```java
//David Schmidt
//CIS1154
//2-25-2017
//This program simulates a traffic queue.

import java.io.*;
import java.util.*;
import java.math.*;


public class TrafficSimulation {


public static void main (String[] args) throws IOException
{


/* This main method simulates a traffic queue
* See "Data Abstraction and Structures" by Headington
* and Riley (Heath publisher) for more information
*/


// Create four queue objects named North, South, East, and West.
LinkedQueue<Character> North = new LinkedQueue<Character>();
LinkedQueue<Character> South = new LinkedQueue<Character>();
LinkedQueue<Character> East = new LinkedQueue<Character>();
LinkedQueue<Character> West = new LinkedQueue<Character>();
int timeUnit; // A for loop control variable to simulate 3 second intervals
              // for an hour(1200 iterations).
int stoppedCars = 0; // The number of cars which didn't pass through the light.
int passedCars = 0; // The number of cars which passed through the intersection.
boolean EWLightIsGreen = true; // A boolean value which signaled when the light changed.
                         // If true, East and West traffic flowed.
                         // If false, North and South traffic flowed.
boolean timeToChangeLight = false; //true every 30 seconds
double arrivalProbability = 0.90; //The probability a car will arrive in a given lane.

for (timeUnit = 0; timeUnit < 1200; timeUnit++) //loop 1200 times
{

// Adjust simulation values
timeToChangeLight = (timeUnit % 90 == 0); //Happens every 30 seconds...could adjust

// Switch the light every simulated 30 or 90 seconds.
if (timeToChangeLight)
   EWLightIsGreen = !EWLightIsGreen;

//Simulate Intersection activity
if ((!East.isEmpty()) && EWLightIsGreen) //East can go through light if green;
otherwise, it can't
{
   East.dequeue();
   passedCars++;
}//if

else if ((!West.isEmpty()) && EWLightIsGreen) //West can go through light if green;
otherwise, it can't
{
   West.dequeue();
   passedCars++;
}//else if

else if((!North.isEmpty()) && !EWLightIsGreen) //North can go through light if green;
otherwise, it can't
{
   North.dequeue();
   passedCars++;;
}//else if
```

```
67
68  else if((!South.isEmpty()) && !EWLightIsGreen) //South can go through light if green;
    otherwise, it can't
69  {
70     South.dequeue();
71     passedCars++;
72  }//else if
73
74  //Simulate arriving cars
75  if (Math.random() <= arrivalProbability) //A car arrives
76  {
77    East.enqueue(new Character('C'));
78    West.enqueue(new Character('A'));
79    North.enqueue(new Character('R'));
80    South.enqueue(new Character('S'));
81  }//if
82
83  }//for
84
85  //Print out the results.
86  System.out.println("Number of stopped cars:
    "+(East.size()+West.size()+North.size()+South.size()));
87  System.out.println("Number of passed cars: "+ passedCars);
88
89  }//main
90
91  }//class
92
```