// Java program for the above approach

public class GFG {

// Starting counter

int counter = 1;

static int N;

// Function to print odd numbers

public void printOddNumber()

{

synchronized (this)

{

// Print number till the N

while (counter < N) {

// If count is even then print

while (counter % 2 == 0) {

// Exception handle

try {

wait();

}

catch (

InterruptedException e) {

e.printStackTrace();

}

}

// Print the number

System.out.print(counter + " ");

// Increment counter

counter++;

// Notify to second thread

notify();

}

}

}

// Function to print even numbers

public void printEvenNumber()

{

synchronized (this)

{

// Print number till the N

while (counter < N) {

// If count is odd then print

while (counter % 2 == 1) {

// Exception handle

try {

wait();

}

catch (

InterruptedException e) {

e.printStackTrace();

}

}

// Print the number

System.out.print(

counter + " ");

// Increment counter

counter++;

// Notify to 2nd thread

notify();

}

}

}

// Driver Code

public static void main(String[] args)

{

// Given Number N

N = 10;

// Create an object of class

GFG mt = new GFG();

// Create thread t1

Thread t1 = new Thread(new Runnable() {

public void run()

{

mt.printEvenNumber();

}

});

// Create thread t2

Thread t2 = new Thread(new Runnable() {

public void run()

{

mt.printOddNumber();

}

});

// Start both threads

t1.start();

t2.start();

}

}