PROGRAM 10:

Demonstrate Inter process Communication and deadlock

1. Demonstration of Inter process Communication

Observation Writeup:

classmate	8
Lab program - 10 Page Page	R
10) Demonstrate Interprocess communecation and	
deadlock	
Interprocess communication.	
Class QE 200 tranger	
bookean value set ? false;	
Value ser in purse,	
synchronized int get()?	
whele (I value set) {	
trus	
5.0.1 ("In Consumer waiting")?	
3 catch (Interrupted Exception e) {	
S. o. p ("Interrupted Exception caught)	2
50.p (Mact: 1+m)	
Valueset = false:	
3.0.1 / 10 throats b	
100 CCF GC 10	
octurn no	7
publication word small (steement of all a	
Synchronized voed at (P. + m/s	
Synchronized void put (int n) { whele (valueset)	
(008) wiretry Eward) seed to sail	
S. O. p / Produces 1220 + 12	
wait (10	
5 coatch ("Interrupted For to 18	
S-O. p ("tatermente France)?	Τ.
whele (valueset) trys S. D. P. ("Producer waiting"); wait (); f catch ("Interrupted Exception e) { S-0. P. ("Interrupte Exception caught"); their many terrupte Exception caught"	1,
	_
valueset = true.	_
5.0.p ("Put: "+n).	_
S.O. D Calotte	
5.0. p (°Intimate Consumer')?	

catch (Numberformat Excepteon e1)
£ flag=1;
out = "Number Format Exception" + e1;
repaint (1;
catch (A liffmitic Exception ea)
1 5 (ag = 1°
t tlag = 1 °; out = "Divide by O Exeptson 1" + e2; repaint ();
2 repaint of some
5.60 2000 Amosuno 14,0,5
public void paint (Graphics g)
if (tlag=zo)
g. drawstring (out, outRevilt. getXC)+
out Poult attale Ht 12 m + to 1 m + the
outkeault getwedth (), outkeaul get 4()+
outkesult getHeeght(1-8);
g. drawsterg Cout, 200, 200/.
5/ag = 0° (100)
t
public static void main (steing 57 og args)
3 Sagarage
Deves March day a new Dear and
Diriston Main I d'un 2 neur Divission Mains ();
dn. setsize (new Dimension (800, 400));
du set l'êtle ("Deverson Of Integers");
dm. set Visible (true);
Eccles ("Externibled Exception e)
CO. P ("Interrupte treepton tell
F S S S S S S S S S S S S S S S S S S S
the national
robustite - true
() Date
Page
S.O. p (* Press Control-C to gtop.');
S.O. p (Press Control-C to stop.);
Eve Produces (23) }
3 . 6= 4.000.
new Thread ("they Roduer") startes:

Soft copy of the program:

```
class Q {
int n;
boolean valueSet = false;
synchronized int get() {
while(!valueSet)
try {
System.out.println("\nConsumer waiting\n");
wait();
} catch(InterruptedException e) {
System.out.println("InterruptedException caught");
System.out.println("Got: " + n);
valueSet = false;
System.out.println("\nIntimate Producer\n");
notify();
return n;
synchronized void put(int n) {
while(valueSet)
try {
System.out.println("\nProducer waiting\n");
wait();
} catch(InterruptedException e) {
System.out.println("InterruptedException caught");
this.n = n;
valueSet = true;
System.out.println("Put: " + n);
System.out.println("\nIntimate Consumer\n");
notify();
class Producer implements Runnable {
Q q;
Producer(Q q) {
this.q = q;
new Thread(this, "Producer").start();
public void run() {
int i = 0;
while(i<15) {
q.put(i++);
```

```
class Consumer implements Runnable {
Q q;
Consumer(Q q) {
this.q = q;
new Thread(this, "Consumer").start();
public void run() {
int i=0;
while(i<15) {
int r=q.get();
System.out.println("consumed:"+r);
i++;
class PCFixed {
public static void main(String args[]) {
Q q = new Q();
new Producer(q);
new Consumer(q);
System.out.println("Press Control-C to stop.");
```

Output:

C:\Users\satis\OneDrive\Documents\ooj_lab>javac PCFixed.java C:\Users\satis\OneDrive\Documents\ooj_lab>java PCFixed Press Control-C to stop. Put: 0 Intimate Consumer Producer waiting Got: 0 Intimate Producer Put: 1 Intimate Consumer Producer waiting consumed:0 Got: 1 Intimate Producer consumed:1 Put: 2 Intimate Consumer Producer waiting Intimate Producer consumed:2 Put: 3

Intimate Consumer Producer waiting Got: 3 Intimate Producer consumed:3 Put: 4 Intimate Consumer Producer waiting Got: 4 Intimate Producer consumed:4 Put: 5 Intimate Consumer Producer waiting Got: 5 Intimate Producer consumed:5 Put: 6 Intimate Consumer Producer waiting

Intimate Producer consumed:6
Put: 7 Intimate Consumer Producer waiting Got: 7 Intimate Producer consumed:7 Put: 8 Intimate Consumer **Producer** waiting Got: 8 Intimate Producer consumed:8 Put: 9 Intimate Consumer Producer waiting Intimate Producer consumed:9 Put: 10

Intimate Producer consumed:10 Put: 11 Intimate Consumer Producer waiting Got: 11 Intimate Producer consumed:11 Put: 12 Intimate Consumer Producer waiting Got: 12 Intimate Producer consumed:12 Put: 13 Intimate Consumer **Producer** waiting Got: 13 Intimate Producer consumed:13 Put: 14 Intimate Consumer

2.Demonstration of Deadlock

Observation Writeup:

	Flower brow order
200	Demonstration of deadlock. class A E (01)
	class A ? (3125) s sales
	q. put(etf); 3
	synchronized void for (Bb)
	synchronized void foo (Bb) E String name = Throad. current Thread().get Name();
	Manie (1°)
	5.0.p (name t "entered A-foo"); try? Threads[eep(1000];}
	Follows try & Threads [cep (1000]; 220)
	eatch { Exception e) { 5.0. p (AInterrupted);}
	* Y = P :23 × 11
(1)	b- [ast 1);
	b-(ast 1);
	public void resolve f
	synchronized void last () { 5.0. p ("Inside A. last
	synchronezed void last () { 5.0. p ("Inside A.lasi")}
	Pat r= q. qet ();
	c fass (B & bourseaux") q. 0.2
	sepachronized void bar (A a) 9
	synchronized void bar (A a) { String name = Thread. currentThread(). gettlamel;
	8.0.8 (name + "entered B. bar");
	try { Thread. sleep (1000);}
	coatch (Exception e) & S. O. p ("B Interrupted");}
	3-0.p (name + "trying to call A. last()"). a. last
1870 C3 8	13212 agains box of the colling
1 1 C3 /	synchronized void last () { 5.0.p ("Forside A. last");}
	The Moducer (2)

Date
Fuge
class Deadlock Proplements Runnable
Class readjoint tropies.
Peadlock () E Peadlock () E Peadlock () E
 De alloch (If
 Thread current Thread (). set Name ("Main Thread);
Thread current Thread (). set Name ("Main Thread"); Thread t = new Thread (-this, 'Racing Thread"); t-start () a. too (b). S.o.p ("Back on moin thread"); 3
7- start (1° a. 500(b):
S. o. o ("Back in main thread");
3
public void run() & b. bar(a); S. o. p (Back in other thread);
S. O. p (Back in other thread);
3
public static void main (steing [] args) É new Deadlock();
new Deadlock();
3
7

Soft copy of the program:

```
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(); }
      synchronized 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"); }
System.out.println(name + " trying to call A.last()"); a.last(); }
synchronized void last() { System.out.println("Inside A.last"); }
class Deadlock implements Runnable
 A = new A(); B b = new B();
 Deadlock( ) {
    Thread.currentThread().setName("MainThread");
    Thread t = new Thread(this, "RacingThread");
    t.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.
 System.out.println("Back in other thread");
public static void main(String args[]) { new Deadlock(); }
```

Output:

```
C:\Users\satis\OneDrive\Documents\ooj_lab>javac Deadlock.java
C:\Users\satis\OneDrive\Documents\ooj_lab>java Deadlock
MainThread entered A.foo
RacingThread entered B.bar
RacingThread trying to call A.last()
MainThread trying to call B.last()
^C
C:\Users\satis\OneDrive\Documents\ooj_lab>
```