

Problem javnog kupatila

Ognjen Čavić E2 161/2024

Novembar 2024

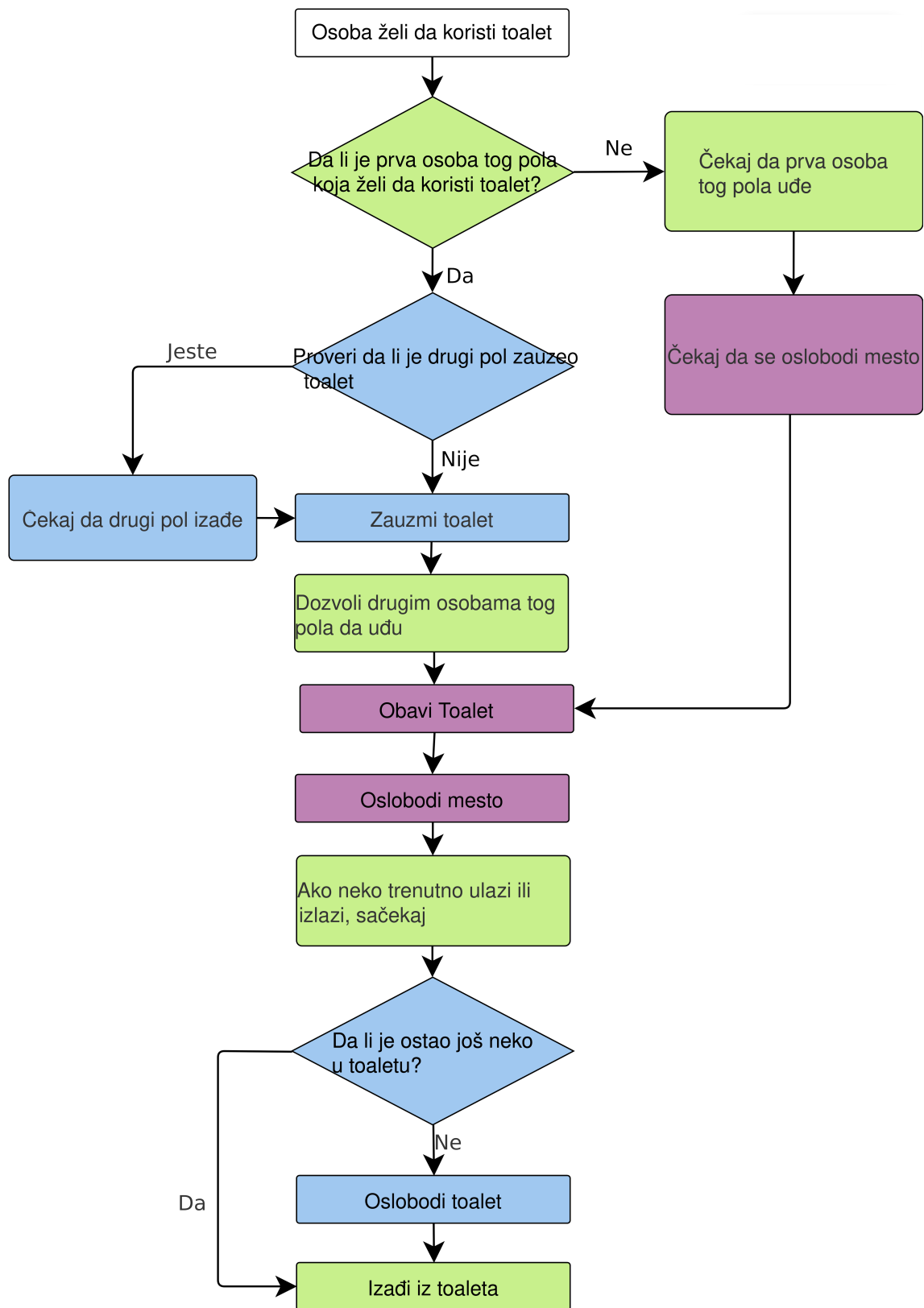
1 Opis problema

Problem javnog toaleta (engl. *Unisex Bathroom problem*) predstavlja problem sinhronizacije izvršavanja više niti. Javno kupatilo sa $N > 1$ mesta mogu da koriste i muškarci i žene sa ograničenjem da osobe različitog pola ne mogu istovremeno koristiti kupatilo. Ukoliko se svakoj niti "dodeli pol" potrebno je pomoću semafora obezbediti da se maksimalno N niti (osoba) istog pola nalazi u toaletu, tako da pol koji trenutno koristi kupatilo periodično menja, kako bi se izbeglo da jedan pol mora da čeka da sve osobe drugog pola obave toalet. Toalet ovde predstavlja određenu kritičnu sekciju koda u kojoj se može nalaziti maksimalno N niti.

2 Logika rešenja

Rešenje problema jeste jednostavno, jer sve što je potrebno je ograničiti broj osoba koja može da koristi toalet u bilo kom momentu, zabraniti ulaz osobama suprotnog pola. Međutim, implementacija tog rešenja pomoću semafora jeste malo komplikovanija. Na slici 1 je prikazan algoritam kroz koji svaka nit tj. osoba prolazi. Koraci obojeni istom bojom podrazumevaju ulazak ili izlazak iz sekcije pod istim semaforom.

Prvi semafor na koji osoba nailazi, obojen zelenom bojom, predstavlja proveru mogućnosti ulaska i ponaša se slično mutex-u. Prva osoba svakog od polova će proći ovaj semafor, proveriti da li je toalet prazan, ukoliko jeste, uzeće toalet za svoj pol tj. zauzeće semafor koji predstavlja sam toalet, koji je na slici 1 obojen plavom bojom. Svaka druga nit koja naiđe na semafor provere mogućnosti ulaska neće proći sve dok ga prva nit ne oslobodi što će se desiti kada taj pol zauzme toalet. To takođe znači da mora postojati dve instance ovog semafora tj. jedna za svaki pol. Sama mesta u toaletu su predstavljena semaforom koji mogu da prođu maksimalno N niti i na slici 1 je obojen ljubičastom bojom.



Slika 1: Šematski prikaz algoritma na koji svaka nit nailazi

3 Implementacija

3.1 Klasa Person

Klasa kojom se predstavlja osoba je samo struktura gde se čuvaju pol i neki broj koji služi da se u ispisu osobe mogu razlikovati, te jasnije pokazati dešavanja. Pol je predstavljen enumeracijom sa dva člana jer je to najjednostavnije za implementaciju ali i C# ima metode za pretvaranje u string što je korisno za ispis.

```
1  enum gender {
2      MALE,
3      FEMALE
4  };
5
6  internal class Person
7  {
8      public gender gender;
9      public int id;
10
11     public Person(gender gender, int id)
12     {
13         this.gender = gender;
14         this.id = id;
15     }
16 }
```

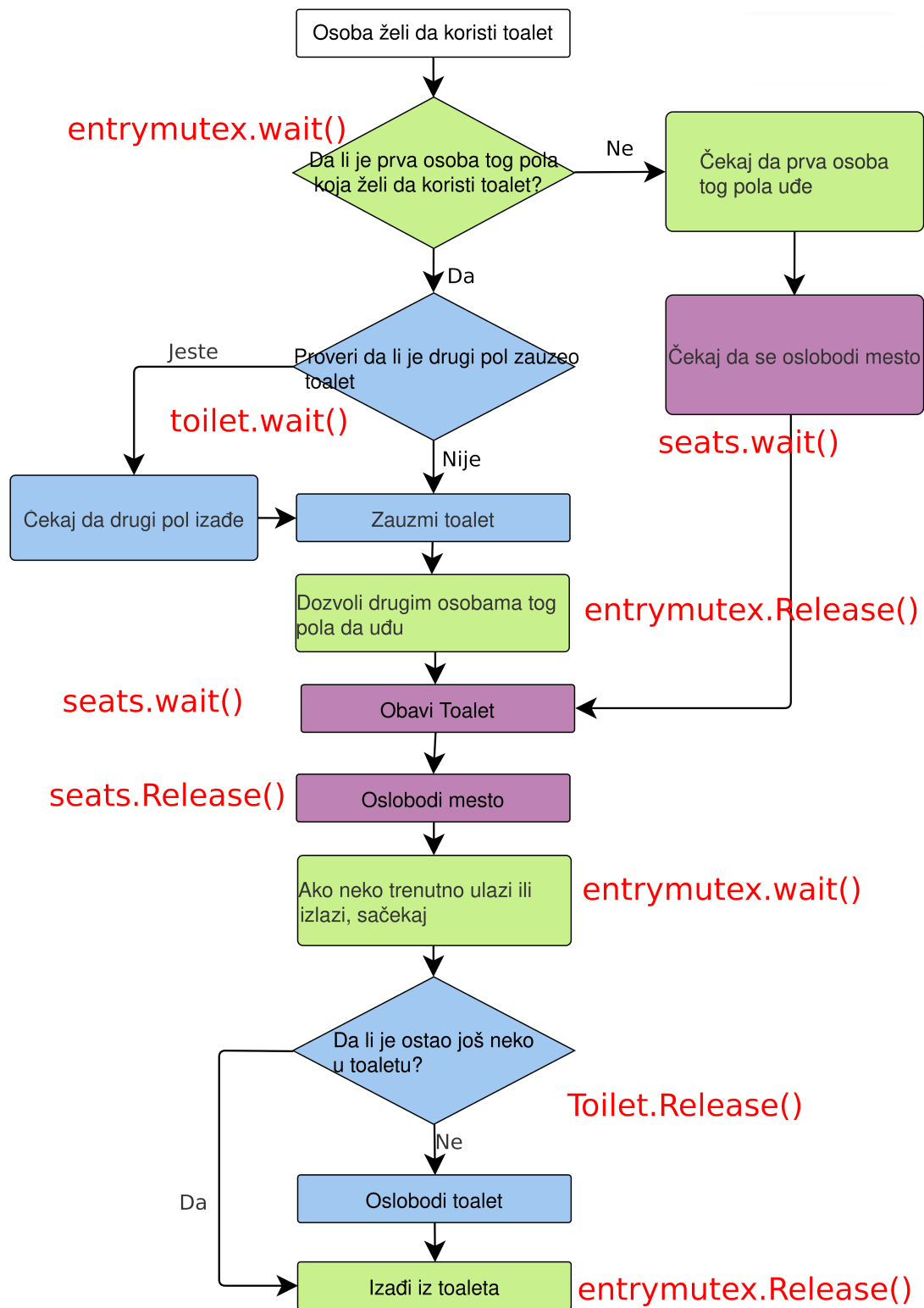
3.2 Klasa Toilet

U ovoj klasi je implementiran algoritam sa slike 1 i predstavlja srž ovog zadatka. Polja ove klase su svi semafori korišćeni u rešenju sa tim da je dodat i semafor poruke, koji dozvoljava samo jednoj niti da ispisuje na konzolu, što obezbeđuje smislen redosled poruka. Na primer, ako se desi da je muškarac završio sa toaletom, dok on ispiše poruku, može svašta da se desi i ispiše na ekran, kada se njegova poruka konačno ispiše, ispadne da program ne radi kako je namenjeno. Ovaj semafor ne sprečava pojavu da dok se piše poruka brojno stanje ljudi promeni. Preostalo polje ove klase su dva brojača, jedan za svaki pol, koji služe da se prati kada ulazi prva osoba i kada izlazi poslednja osoba. Metoda za ispis je vrlo jednostavna i sve što radi je da pre ispisivanja na ekran rezerviše semafor za poruke i čim se to završi ga oslobađa.

```
1  public static SemaphoreSlim toilet = new SemaphoreSlim(1);
2  public static SemaphoreSlim[] entryMutex = { new SemaphoreSlim(1), new
3      SemaphoreSlim(1), };
4  public static SemaphoreSlim seats = new SemaphoreSlim(Program.TOILET_CAPACITY);
5  public static SemaphoreSlim messageMutex = new SemaphoreSlim(1);
6  public static int[] cnt = { 0, 0 };
7
8  public static void print(string msg)
9  {
10     messageMutex.Wait();
11     Console.WriteLine(msg);
12     messageMutex.Release();
13 }
```

Algoritam sa slike 1 je implementiran u metodi **SimulateToilet** sa tim da mu se prosleđuje lista osoba i za svaku se stvara nova nit. Slika 2 prikazuje isto što i slika 1 ali je uz korake dodato koju metodu kog semafora one pozivaju. Rutine za svaki pol su identične sa tim da koriste drugačiji semafor za proveru mogućnosti ulaska, tako da se samo koristi indeks odgovarajućeg semafora.

```
1 public static void SimulateToilet(List<Person> people)
2 {
3     var stopwatch = new Stopwatch();
4     stopwatch.Start();
5     Parallel.ForEach(people, person =>
6     {
7         entryMutex[(int)person.gender].Wait();
8         if (++cnt[(int)person.gender] == 1)
9         {
10             toilet.Wait();
11             print($"{person.gender}S occupied the bathroom");
12         }
13         entryMutex[(int)person.gender].Release();
14
15         seats.Wait();
16         print($"{person.gender} {person.id} started using the toilet, Seats free:
17             {seats.CurrentCount}");
18         Thread.Sleep(10000);
19         seats.Release();
20         print($"{person.gender} {person.id} finished using the toilet, Seats free:
21             {seats.CurrentCount}");
22
23         entryMutex[(int)person.gender].Wait();
24         if (--cnt[(int)person.gender] == 0)
25         {
26             print($"{person.gender}S have left he bathroom");
27             toilet.Release();
28         }
29         entryMutex[(int)person.gender].Release();
30     });
31     stopwatch.Stop();
32     print($"Execution time: {(float)stopwatch.ElapsedMilliseconds / 1000}");
33 }
```



Slika 2: Šematski prikaz algoritma na koji svaka nit nailazi sa naznačenim pozivima semafora

4 Izbegavanje izgladnjivanja

Opisano rešenje radi odlično u smislu da žene i muškarci ne mogu istovremeno koristiti toalet. Međutim problem je to što dok sve osobe jednog pola ne završe toalet, drugi pol neće doći na red. Mehanizam kojim se to rešava jeste uvođenjem još jednog semafora koji propušta do M niti i koji se zauzima čim nit počne da se izvršava, a oslobađa pre nego što osoba treba da koristi toalet. Jedan apstrakovan način da se o tome razmišlja jeste da ceo red osoba koje čekaju ispred toaleta podelimo na dva dela, gde se u jednom delu nalazi do M osoba, dok se ostatak nalazi u drugom. Tih M osoba su ljudi koji mogu ući u toalet, kako jedna od tih M osoba uđe u toalet, osoba iz drugog dela reda prelazi u prvi deo. Kada se prvi deo reda napuni osobama pola koje trenutno ne koriste toalet, više neće biti osoba koje mogu ući u toalet i kada ga poslednja osoba napusti, ljudi iz prvog dela reda počinju da ga koriste tj. menja se koji pol trenutno koristi toalet.

```
1 public static SemaphoreSlim genderSwitch = new SemaphoreSlim(Program.M);
```

Bitna stvar jeste da što je M manje, pol koji koristi toalet će se češće menjati. Takođe je važno reći da ako ima više ljudi jednog pola od drugog, velika vrednost M dovodi do toga da opet dođe do izgladnjivanja.

4.1 Implementacija sa izbegavanjem izgladnjivanja

```
1 public static void SimulateToilet(List<Person> people)
2 {
3     var stopwatch = new Stopwatch();
4     stopwatch.Start();
5     Parallel.ForEach(people, person =>
6     {
7         genderSwitch.Wait();
8         entryMutex[(int)person.gender].Wait();
9         if (++cnt[(int)person.gender] == 1)
10        {
11            toilet.Wait();
12            print($"{person.gender}S occupied the bathroom");
13        }
14        entryMutex[(int)person.gender].Release();
15
16        genderSwitch.Release();
17
18        seats.Wait();
19        print($"{person.gender} {person.id} started using the toilet, Seats free:
20            {seats.CurrentCount}");
21        Thread.Sleep(10000);
22        seats.Release();
23        print($"{person.gender} {person.id} finished using the toilet, Seats free:
24            {seats.CurrentCount}");
25
26        entryMutex[(int)person.gender].Wait();
27        if (--cnt[(int)person.gender] == 0)
28        {
29            print($"{person.gender}S have left he bathroom");
30            toilet.Release();
31        }
32        entryMutex[(int)person.gender].Release();
33    });
34    stopwatch.Stop();
35    print($"Execution time: {(float)stopwatch.ElapsedMilliseconds / 1000}");
36 }
```

5 Rezultat

Pre pokretanja programa je potrebno inicijalizovati konstante koje definišu kapacitet toaleta, konstantu M tj. podelu reda ljudi koji čekaju da uđu i na kraju same ljude. U ovom primeru je napravljeno 30 ljudi, i pola su muškarci, a druga polovina su žene. Ta lista ljudi je prosleđena metodi koja se bavi simulacijom toaleta. M je postavljeno kao jedna šestina ukupnog broja ljudi, jer to se pokazalo da daje dobru ravnotežu između upotrebe toaleta i učestanosti promene pola koji koristi toalet. Generalna preporuka je postaviti M da bude trećina broja ljudi od onog pola koji je u manjini.

```
1  internal class Program
2  {
3      public const int TOILET_CAPACITY = 4;
4      public const int PEOPLE_COUNT = 30;
5      public const int M = PEOPLE_COUNT / (2 * 3);
6
7      static void Main(string[] args)
8      {
9          List<Person> people = new List<Person>();
10         for (int i = 0; i < 30; i++)
11         {
12             if (i < 15) {
13                 people.Add(new Person(gender.MALE, i));
14             } else {
15                 people.Add(new Person(gender.FEMALE, i));
16             }
17         }
18         Toilet.SimulateToilet(people);
19         Console.ReadLine();
20     }
21 }
22
23 }
```

Na slikama 3 i 4 je prikazan rezultat izvršavanja gde je moguće videti da se u proseku na svakih 6-7 osoba menja koji pol koristi toalet. Takođe se može primetiti da je redosled ispisa poruka dobar ali povremeno same poruke prikazuju pogrešan broj slobodnih mesta. To se može desiti na više načina, ali glavni način na koji se to ovde dešava jeste da izađe jedna osoba, ispiše na ekran broj ostalih mesta tačno, zatim druga izađe, ali dok ona ispisuje poruku, druga osoba uđe. Jedan način da se to reši jeste da se pristup svim semaforima blokira dok se ispisuje poruka, ali to će drastično usporiti program. Još jedan način jeste pretvoriti **print** metodu u makro poziv ili **inline** funkciju, ali to su stvari koje C# ili ne podržava, ili nisu trivijalne za izvesti.

```
MALES occupied the bathroom
MALE 0 started using the toilet, Seats free: 3
MALE 1 started using the toilet, Seats free: 2
MALE 2 started using the toilet, Seats free: 1
MALE 3 started using the toilet, Seats free: 0
MALE 1 finished using the toilet, Seats free: 1
MALE 0 finished using the toilet, Seats free: 2
MALE 4 started using the toilet, Seats free: 1
MALE 2 finished using the toilet, Seats free: 2
MALE 3 finished using the toilet, Seats free: 3
MALE 4 finished using the toilet, Seats free: 4
MALES have left he bathroom
FEMALES occupied the bathroom
FEMALE 15 started using the toilet, Seats free: 3
FEMALE 16 started using the toilet, Seats free: 2
FEMALE 17 started using the toilet, Seats free: 1
FEMALE 20 started using the toilet, Seats free: 0
FEMALE 17 finished using the toilet, Seats free: 1
FEMALE 20 finished using the toilet, Seats free: 2
FEMALE 16 finished using the toilet, Seats free: 3
FEMALE 15 finished using the toilet, Seats free: 4
FEMALE 18 started using the toilet, Seats free: 3
FEMALE 19 started using the toilet, Seats free: 2
FEMALE 21 started using the toilet, Seats free: 1
FEMALE 19 finished using the toilet, Seats free: 2
FEMALE 21 finished using the toilet, Seats free: 3
FEMALE 18 finished using the toilet, Seats free: 4
FEMALES have left he bathroom
MALES occupied the bathroom
MALE 6 started using the toilet, Seats free: 3
MALE 8 started using the toilet, Seats free: 2
MALE 13 started using the toilet, Seats free: 1
MALE 10 started using the toilet, Seats free: 0
MALE 13 finished using the toilet, Seats free: 1
MALE 8 finished using the toilet, Seats free: 3
MALE 10 finished using the toilet, Seats free: 2
MALE 6 finished using the toilet, Seats free: 4
MALE 14 started using the toilet, Seats free: 3
MALE 11 started using the toilet, Seats free: 2
MALE 5 started using the toilet, Seats free: 1
MALE 14 finished using the toilet, Seats free: 2
MALE 5 finished using the toilet, Seats free: 3
MALE 11 finished using the toilet, Seats free: 4
MALES have left he bathroom
FEMALES occupied the bathroom
```

Slika 3: Prvi deo prikaza rezultata izvršavanja


```
MALE 11 started using the toilet, Seats free: 2
MALE 5 started using the toilet, Seats free: 1
MALE 14 finished using the toilet, Seats free: 2
MALE 5 finished using the toilet, Seats free: 3
MALE 11 finished using the toilet, Seats free: 4
MALES have left he bathroom
FEMALES occupied the bathroom
FEMALE 23 started using the toilet, Seats free: 3
FEMALE 27 started using the toilet, Seats free: 1
FEMALE 24 started using the toilet, Seats free: 2
FEMALE 25 started using the toilet, Seats free: 0
FEMALE 27 finished using the toilet, Seats free: 1
FEMALE 26 started using the toilet, Seats free: 1
FEMALE 22 started using the toilet, Seats free: 0
FEMALE 25 finished using the toilet, Seats free: 0
FEMALE 24 finished using the toilet, Seats free: 1
FEMALE 23 finished using the toilet, Seats free: 2
FEMALE 28 started using the toilet, Seats free: 1
FEMALE 22 finished using the toilet, Seats free: 2
FEMALE 26 finished using the toilet, Seats free: 3
FEMALE 28 finished using the toilet, Seats free: 4
FEMALES have left he bathroom
MALES occupied the bathroom
MALE 9 started using the toilet, Seats free: 3
MALE 7 started using the toilet, Seats free: 2
MALE 12 started using the toilet, Seats free: 1
MALE 9 finished using the toilet, Seats free: 2
MALE 7 finished using the toilet, Seats free: 3
MALE 12 finished using the toilet, Seats free: 4
MALES have left he bathroom
FEMALES occupied the bathroom
FEMALE 29 started using the toilet, Seats free: 3
FEMALE 29 finished using the toilet, Seats free: 4
FEMALES have left he bathroom
Execution time: 100.127
```

Slika 4: Drugi deo prikaza rezultata izvršavanja