BLG 503
PARALEL PROGRAMLAMA

PROJE RAPORU

Ege Yağ 504161510

Prof. Dr. Nadia Erdoğan

Istanbul Teknik Üniversitesi

1. GİRİŞ

1.1. Tanım

Bu projede, Java dilinde bir banka simulasyonu yapılmıştır. Müşteriler bir thread tarafından üretilip bir listeye koyulurken başka bir thread tarafından bu listeden çıkartılmış ve bir banka çalışanı ile eşleştirilmiş, böylece multithreaded bir yapı kullanılmıştır. Bu sayede müşterilerin bankada sırada beklemesi ve sırası geldiğinde bir müşteri temsilcisi ile bankacılık işlemlerini tamamlamaları simule edilmiştir.

1.2. Amaç

Projedeki temel amaç paralel programlama dersinde öğrenilen yaklaşımları somut bir problemin çözümünde kullanarak örneğin paylaşılan bir alana yazma okuma alanında oluşabilecek problemlere çözüm bulabilmek ve performans artışı sağlayabilmektir.

1.3. Çözüm

Proje bir Producer-Consumer mantığında gerçeklenmiştir ve paralel programlama mantığından faydalanılmıştır.

Numara veren cihazı temsilen bir thread ile bir yandan üretilen Customer tipinden objeler bir LinkedList içine koyulurken, banka çalışanını temsil eden başka bir thread ise bir tarafından bu listeden objeleri çekmektedir. Customer class ı ile müşteriler gerçeklenmiştir. Her müşterinin kendine özgü numarası, yapacağı işlemin tutacağı zaman, bankaya geldiği zaman ve öncelikli bir kullanıcı olup olmadığı bilgileri tutulmaktadır. Bir müşteri çalışan tarafından işleme alındığında Thread o müşterinin yapacağı işlemin tutacağı zaman kadar uyutularak bankacılık işlemi simule edilmiştir. Müşterilerin tutulduğu LinkliListeye erişim synchronized bloğu içinde ve lock olarak bekleyen kullanıcıların tutulduğu liste kullanılarak gerçeklenmiştir.

Senkronizasyonun sağlanmasını kolaylaştırmak için başlangıçta thread-safe bir tür olan BlockingQueue kullanmaya karar verilmiş olsa da,

sonrasında öncelikli kullanıcıların öne geçirilmesi gerekmesi sebebi ile sadece en üstteki değil belirli bir indeksteki elemanın alınması gerekliliği ve bu yapının queue yapısına uygun olmaması, bu veri yapısını linkli liste ile değiştirme ve senkronizasyonu kendim gerçeklemem konusunda değişikliğe sebep olmuştur.

Kullanıcıdan bazı parametreler alınmıştır. Kullanıcının belirttiği olasılıkta bankaya yeni müşteri gelmektedir ve belirtilen boyuttaki kuyruk limitinde müşteri beklediği simule edilmiştir.

Eğer müşteriler kuyruğu dolu görürse, bankadan ayrılmaktadırlar. Bankaya gelen, hizmet alabilen, alamayan her müşterinin, müşteri temsilcilerinin hangi müşterilere hangi zaman aralıklarında hizmet verdiği ve simülasyonun başından beri kaç saniye boş kaldıkları gibi istatistikler de simülasyon sonunda gösterilmektedir.

Öncelikli müşteri kısıtı aşağıdaki kod bloğu ile gerçeklenmiştir.

Listenin boyutuna bakılarak eğer bir adet müşteri varsa gelen müşteri seçilmiştir. Eğer boyut 1 den büyükse ve gelem müşteri öncelikli değilse sıradaki müşteriye bakılmıştır. Sıradaki müşterinin öncelikli olması durumunda, bekleyen müşteri listesinden çıkarılarak bu müşteriye hizmet verilmesi sağlanmıştır. Bu işlemi yaptıktan sonra `tookPreviousVipCustomerInstead` bayrağı 1 arttırılmıştır.

Öncelikli olmayan müşteriden bir sonra gelmiş olan müşterinin de öncelikli olmaması durumunda bir sonraki müşteriye bakılmıştır. Yine aynı şekilde bayrak kullanılmıştır.
Son durumda eğer bayrak 2 ye ulaştıysa sıra kendisinde olmasına rağmen 2 kez öncelikli müşterinin önüne geçirildiği anlamına gelmektedir ve bu durumda o anki müşteriye hizmet verilir ve bayrak sıfırlanmıştır.

Veya kendinden sonraki ve ondan sonraki müşterilerin de öncelikli olmaması durumunda yine o anki müşteriye hizmet verilir ve bayrak sıfırlanmıştır.

Consumer:

```
synchronized (customerQ) {
    // Look for the current customer if it is a vip poll her
    Customer c0 = customerQ.get(0);
    if (c0.getIfpriorityCustomer()){
        System.out.println("\tSu anki musteri oncelikli bir musteri.");
        customerQ.remove(0);
        return c0;
    }else if (customerQ.size()==1){
        System.out.println("\tSu anki musteri kuyruktaki tek musteri.");
        customerQ.remove(0);
        return c0;
    }else{
        //If not vip look for the previous
        Customer c1 = customerQ.get(1);
        if (c1.getIfpriorityCustomer() && tookPreviousVipCustomerInstead<2){
            //remove this from list and return it
            tookPreviousVipCustomerInstead ++;
            System.out.println("\tSu anki musteri oncelikli degil. Bir sonraki musteri oncelikli oldugu icin ona hizmet verilecek.");
            c1 = customerQ.remove(1);
            return c1;
        }else if (customerQ.size()>2 && customerQ.get(2).getIfpriorityCustomer() && tookPreviousVipCustomerInstead<2){
            //If not vip look for the second previous
            Customer c2 = customerQ.get(2);
            //remove this from list and return it
            tookPreviousVipCustomerInstead++;
            System.out.println("\tSu anki ve bir sonraki musteriler oncelikli degil. Bir sonraki musteri oncelikli oldugu icin ona hizmet verilecek.");
            c2 = customerQ.remove(2);
            return c2;
        }else if(tookPreviousVipCustomerInstead == 2){
            System.out.println("\tBu musteriden once 2 oncelikli musteri one gecirildi. Simdi bu musteriye hizmet verilecek...");
            customerQ.remove(0);
            tookPreviousVipCustomerInstead=0;
            return c0;
        }else{
            System.out.println("\tBu musteri ve sonraki 2 musteri oncelikli degil, bu musteriye hizmet verilecek...");
            customerQ.remove(0);
            tookPreviousVipCustomerInstead=0;
            return c0;
        }
    }
```

Producer:

```
public void insertCustomerQ(Customer customer)
{
    synchronized (customerQ) {
        customerQ.add(customer);
    }
}
```

2. TASARIM

Projede paralel programlama yaklaşımlarının uygulanabilmesi için Java “thread” sınıfından, ”Runnable” Interface inden ve aynı anda birden fazla threadin okuma yazma yaptığı bir listeyi gerçeklerken lock yapısından ve ”synchronised ” kod bloğundan yararlanılmıştır. Bu kod bloğu alternatif olarak, customer listesi yerine yaratılan bir lock objesi ile `synchronized (lock)` şeklinde locklanarak `lock.wait();` ve `lock.notify();` yapıları kullanarak da deneyimlenmiş ve aynı sonuç alınmıştır.

```
System.out.println("\tMusteri #" + simulation.customerIDCounter + " kuyrukta bekliyor.");
Thread t1 = new Thread(new Runnable(){
    public void run() {
        bank.insertCustomerQ( new Customer(simulation.customerIDCounter, simulation.transactionTime,
                                    simulation.currentTime,random.nextBoolean()));
        try {
            Thread.sleep(simulation.transactionTime);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }

});
t1.start();

t1.join();


 Thread t2 = new Thread(new Runnable(){
     public void run() {
         while (bank.numFreeEmployees() > 0 && bank.numWaitingCustomers() > 0) {
              Customer customer = bank.decideNextCustomerToBeServed();
              Employee Employee = bank.removeFreeEmployeeQ();
              Employee.freeToBusy(customer, simulation.currentTime);
              bank.insertBusyEmployeeQ(Employee);
              simulation.numServed++;

              System.out.println("\tMusteri #" + customer.getCustomerID() + " #"
                      + Employee.getEmployeeID() + "numarali calisan ile " + customer.getTransactionTime()
                      + " sure islem yapacak.");
          }

     }

 });


 t2.start();

 t2.join();
```

3. SONUÇ

Programın çıktılarının ekran görüntüleri aşağıda gösterilmiştir.

```
        === Simulasyon Parameterelerini giriniz ===

Simulasyon suresi giriniz (max 10000): 100
Musterinin maximum transaction suresini giriniz (max is 500): 20
Bankaya yeni musteri gelme olasiligini giriniz (0% < & <= 100%)  90
Musteri temsilcisi sayisini giriniz (max 10): 4
Maksimum bekleyebilecek musteri sayisini giriniz (max 50): 15
Random data olusturuluyor.

        === Simulasyon Basliyor ===

-------------------------------------------------------------
Su an zaman  : 1
Kuyruk : 0/15
        Musteri #1 15 sure kalmak uzere bankaya geldi.
        Musteri #1 kuyrukta bekliyor.
        Su anki musteri kuyruktaki tek musteri.
        Musteri #1 #1numarali calisan ile 15 sure islem yapacak.
-------------------------------------------------------------
Su an zaman  : 2
Kuyruk : 0/15
        Musteri #2 8 sure kalmak uzere bankaya geldi.
        Musteri #2 kuyrukta bekliyor.
        Su anki musteri oncelikli bir musteri.
        Musteri #2 #2numarali calisan ile 8 sure islem yapacak.
-------------------------------------------------------------
Su an zaman  : 3
Kuyruk : 0/15
        Musteri #3 4 sure kalmak uzere bankaya geldi.
        Musteri #3 kuyrukta bekliyor.
        Su anki musteri oncelikli bir musteri.
        Musteri #3 #3numarali calisan ile 4 sure islem yapacak.
-------------------------------------------------------------
Su an zaman  : 4
Kuyruk : 0/15
        Musteri #4 7 sure kalmak uzere bankaya geldi.
        Musteri #4 kuyrukta bekliyor.
        Su anki musteri kuyruktaki tek musteri.
        Musteri #4 #4numarali calisan ile 7 sure islem yapacak.
-------------------------------------------------------------
Su an zaman  : 5
Kuyruk : 0/15
        Musteri #5 19 sure kalmak uzere bankaya geldi.
        Musteri #5 kuyrukta bekliyor.
```

```
Su an zaman  : 6
Kuyruk : 1/15
        Yeni musteri gelmedi!
------------------------------------------------------------
Su an zaman  : 7
Kuyruk : 1/15
        Musteri #6 12 sure kalmak uzere bankaya geldi.
        Musteri #6 kuyrukta bekliyor.
        Musteri #3 islemini bitirdi.
        Calisan #3 artik bos.
        Su anki musteri oncelikli bir musteri.
        Musteri #5 #3numarali calisan ile 19 sure islem yapacak.
------------------------------------------------------------
Su an zaman  : 8
Kuyruk : 1/15
        Yeni musteri gelmedi!
------------------------------------------------------------
Su an zaman  : 9
Kuyruk : 1/15
        Musteri #7 15 sure kalmak uzere bankaya geldi.
        Musteri #7 kuyrukta bekliyor.
------------------------------------------------------------
Su an zaman  : 10
Kuyruk : 2/15
        Musteri #8 6 sure kalmak uzere bankaya geldi.
        Musteri #8 kuyrukta bekliyor.
        Musteri #2 islemini bitirdi.
        Calisan #2 artik bos.
        Su anki musteri oncelikli bir musteri.
        Musteri #6 #2numarali calisan ile 12 sure islem yapacak.
------------------------------------------------------------
Su an zaman  : 11
Kuyruk : 2/15
        Musteri #9 10 sure kalmak uzere bankaya geldi.
        Musteri #9 kuyrukta bekliyor.
        Musteri #4 islemini bitirdi.
        Calisan #4 artik bos.
        Su anki musteri oncelikli bir musteri.
        Musteri #7 #4numarali calisan ile 15 sure islem yapacak.
------------------------------------------------------------
Su an zaman  : 12
Kuyruk : 2/15
        Musteri #10 11 sure kalmak uzere bankaya geldi.
        Musteri #10 kuyrukta bekliyor.
```

```
Su an zaman  : 95
Kuyruk : 14/15
        Musteri #80 10 sure kalmak uzere bankaya geldi.
        Musteri #80 kuyrukta bekliyor.
        Musteri #34 islemini bitirdi.
        Calisan #3 artik bos.
        Musteri #55 islemini bitirdi.
        Calisan #1 artik bos.
        Bu musteriden once 2 oncelikli musteri one gecirildi. Simdi bu musteriye hizmet verilecek...
        Musteri #53 #3numarali calisan ile 10 sure islem yapacak.
        Su anki musteri oncelikli degil. Bir sonraki musteri oncelikli oldugu icin ona hizmet verilecek.
        Musteri #57 #1numarali calisan ile 16 sure islem yapacak.
----------------------------------------------------------------
Su an zaman  : 96
Kuyruk : 13/15
        Musteri #81 15 sure kalmak uzere bankaya geldi.
        Musteri #81 kuyrukta bekliyor.
        Musteri #56 islemini bitirdi.
        Calisan #4 artik bos.
        Su anki musteri oncelikli degil. Bir sonraki musteri oncelikli oldugu icin ona hizmet verilecek.
        Musteri #58 #4numarali calisan ile 8 sure islem yapacak.
----------------------------------------------------------------
Su an zaman  : 97
Kuyruk : 13/15
        Musteri #82 13 sure kalmak uzere bankaya geldi.
        Musteri #82 kuyrukta bekliyor.
----------------------------------------------------------------
Su an zaman  : 98
Kuyruk : 14/15
        Musteri #83 3 sure kalmak uzere bankaya geldi.
        Musteri #83 kuyrukta bekliyor.
----------------------------------------------------------------
Su an zaman  : 99
Kuyruk : 15/15
        Musteri #84 12 sure kalmak uzere bankaya geldi.
        Musteri kuyrugu dolu. Musteri #84 ayrildi...
----------------------------------------------------------------
Su an zaman  : 100
Kuyruk : 15/15
        Musteri #85 8 sure kalmak uzere bankaya geldi.
        Musteri kuyrugu dolu. Musteri #85 ayrildi...
```

```
        === Simulasyon raporu ===


        # Toplam gelen musteri sayisi                : 85
        # Islem yapamadan ayrilan musteri sayisi     : 26
        # Islem yapabilen musteri sayisi             : 44


------------------------------------------------------------------
        === Banka calisani bilgileri. ===

        # Bekleyen musteriler  : 15
        # Mesgul calisanlar    : 4
        # Bos calisanlar       : 0


        Toplam bekleme suresi   : 95
        Ortalama bekleme suresi : 6,33


------------------------------------------------------------------
        === Mesgul banka calisani bilgileri. ===

        Calisan ID                : 4
        Toplam bos zaman          : 3
        Toplam dolu zaman         : 97
        Toplam #musteri           : 10
        Ortalama transaction suresi : 9,70

        Calisan ID                : 3
        Toplam bos zaman          : 2
        Toplam dolu zaman         : 98
        Toplam #musteri           : 10
        Ortalama transaction suresi : 9,80

        Calisan ID                : 2
        Toplam bos zaman          : 1
        Toplam dolu zaman         : 99
        Toplam #musteri           : 11
        Ortalama transaction suresi : 9,00
```

```
        Calisan ID                : 1
        Toplam bos zaman          : 0
        Toplam dolu zaman         : 100
        Toplam #musteri           : 13
        Ortalama transaction suresi : 7,69



------------------------------------------------------------
        === Bos banka calisani bilgileri. ===

                Bos banka calisani yok.
```