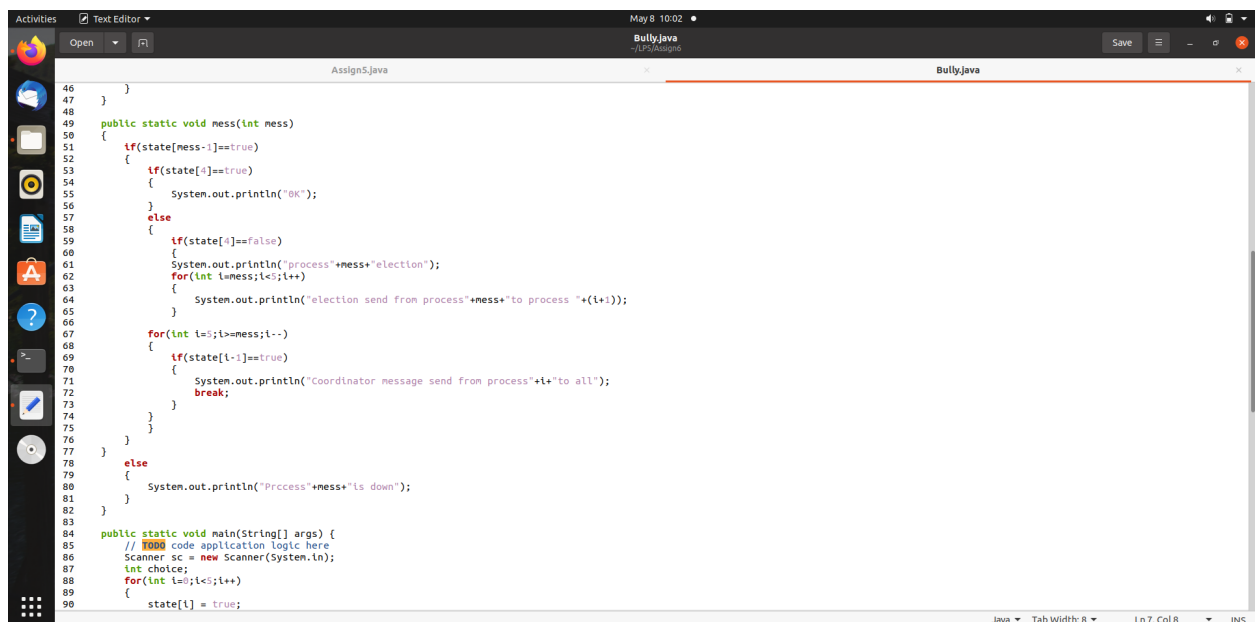
Name- Vaibhav Bichave

Roll no.-43209

CODE IMPLEMENTATION :



```
1 //package bully;
2
3 import java.util.Scanner;
4
5 public class Bully {
6
7     /**
8      * @param args the command line arguments
9      */
10    static boolean state[] = new boolean[5];
11    int coordinator;
12    public static void up(int up)
13    {
14        if(state[up-1]==true)
15        {
16            System.out.println("process "+up+" is already up");
17        }
18        else
19        {
20            state[up-1] = true;
21            System.out.println("process "+up+" held election");
22            for(int l=up;l<=5;l++)
23            {
24                System.out.println("election message sent from process "+up+" to process "+(l+1));
25            }
26            for(int l=up;l<=5;l++)
27            {
28                if(state[l-1]==true)
29                {
30                    System.out.println("alive message send from process "+l+" to process "+up);
31                    break;
32                }
33            }
34        }
35    }
36    public static void down(int down)
37    {
38        if(state[down-1]==false)
39        {
40            System.out.println("process "+down+" is already down.");
41        }
42        else
43        {
44            state[down-1] = false;
45        }
46    }
47 }
48
49 public static void mess(int mess)
50 {
51     if(state[mess-1]==true)
52     {
53         if(state[4]==true)
54         {
55             System.out.println("OK");
56         }
57     }
58     else
59     {
60         if(state[4]==false)
61         {
62             System.out.println("process "+mess+" election");
63             for(int l=mess;l<=5;l++)
64             {
65                 System.out.println("election send from process "+mess+" to process "+(l+1));
66             }
67             for(int l=5;l>=mess;l--)
68             {
69                 if(state[l-1]==true)
70                 {
71                     System.out.println("Coordinator message send from process "+l+" to all");
72                     break;
73                 }
74             }
75         }
76     }
77     else
78     {
79         System.out.println("Process "+mess+" is down");
80     }
81 }
82
83 public static void main(String[] args) {
84     // code application logic here
85     Scanner sc = new Scanner(System.in);
86     int choice;
87     for(int i=0;i<=5;i++)
88     {
89         state[i] = true;
90     }
91 }
```



```
46 }
47
48
49 public static void mess(int mess)
50 {
51     if(state[mess-1]==true)
52     {
53         if(state[4]==true)
54         {
55             System.out.println("OK");
56         }
57     }
58     else
59     {
60         if(state[4]==false)
61         {
62             System.out.println("process "+mess+" election");
63             for(int l=mess;l<=5;l++)
64             {
65                 System.out.println("election send from process "+mess+" to process "+(l+1));
66             }
67             for(int l=5;l>=mess;l--)
68             {
69                 if(state[l-1]==true)
70                 {
71                     System.out.println("Coordinator message send from process "+l+" to all");
72                     break;
73                 }
74             }
75         }
76     }
77     else
78     {
79         System.out.println("Process "+mess+" is down");
80     }
81 }
82
83 public static void main(String[] args) {
84     // code application logic here
85     Scanner sc = new Scanner(System.in);
86     int choice;
87     for(int i=0;i<=5;i++)
88     {
89         state[i] = true;
90     }
91 }
```

Activities Text Editor May 8 10:02 Bully.java ~/LP5/Assign6 Save

Assign5.java Bully.java

```
99 System.out.println("1 up a process.");
100 System.out.println("2 down a process.");
101 System.out.println("3 send a message");
102 System.out.println("4.Exit");
103 choice = sc.nextInt();
104 switch(choice)
105 {
106     case 1:
107     {
108         System.out.println("bring proces up");
109         int up = sc.nextInt();
110         if(up==5)
111         {
112             System.out.println("process 5 is co-ordinator");
113             state[4] = true;
114         }
115     }
116     else
117     {
118         up(up);
119     }
120 }
121 break;
122 case 2:
123 {
124     System.out.println("bring down any process.");
125     int down = sc.nextInt();
126     down(down);
127 }
128 break;
129 case 3:
130 {
131     System.out.println("which process will send message.");
132     int mess = sc.nextInt();
133     mess(mess);
134 }
135 break;
136 }
137 }
138 }
139 while(choice!=4);
140 }
141 }
142 }
143 }
```

Java Tab Width: 8 Ln 7, Col 8 INS

Activities Text Editor May 8 10:03 Ring.java ~/LP5/Assign6 Save

Assign5.java Bully.java Ring.java

```
1 import java.util.Scanner;
2
3
4 public class Ring {
5
6     public static void main(String[] args) {
7
8         // TODO Auto-generated method stub
9
10        int temp, i, j;
11        char str[] = new char[10];
12        Rr proc[] = new Rr[10];
13
14        // object initialisation
15        for (i = 0; i < proc.length; i++)
16            proc[i] = new Rr();
17
18        // scanner used for getting input from console
19        Scanner in = new Scanner(System.in);
20        System.out.println("Enter the number of process : ");
21        int num = in.nextInt();
22
23        // getting input from users
24        for (i = 0; i < num; i++) {
25            proc[i].index = i;
26            System.out.println("Enter the id of process : ");
27            proc[i].id = in.nextInt();
28            proc[i].state = "active";
29            proc[i].f = 0;
30        }
31
32
33        // sorting the processes from on the basis of id
34        for (i = 0; i < num - 1; i++) {
35            for (j = 0; j < num - i; j++) {
36                if (proc[j].id > proc[j + 1].id) {
37                    temp = proc[j].id;
38                    proc[j].id = proc[j + 1].id;
39                    proc[j + 1].id = temp;
40                }
41            }
42        }
43
44        for (i = 0; i < num; i++) {
45            System.out.println("id of process : " + i + " is " + proc[i].id);
46        }
```

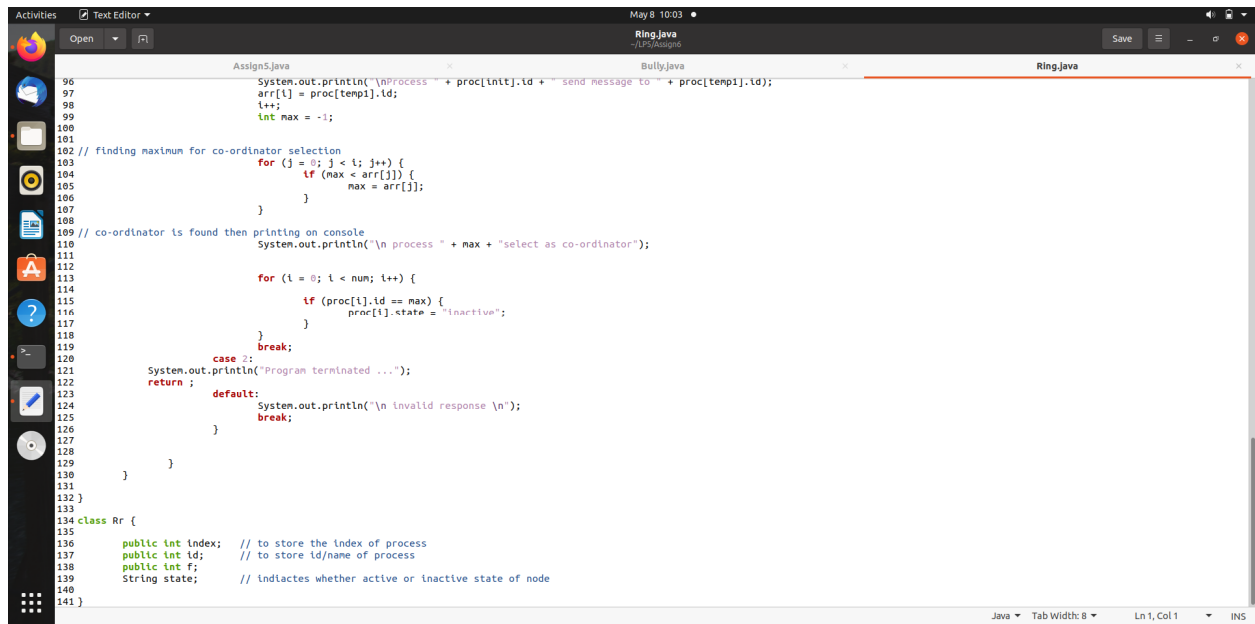
Java Tab Width: 8 Ln 1, Col 1 INS

```
Activities Text Editor May 8 10:03
Ring.java
Assign5.java Bully.java Ring.java
Save

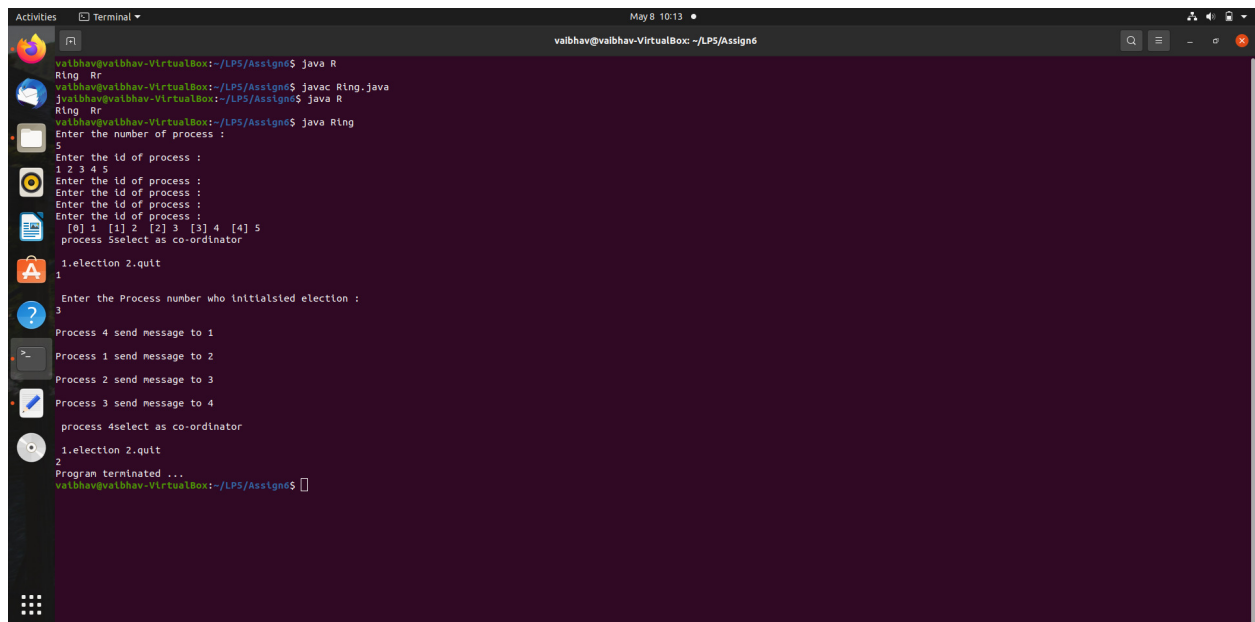
50
51
52 int init;
53 int ch;
54 int temp1;
55 int temp2;
56 int ch1;
57 int arr[] = new int[10];
58
59 proc[num - 1].state = "inactive";
60
61 System.out.println("\n process " + proc[num - 1].id + "select as co-ordinator");
62
63 while (true) {
64     System.out.println("\n 1.election 2.quit ");
65     ch = ln.nextInt();
66
67     for (i = 0; i < num; i++) {
68         proc[i].f = 0;
69     }
70
71     switch (ch) {
72     case 1:
73         System.out.println("\n Enter the Process number who initialised election : ");
74         init = ln.nextInt();
75         temp2 = init;
76         temp1 = init + 1;
77         i = 0;
78
79         while (temp2 != temp1) {
80             if ("active".equals(proc[temp1].state) && proc[temp1].f == 0) {
81                 System.out.println("\nProcess " + proc[init].id + " send message to " + proc[temp1].id);
82                 proc[temp1].f = 1;
83                 init = temp1;
84                 arr[i] = proc[temp1].id;
85                 i++;
86             }
87             if (temp1 == num) {
88                 temp1 = 0;
89             } else {
90                 temp1++;
91             }
92         }
93     }
94 }
95
```

```
Activities Text Editor May 8 10:03
Ring.java
Assign5.java Bully.java Ring.java
Save

94
95
96 System.out.println("\nProcess " + proc[init].id + " send message to " + proc[temp1].id);
97 arr[i] = proc[temp1].id;
98 i++;
99 int max = -1;
100
101 // Finding maximum for co-ordinator selection
102 for (j = 0; j < i; j++) {
103     if (max < arr[j]) {
104         max = arr[j];
105     }
106 }
107
108 // co-ordinator is found then printing on console
109 System.out.println("\n process " + max + "select as co-ordinator");
110
111 for (i = 0; i < num; i++) {
112     if (proc[i].id == max) {
113         proc[i].state = "inactive";
114     }
115 }
116 break;
117
118 case 2:
119     System.out.println("Program terminated ...");
120     return;
121 default:
122     System.out.println("\n invalid response \n");
123     break;
124 }
125 }
126 }
127 }
128 }
129 }
130 }
131 }
132 }
133
134 class Rr {
135
136     public int index; // to store the index of process
137     public int id; // to store id/name of process
138     public int f;
139     String state; // indicates whether active or inactive state of node
140 }
```



```
196      System.out.println("\nProcess " + proc[init].id + " send message to " + proc[tempi].id);
197      arr[l] = proc[tempi].id;
198      l++;
199      int max = -1;
200
201
202 // finding maxmun for co-ordinator selection
203      for (j = 0; j < l; j++) {
204          if (max < arr[j]) {
205              max = arr[j];
206          }
207      }
208
209 // co-ordinator is found then printing on console
210      System.out.println("\n process " + max + "select as co-ordinator");
211
212
213      for (i = 0; i < num; i++) {
214          if (proc[i].id == max) {
215              proc[i].state = "inactive";
216          }
217          break;
218      }
219
220      System.out.println("Program terminated ...");
221      return ;
222      default:
223          System.out.println("\n invalid response \n");
224          break;
225      }
226  }
227  }
228
229  }
230  }
231  }
232  }
233
234 class Rr {
235
236     public int index; // to store the index of process
237     public int id; // to store id/name of process
238     public int f;
239     String state; // indlates whether active or inactive state of node
240
241 }
```



```
vaibhav@vaibhav-VirtualBox: ~/LPS/Assign6$ java R
Ring Rr
vaibhav@vaibhav-VirtualBox:~/LPS/Assign6$ javac Ring.java
vaibhav@vaibhav-VirtualBox:~/LPS/Assign6$ java R
Ring Rr
vaibhav@vaibhav-VirtualBox:~/LPS/Assign6$ java Ring
Enter the number of process :
5
Enter the id of process :
1 2 3 4 5
Enter the id of process :
Enter the id of process :
Enter the id of process :
Enter the id of process :
[0] 1 [1] 2 [2] 3 [3] 4 [4] 5
process 5select as co-ordinator
1.election 2.quit
1
Enter the Process number who initialisted election :
3
Process 4 send message to 1
Process 1 send message to 2
Process 2 send message to 3
Process 3 send message to 4
process 4select as co-ordinator
1.election 2.quit
2
Program terminated ...
vaibhav@vaibhav-VirtualBox:~/LPS/Assign6$
```

```
Activities Terminal May 8 10:12 vaibhav@vaibhav-VirtualBox: ~/LP5/Assign6

vaibhav@vaibhav-VirtualBox:~/LP5/Assign6$ cd ../Assign6/
vaibhav@vaibhav-VirtualBox:~/LP5/Assign6$ javac *.java
vaibhav@vaibhav-VirtualBox:~/LP5/Assign6$ ls
Assign6.pdf Bully.class Bully.java Ring.class Ring.java Rr.class
vaibhav@vaibhav-VirtualBox:~/LP5/Assign6$ 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.
3
.....
1 up a process.
2 down a process
3 send a message
4 Exit
1
bring proces up
2
process2is already up
.....
1 up a process.
2 down a process
3 send a message
4 Exit
1
bring proces up
3
process 3held election
election message sent from process3to process4
election message sent from process3to process5
alive message send from process3to process3
.....
1 up a process.
2 down a process
3 send a message
4 Exit
3
which process will send message
vaibhav
Exception in thread "main" java.util.InputMismatchException
    at java.base/java.util.Scanner.throwFor(Scanner.java:939)
    at java.base/java.util.Scanner.next(Scanner.java:1594)
    at java.base/java.util.Scanner.next(Scanner.java:1465)
```