Code:

```java
package distributed_system_os.assignment2;
import java.util.ArrayList;
import java.util.Set;
import java.util.concurrent.ConcurrentHashMap;

public class Main implements Runnable{

    enum State { RELEASED, WANTED, HELD };

    private int pid;
    private int n;
    private int clock = 0;
    private ArrayList<Main> processes;
    private int requestTimeStamp = -1;
    private State state = State.RELEASED;
    private int replyCount = 0;
    private Set<Integer> deferredReplies = ConcurrentHashMap.newKeySet();
    public Main(int pid, int n, ArrayList<Main> processes) {
        this.pid = pid;
        this.n = n;
        this.processes = processes;
    }

    public void run() {
        System.out.println("Process " + this.pid + " started.");
        try {
            Thread.sleep(this.pid * 1000);
            this.requestCriticalSection();
    } catch (Exception e) {
        System.out.println(e);
    }

}

private synchronized void incrementClock(int receivedTimestamp) {
    this.clock = Math.max(this.clock, receivedTimestamp) + 1;
}

private void requestCriticalSection() {
    synchronized(this) {
        this.clock++;
        this.state = State.WANTED;
        this.requestTimeStamp = clock;
        System.out.println("Process " + pid + " requests CS at time " + clock);
    }
    for(Main p : processes) {
        if(p.pid != this.pid){
        p.receiveRequest(this.pid, this.requestTimeStamp);
        }
    }

    this.waitForReply();

    this.enterCriticalSection();

}
```

```
    private void receiveRequest(int frompid, int fromTimeStamp) {
        synchronized(this) {
            this.incrementClock(fromTimeStamp);
            System.out.println("Process " + this.pid + " received REQUEST from " + frompid + "
with timestamp " + fromTimeStamp + ", local clock = " + clock);
            boolean deferReply = false;
            if(this.state == State.HELD) {
                deferReply = true;
            }
            else if(this.state == State.WANTED && this.requestTimeStamp < fromTimeStamp ||
this.state == State.WANTED && this.requestTimeStamp == fromTimeStamp && this.pid <
frompid) {
                deferReply = true;
            }
            else if(this.state == State.RELEASED) {
                deferReply = false;
            }

            if (deferReply) {
                this.deferredReplies.add(frompid);
                System.out.println("Process " + this.pid + " defers reply to " + frompid);
            } else {
                sendReply(frompid);
            }
        }
    }

    private void sendReply(int frompid) {
        System.out.println("Process " + pid + " sending REPLY to " + frompid);
        for(Main p : this.processes) {
```

```
        if(p.pid == frompid) {
            p.receiveReply(this.pid);
            break;
        }
    }
}

private void receiveReply(int fromPid) {
    synchronized (this) {
        replyCount++;
        System.out.println("Process " + this.pid + " received REPLY from " + fromPid);
    }
}

private void waitForReply() {
    while(true) {
        synchronized(this){
        if(this.replyCount == this.n - 1) {
            break;
        }
    }
    }
}

private void enterCriticalSection() {
    synchronized(this) {
    System.out.println("Process " + this.pid + " Entering Critical Section");
    this.state = State.HELD;
    }
    try {
        Thread.sleep(2000);
    }
    catch(Exception e) {
        System.out.println(e);
    }
    this.exitCriticalSection();
}

private void exitCriticalSection() {
    synchronized(this) {
    System.out.println("Process " + this.pid + " Exiting Critical Section");
    this.state = State.RELEASED;
    for(Integer pid : this.deferredReplies) {
        this.sendReply(pid);
    }
    this.deferredReplies.clear();
    }
}

public static void main(String[] args) {
    int n = 5;
    ArrayList<Main> processes = new ArrayList<>();

    for(int i = 0; i < n; i++) {
        processes.add(new Main(i, n, processes));
    }

    for(Main process : processes) {
            new Thread(process).start();
        }
    }
}
```

Output set 1:

```
Process 4 started.
Process 2 started.
Process 3 started.
Process 1 started.
Process 0 started.
Process 6 started.
Process 5 started.
Process 0 requests CS at time 1
Process 1 received REQUEST from 0 with timestamp 1, local clock = 2
Process 1 sending REPLY to 0
Process 0 received REPLY from 1
Process 2 received REQUEST from 0 with timestamp 1, local clock = 2
Process 2 sending REPLY to 0
Process 0 received REPLY from 2
Process 3 received REQUEST from 0 with timestamp 1, local clock = 2
Process 3 sending REPLY to 0
Process 0 received REPLY from 3
Process 4 received REQUEST from 0 with timestamp 1, local clock = 2
Process 4 sending REPLY to 0
Process 0 received REPLY from 4
Process 5 received REQUEST from 0 with timestamp 1, local clock = 2
Process 5 sending REPLY to 0
Process 0 received REPLY from 5
Process 6 received REQUEST from 0 with timestamp 1, local clock = 2
Process 6 sending REPLY to 0
Process 0 received REPLY from 6
Process 0 Entering Critical Section
Process 1 requests CS at time 3
Process 0 received REQUEST from 1 with timestamp 3, local clock = 4
Process 0 defers reply to 1
Process 2 received REQUEST from 1 with timestamp 3, local clock = 4
Process 2 sending REPLY to 1
Process 1 received REPLY from 2
Process 3 received REQUEST from 1 with timestamp 3, local clock = 4
Process 3 sending REPLY to 1
Process 1 received REPLY from 3
Process 4 received REQUEST from 1 with timestamp 3, local clock = 4
Process 4 sending REPLY to 1
Process 1 received REPLY from 4
Process 5 received REQUEST from 1 with timestamp 3, local clock = 4
Process 5 sending REPLY to 1
Process 1 received REPLY from 5
Process 6 received REQUEST from 1 with timestamp 3, local clock = 4
Process 6 sending REPLY to 1
Process 1 received REPLY from 6
Process 2 requests CS at time 5
Process 0 Exiting Critical Section
Process 0 sending REPLY to 1
Process 1 received REPLY from 0
Process 0 received REQUEST from 2 with timestamp 5, local clock = 6
Process 0 sending REPLY to 2
Process 2 received REPLY from 0
Process 1 Entering Critical Section
Process 1 received REQUEST from 2 with timestamp 5, local clock = 6
Process 1 defers reply to 2
Process 3 received REQUEST from 2 with timestamp 5, local clock = 6
Process 3 sending REPLY to 2
Process 2 received REPLY from 3
Process 4 received REQUEST from 2 with timestamp 5, local clock = 6
Process 4 sending REPLY to 2
Process 2 received REPLY from 4
Process 5 received REQUEST from 2 with timestamp 5, local clock = 6
Process 5 sending REPLY to 2
Process 2 received REPLY from 5
Process 6 received REQUEST from 2 with timestamp 5, local clock = 6
Process 6 sending REPLY to 2
Process 2 received REPLY from 6
Process 3 requests CS at time 7
Process 0 received REQUEST from 3 with timestamp 7, local clock = 8
Process 0 sending REPLY to 3
Process 3 received REPLY from 0
Process 1 received REQUEST from 3 with timestamp 7, local clock = 8
Process 1 defers reply to 3
Process 2 received REQUEST from 3 with timestamp 7, local clock = 8
Process 2 defers reply to 3
Process 4 received REQUEST from 3 with timestamp 7, local clock = 8
Process 4 sending REPLY to 3
Process 3 received REPLY from 4
Process 5 received REQUEST from 3 with timestamp 7, local clock = 8
Process 5 sending REPLY to 3
Process 3 received REPLY from 5
Process 6 received REQUEST from 3 with timestamp 7, local clock = 8
Process 6 sending REPLY to 3
Process 3 received REPLY from 6
Process 4 requests CS at time 9
Process 0 received REQUEST from 4 with timestamp 9, local clock = 10
Process 0 sending REPLY to 4
Process 4 received REPLY from 0
Process 1 received REQUEST from 4 with timestamp 9, local clock = 10
Process 1 defers reply to 4
Process 2 received REQUEST from 4 with timestamp 9, local clock = 10
Process 2 defers reply to 4
Process 3 received REQUEST from 4 with timestamp 9, local clock = 10
Process 3 defers reply to 4
Process 5 received REQUEST from 4 with timestamp 9, local clock = 10
Process 5 sending REPLY to 4
Process 4 received REPLY from 5
Process 6 received REQUEST from 4 with timestamp 9, local clock = 10
Process 6 sending REPLY to 4
Process 4 received REPLY from 6
Process 1 Exiting Critical Section
Process 1 sending REPLY to 2
Process 2 received REPLY from 1
Process 1 sending REPLY to 3
Process 3 received REPLY from 1
Process 2 Entering Critical Section
Process 1 sending REPLY to 4
Process 4 received REPLY from 1
Process 5 requests CS at time 11
Process 0 received REQUEST from 5 with timestamp 11, local clock = 12
Process 0 sending REPLY to 5
Process 5 received REPLY from 0
Process 1 received REQUEST from 5 with timestamp 11, local clock = 12
Process 1 sending REPLY to 5
Process 5 received REPLY from 1
Process 2 received REQUEST from 5 with timestamp 11, local clock = 12
Process 2 defers reply to 5
Process 3 received REQUEST from 5 with timestamp 11, local clock = 12

Process 3 defers reply to 5

Process 4 received REQUEST from 5 with timestamp 11, local clock = 12

Process 4 defers reply to 5

Process 6 received REQUEST from 5 with timestamp 11, local clock = 12

Process 6 sending REPLY to 5

Process 5 received REPLY from 6

Process 6 requests CS at time 13

Process 0 received REQUEST from 6 with timestamp 13, local clock = 14

Process 0 sending REPLY to 6

Process 6 received REPLY from 0

Process 1 received REQUEST from 6 with timestamp 13, local clock = 14

Process 1 sending REPLY to 6

Process 6 received REPLY from 1

Process 2 received REQUEST from 6 with timestamp 13, local clock = 14

Process 2 defers reply to 6

Process 3 received REQUEST from 6 with timestamp 13, local clock = 14

Process 3 defers reply to 6

Process 4 received REQUEST from 6 with timestamp 13, local clock = 14

Process 4 defers reply to 6

Process 5 received REQUEST from 6 with timestamp 13, local clock = 14

Process 5 defers reply to 6

Process 2 Exiting Critical Section

Process 2 sending REPLY to 3

Process 3 received REPLY from 2

Process 2 sending REPLY to 4

Process 3 Entering Critical Section

Process 4 received REPLY from 2

Process 2 sending REPLY to 5

Process 5 received REPLY from 2

Process 2 sending REPLY to 6
Process 6 received REPLY from 2
Process 3 Exiting Critical Section
Process 3 sending REPLY to 4
Process 4 received REPLY from 3
Process 4 Entering Critical Section
Process 3 sending REPLY to 5
Process 5 received REPLY from 3
Process 3 sending REPLY to 6
Process 6 received REPLY from 3
Process 4 Exiting Critical Section
Process 4 sending REPLY to 5
Process 5 received REPLY from 4
Process 4 sending REPLY to 6
Process 5 Entering Critical Section
Process 6 received REPLY from 4
Process 5 Exiting Critical Section
Process 5 sending REPLY to 6
Process 6 received REPLY from 5
Process 6 Entering Critical Section
Process 6 Exiting Critical Section
```

Output set 2:

```
Process 4 started.
Process 3 started.
Process 2 started.
Process 1 started.
Process 0 started.
Process 0 requests CS at time 1
Process 1 received REQUEST from 0 with timestamp 1, local clock = 2
Process 1 sending REPLY to 0
Process 0 received REPLY from 1
Process 2 received REQUEST from 0 with timestamp 1, local clock = 2
Process 2 sending REPLY to 0
Process 0 received REPLY from 2
Process 3 received REQUEST from 0 with timestamp 1, local clock = 2
Process 3 sending REPLY to 0
Process 0 received REPLY from 3
Process 4 received REQUEST from 0 with timestamp 1, local clock = 2
Process 4 sending REPLY to 0
Process 0 received REPLY from 4
Process 0 Entering Critical Section
Process 1 requests CS at time 3
Process 0 received REQUEST from 1 with timestamp 3, local clock = 4
Process 0 defers reply to 1
Process 2 received REQUEST from 1 with timestamp 3, local clock = 4
Process 2 sending REPLY to 1
Process 1 received REPLY from 2
Process 3 received REQUEST from 1 with timestamp 3, local clock = 4
Process 3 sending REPLY to 1
Process 1 received REPLY from 3
Process 4 received REQUEST from 1 with timestamp 3, local clock = 4
Process 4 sending REPLY to 1
Process 1 received REPLY from 4
Process 2 requests CS at time 5
Process 0 received REQUEST from 2 with timestamp 5, local clock = 6
Process 0 defers reply to 2
Process 1 received REQUEST from 2 with timestamp 5, local clock = 6
Process 1 defers reply to 2
Process 3 received REQUEST from 2 with timestamp 5, local clock = 6
Process 3 sending REPLY to 2
Process 2 received REPLY from 3
Process 4 received REQUEST from 2 with timestamp 5, local clock = 6
Process 4 sending REPLY to 2
Process 2 received REPLY from 4
Process 0 Exiting Critical Section
Process 0 sending REPLY to 1
Process 1 received REPLY from 0
Process 0 sending REPLY to 2
Process 1 Entering Critical Section
Process 2 received REPLY from 0
Process 3 requests CS at time 7
Process 0 received REQUEST from 3 with timestamp 7, local clock = 8
Process 0 sending REPLY to 3
Process 3 received REPLY from 0
Process 1 received REQUEST from 3 with timestamp 7, local clock = 8
Process 1 defers reply to 3
Process 2 received REQUEST from 3 with timestamp 7, local clock = 8
Process 2 defers reply to 3
Process 4 received REQUEST from 3 with timestamp 7, local clock = 8
Process 4 sending REPLY to 3
Process 3 received REPLY from 4
Process 4 requests CS at time 9
Process 0 received REQUEST from 4 with timestamp 9, local clock = 10
Process 0 sending REPLY to 4
Process 4 received REPLY from 0
Process 1 received REQUEST from 4 with timestamp 9, local clock = 10
Process 1 defers reply to 4
Process 2 received REQUEST from 4 with timestamp 9, local clock = 10
Process 2 defers reply to 4
Process 3 received REQUEST from 4 with timestamp 9, local clock = 10
Process 3 defers reply to 4
Process 1 Exiting Critical Section
Process 1 sending REPLY to 2
Process 2 received REPLY from 1
Process 1 sending REPLY to 3
Process 2 Entering Critical Section
Process 3 received REPLY from 1
Process 1 sending REPLY to 4
Process 4 received REPLY from 1
Process 2 Exiting Critical Section
Process 2 sending REPLY to 3
Process 3 received REPLY from 2
Process 2 sending REPLY to 4
Process 4 received REPLY from 2
Process 3 Entering Critical Section
Process 3 Exiting Critical Section
Process 3 sending REPLY to 4
Process 4 received REPLY from 3
Process 4 Entering Critical Section
Process 4 Exiting Critical Section
```