Lab program - 10

c) Demonstrate inter process communication and deadlock

→ class A
{
  synchronized void foo (B b)
  {
    String name = Thread. currentThread (). getName();
    Sop(name + " entered A foo ");
    try {
        Thread. sleep (1000);
    }
    Catch (Exception e)
    {
        Sop(" A interrupted ");
    }
    sop (name + " trying to call B.last ()"),
        b. last();
  }
  void last ()
  {
    sop ("Inside A.last ");
  }
}

class B
{
  synchronized void bar (A a)
  {
    String name = Thread. currentThread (). getName();
    Sop (name + " entered B.bar "),
    try
    {
        Thread. sleep (1000),
    }
    Catch (Exception e)
    {
        sop(" B interrupted ");
    }
    sop (name + " trying to call A.last()"),
        a. last();
  }
  void last ()
  {
    sop (" Inside B.last ");
  }
}

```java
class Deadlock implements Runnable
{
    A a = new A();
    B b = new B();

    Deadlock()
    {
        Thread.currentThread().setName("Main thread");
        Thread t = new Thread(this, "Racing thread");

        t.start();
        a.foo(b);
        SOP("Back in main thread");
    }
    public void run()
    {
        b.bar(a);
        SOP("Back in other thread");
    }
    public static void main(String args[])
    {
        SOP("Name: Akshay S , USN: 1BM23CS022");
        new Deadlock();
    }
}
```

output:

Name: Akshay S , USN: 1BM23CS022
Main thread entered A.foo
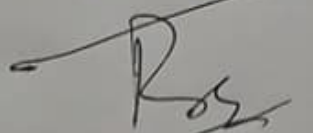Racing Thread entered B.bar
MainThread trying to call B.last()
Inside B.last
Back in main thread
Racing Thread trying to call A.last()
Inside A.last
Back in other thread

4/12/24