b) class A {
synchronized void foo(B b) {
String name = Thread.currentThread().getName();
System.out.println(name + "entered A.foo");
try {
Thread.sleep(1000);
}
catch (Exception e) {
System.out.println("A interrupted");
}
System.out.println(name + "trying to call B.last()");
b.last();
}
void last() {
System.out.println("Inside A.last");
}
}
class B {
synchronized void bar(A a) {
String name = Thread.currentThread().getName();
System.out.println(name + "entered B.bar");
try {
Thread.sleep(1000);
}
catch (Exception e) {
System.out.println("B interrupted");
void last() {
System.out.println("Inside B.last");
}
}

```
t1.start();
a.foo(b); //get lock on a in this thread
System.out.println("Back in main
thread");
}

public void run(){
b.bar(a); //get lock on b in other thread.
inside a.last
Back in main thread.
Racing Thread trying to call A.last()
inside A.last
Back in other thread
}

public static void main(String args[])
{
new Deadlock();
}
}
```

output

Main thread entered A.foo
Racing thread entered B.bar
main Thread trying to call B.last()
inside A.last
Back in main thread
Racing Thread trying to call A.last()
inside A.last
Back in other thread.

Shashank sp 1BM22CS256.

13/8/2024