DC PRAC 6

CODE:

```
import java.util.*;
public class Mutex {
 public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int opt0, opt1;
        int p1 = 1;
        int p2 = 2;
        int p3 = 3;
        int flag = 0;
        int cs = 0;
        Queue < Integer > q = new LinkedList < > ();
        do {
        System.out.println("....menu...");
        System.out.println("1.Request the critical section");
        System.out.println("2.Release the critical section");
        System.out.println("3. Exit");
        opt0 = sc.nextInt();
        switch (opt0) {
        case 1: {
        System.out.println("Select the process.");
        System.out.println("1.p1");
        System.out.println("2.p2");
        System.out.println("3.p3");
        opt1 = sc.nextInt();
        switch (opt1) {
        case 1: {
        if (flag == 0) {
        cs = 1;
        flag = 1;
        } else {
        System.out.println("process p ^ n + cs + ^ n is already in critical section.");
        q.add(p1);
        }
        System.out.println("System Status:");
        System.out.println("critical section is occupied by:" + cs);
        System.out.println("process waiting is: " + q);
        break;
        }
        case 2: {
        if (flag == 0) {
        cs = 2;
        flag = 1;
        } else {
```

```
q.add(p2);
        }
        System.out.println("System Status:");
        System.out.println("critical section is occupied by:" + cs);
        System.out.println("process waiting is: " + q);
        break;
        }
        case 3: {
        if (flag == 0) {
        cs = 3;
        flag = 1;
        } else {
        System.out.println("process p" + cs + "is already in critical section.");
        q.add(p3);
        }
        System.out.println("System Status:");
        System.out.println("critical section is occupoied by:" + cs);
        System.out.println("process waiting is: " + q);
        break;
        }
        }
        break;
        }
        case 2: {
        System.out.println(" process p" + cs + "is removed from section.");
        if (q.isEmpty()) {
        cs = q.peek();
        q.remove();
        System.out.println("System status:");
        System.out.println("Critical Section occupied by p" + cs);
        } else {
        System.out.println("No Process is waiting in the queue");
        flag = 0;
        }
        case 3: {
        break;
        }
        }
        }
        while (3 != opt0);
}
}
```

System.out.println("process p" + cs + "is already in critical section.");

OUTPUT:

```
ax-508@ax508-OptiPlex-3050:~/Desktop$ java Mutex
....menu...
1.Request the critical section
2.Release the critical section
3. Exit
1
Select the process.
1.p1
2.p2
3.p3
1
System Status:
critical section is occupoied by:1
process waiting is: []
....menu...
1.Request the critical section
2.Release the critical section
3. Exit
Select the process.
1.p1
2.p2
3.p3
process plis already in critical section.
System Status:
critical section is occupoied by:1
process waiting is: [3]
....menu...
1.Request the critical section
2.Release the critical section
3. Exit
process plis removed from section.
No Process is waiting in the queue
....menu...
1.Request the critical section
2.Release the critical section
```