import java.util.LinkedList;

import java.util.Queue;

import java.util.Random;

import java.util.ArrayList;

import java.util.List;

class Customer {

private int arrivalTime;

private int orderTime;

public Customer(int arrivalTime) {

this.arrivalTime = arrivalTime;

Random random = new Random();

this.orderTime = random.nextInt(5) + 3;

}

public int getArrivalTime() {

return arrivalTime;

}

public int getOrderTime() {

return orderTime;

}

}

class Barista {

private boolean isAvailable;

private Customer currentCustomer;

private int timeToCompleteOrder;

public Barista() {

this.isAvailable = true;

this.currentCustomer = null;

this.timeToCompleteOrder = 0;

}

public boolean isAvailable() {

return isAvailable;

}

public void startOrder(Customer customer) {

this.isAvailable = false;

this.currentCustomer = customer;

this.timeToCompleteOrder = customer.getOrderTime();

}

public void update() {

if (!isAvailable) {

timeToCompleteOrder--;

if (timeToCompleteOrder <= 0) {

isAvailable = true;

currentCustomer = null;

}

}

}

}

public class Main {

public static void main(String[] args) {

int simulationTime = 60;

int arrivalRate = 5;

Random random = new Random();

Queue<Customer> customerQueue = new LinkedList<>();

List<Barista> baristas = new ArrayList<>();

baristas.add(new Barista());

int customersServed = 0;

int totalWaitTime = 0;

for (int time = 0; time < simulationTime; time++) {

if (random.nextInt(arrivalRate) == 0) {

Customer customer = new Customer(time);

customerQueue.add(customer);

System.out.println("Customer arrived at time: " + time + ". Queue length: " + customerQueue.size());

}

for (Barista barista : baristas) {

barista.update();

if (barista.isAvailable() && !customerQueue.isEmpty()) {

Customer customer = customerQueue.poll();

barista.startOrder(customer);

int waitTime = time - customer.getArrivalTime();

totalWaitTime += waitTime;

customersServed++;

System.out.println("Customer served at time: " + time + ". Wait time: " + waitTime);

}

}

}

System.out.println("\n--- Results ---");

System.out.println("Customers served: " + customersServed);

if (customersServed > 0) {

System.out.println("Average wait time: " + (double) totalWaitTime / customersServed);

} else {

System.out.println("Average wait time: 0 (No customers served)");

}

System.out.println("Customers left in queue: " + customerQueue.size());

}

}