

# Лабораторная работа 11

Модель системы массового обслуживания  $M|M|1$

---

Абу Сувейлим Мухаммед Мунирачи

25 мая 2024

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Абу Сувейлим Мухаммед Мунифович
- студент, НКНбд-01-21
- Российский университет дружбы народов
- 1032215135@pfur.ru

## Вводная часть

---

Цели:

Приобретение навыков моделирования в CPN tools.

1. Королькова, А. В. Моделирование информационных процессов : учебное пособие / А. В. Королькова, Д. С. Кулябов. - М. : РУДН, 2014. – 191 с. : ил.

## Постановка задачи

---

В систему поступает поток заявок двух типов, распределённый по пуассоновскому закону. Заявки поступают в очередь сервера на обработку. Дисциплина очереди - FIFO. Если сервер находится в режиме ожидания (нет заявок на сервере), то заявка поступает на обработку сервером.



## Выполнение работы

---

1. Будем использовать три отдельных листа: на первом листе опишем граф системы (рис. 1), на втором — генератор заявок (рис. 2), на третьем — сервер обработки заявок (рис. 3).

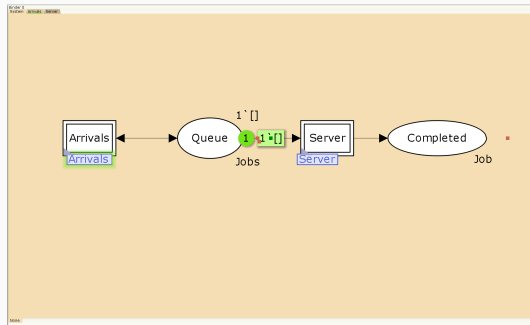


Figure 1: Рис. 1. Граф сети системы обработки заявок в очереди

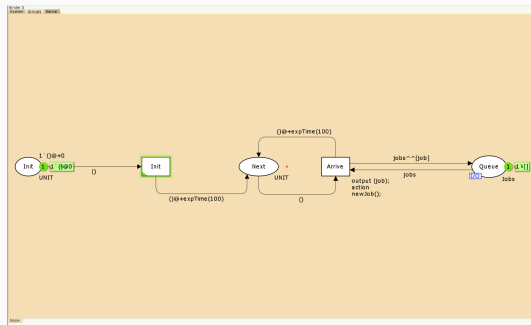


Figure 2: Рис. 2. Граф генератора заявок системы

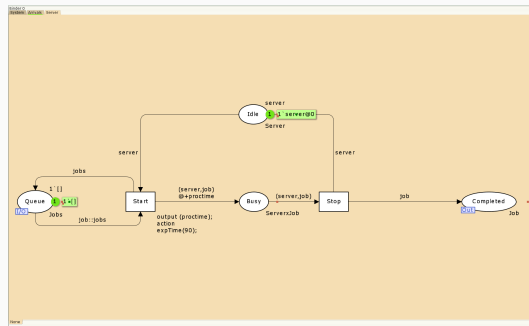


Figure 3: Рис. 3. Граф процесса обработки заявок на сервере системы

## 2. Зададим декларации системы:

```
▼ Declarations
  ▼ SYSTEM
    ▼ colset UNIT = unit timed;
    ▼ colset INT = int;
    ▼ colset Server = with server timed;
    ▼ colset JobType = with A | B;
    ▼ colset Job = record jobType : JobType * AT : INT;
    ▼ colset Jobs = list Job;
    ▼ colset ServerxJob = product Server * Job timed;
    ▼ var proctime : INT;
    ▼ var job : Job;
    ▼ var jobs: Jobs;
    ▼ fun expTime (mean : int) =
      let
        val realMean = Real.fromInt mean
        val rv = exponential((1.0/realMean))
      in
        floor (rv+0.5)
      end;
    ▼ fun intTime() = IntInf.toInt(time());
    ▼ fun newJob() = {jobType = JobType.ran(), AT = intTime()};
```

Figure 4: Рис. 4. Задание деклараций модели СМО  $M|M|1$

3. Зададим параметры модели на графах сети (см. рис 1-3).

4. Для мониторинга параметров очереди системы  $M|M|1$  потребуется палитра Monitoring. Выбираем Break Point (точка останова) и устанавливаем её на переход Start. После этого в разделе меню Monitor появится новый подраздел, который назовём Ostanovka. В этом подразделе необходимо внести изменения в функцию Predicate, которая будет выполняться при запуске монитора:

```
• global longdelay, time = 200;  
▼ Monitors  
  ▼ Ostanovka  
    Type: Break point  
    ▶ Nodes ordered by pages  
    ▼ Predicate  
      fun pred (bindelem) =  
        let  
          fun predBindElem (Server'Start (1,  
                                {job,jobs,proctime})) = Queue_Delay.count() = 200  
            | predBindElem _ = false  
        in  
          predBindElem bindelem  
        end
```

Figure 5: Рис. 5. Функция Predicate монитора Ostanovka



```
▼ Monitors
  ► Ostanovka
  ▼ Queue Delay
    ► Type: Data collection
    ► Nodes ordered by pages
    ► Predicate
    ▼ Observer
      fun obs (bindelem) =
      let
        fun obsBindElem (Server'Start (1, {job,jobs,proctime})) = (intTime() - (#AT job))
          | obsBindElem _ = ~1
        in
          obsBindElem bindelem
        end
      ► Init function
      ► Stop
```

Figure 6: Рис. 6. Функция Observer монитора Queue Delay

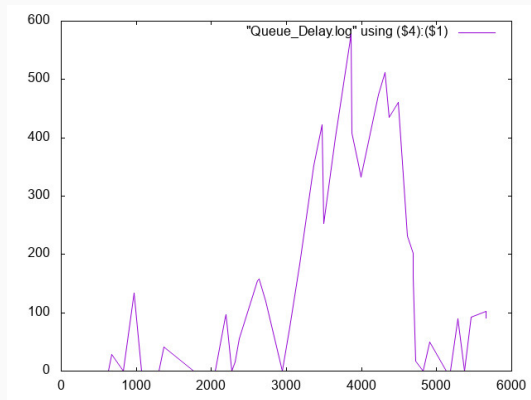


Figure 7: Рис. 7. График изменения задержки в очереди

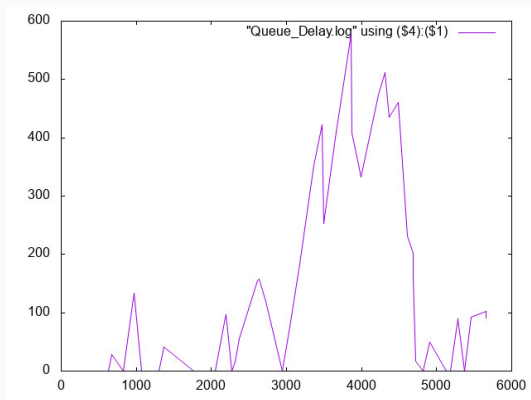


Figure 8: Рис. 8. Функция Observer монитора Queue Delay Real

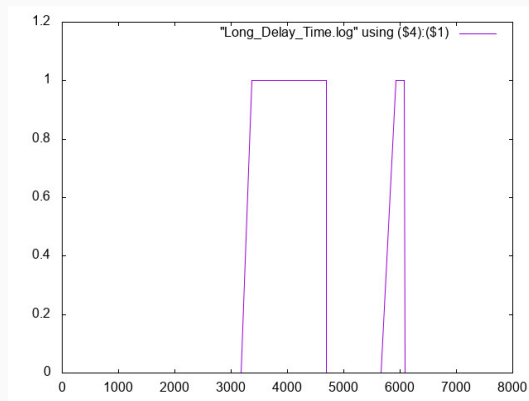
- ▼ Monitors
  - ▶ Ostanovka
  - ▶ Queue Delay
  - ▶ Queue Delay Real
  - ▼ Long Delay Time
    - ▶ Type: Data collection
    - ▶ Nodes ordered by pages
    - ▶ Predicate
    - ▼ Observer

```
fun obs (bindelem) =  
  if IntInf.toInt(Queue_Delay.last()) >= (!longdelaytime)  
  then 1  
  else 0
```
    - ▶ Init function
    - ▶ Stop

Figure 9: Рис. 9. Функция Observer монитора Long Delay Time

```
▼ Declarations  
  ► SYSTEM  
    ▼ globref longdelaytime = 200;
```

Figure 10: Рис. 10. Определение longdelaytime в декларациях



**Figure 11:** Рис. 11. Периоды времени, когда значения задержки в очереди превышали заданное значение

- Изучали как работать с CPN tools. [1]