//9.3Java program to solve Producer-Consumer problem using synchronization. import java.io.\*;

class Q

{

int n;

boolean valueset=false;

synchronized void get()

{

if(valueset==false)

try

{

wait();

}

catch(InterruptedException e)

{

System.out.println(e);

}

System.out.println("got"+n);

valueset=false;

notify();

}

synchronized void put(int n)

{

if(valueset==true)

try

{

wait();

}

catch(InterruptedException e)

{

System.out.println(e);

}

this.n=n;

valueset=true;

System.out.println("put"+n);

notify();

}

}

class Producer extends Thread {

Q q;

Producer(Q q)

{

this.q=q;

}

public void run()

{

int i=0;

while(true)

{

q.put(++i);

}

}

}

class Consumer extends Thread {

Q q;

Consumer(Q q)

{

this.q=q;

}

public void run()

{

while(true)

{

q.get();

}

}

}

class Syn

{

public static void main(String args[]) {

Q q=new Q();

Producer p=new Producer(q); Consumer c=new Consumer(q);

p.start();

c.start();

System.out.println("press control-c to stop"); }

}

Output:

