

# Лабораторная работа №11

Дисциплина: Имитационное моделирование

---

Ганина Т. С.

18 апреля 2025

Группа НФИбд-01-22

Российский университет дружбы народов, Москва, Россия

## Информация

---

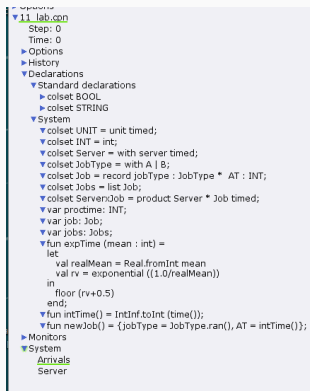
- Ганина Таисия Сергеевна
- Студентка 3го курса, группа НФИбд-01-22
- Фундаментальная информатика и информационные технологии
- Российский университет дружбы народов
- Ссылка на репозиторий гитхаба `tsganina`

## Вводная часть

---

Реализовать модель  $M|M|1$  в CPN tools.

- Реализовать в CPN Tools модель системы массового обслуживания  $M|M|1$ .
- Настроить мониторинг параметров моделируемой системы и нарисовать графики очереди.



```
11 lab.com
  Step: 0
  Time: 0
  ▶ Options
  ▶ History
  ▼ Declarations
    ▼ Standard declarations
      ▶ colset BOOL
      ▶ colset STRING
    ▼ 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 ServerJob = 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()};
  ▶ Monitors
  ▼ System
    Arrivals
    Server
```

Рис. 1: Задание деклараций системы

## Зададим параметры модели на графах сети

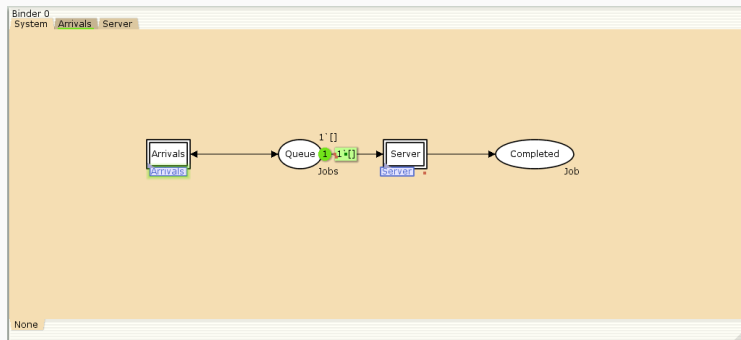


Рис. 2: Параметры элементов основного графа системы обработки заявок в очереди



# Параметры элементов генератора заявок системы

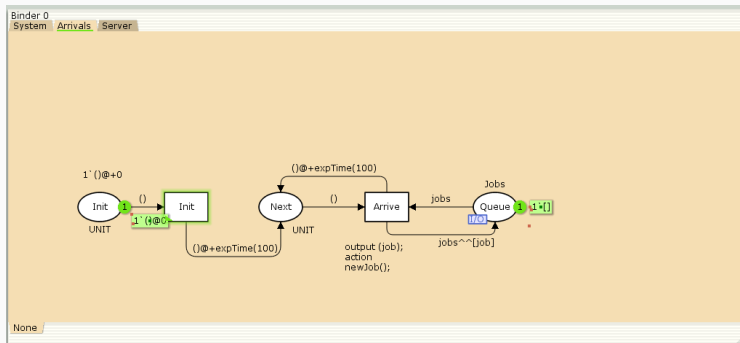


Рис. 3: Параметры элементов генератора заявок системы

# Параметры элементов обработчика заявок системы

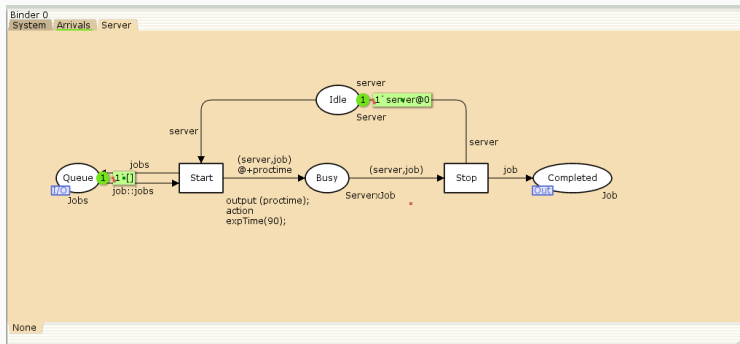


Рис. 4: Параметры элементов обработчика заявок системы

# Запуск системы обработки заявок в очереди

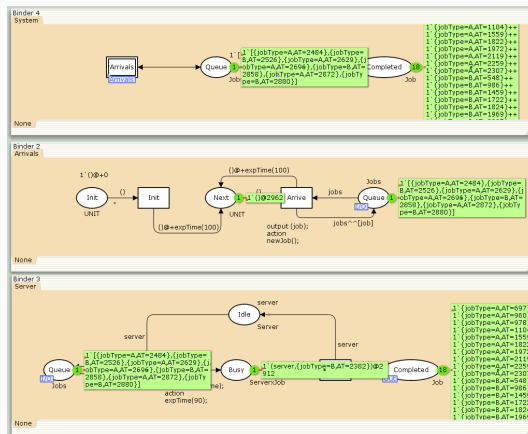


Рис. 5: Запуск системы обработки заявок в очереди

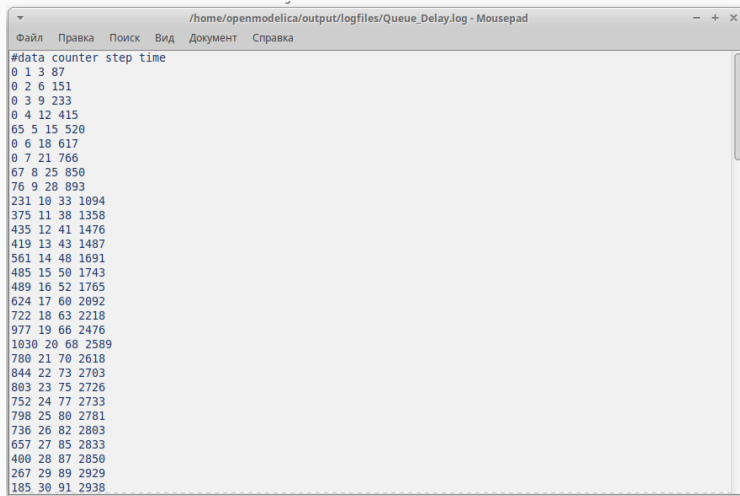
```
Binder 0
System fun obs<Queue Delay Real> fun obs<Queue Delay> fun pred<Ostanovka>
fun pred (bindelem) =
  let
    fun predBindElem (Server'Start (1,
                                   {job,jobs,proctime})) = Queue_Delay.count()=200
      | predBindElem _ = false
  in
    predBindElem bindelem
  end
```

Рис. 6: Функция Predicate монитора Ostanovka

```
Binder 0
System fun obs<Queue Delay Real> fun obs<Queue Delay>
fun obs (bindelem) =
let
  fun obsBindElem (Server'Start (1, {job,jobs,proctime})) = (intTime() - (#AT job))
  | obsBindElem _ = ~1
in
  obsBindElem bindelem
end
```

Рис. 7: Функция Observer монитора Queue Delay

## Файл Queue\_Delay.log



```
/home/openmodelica/output/logfiles/Queue_Delay.log - Mousepad
Файл  Правка  Поиск  Вид  Документ  Справка
#data counter step time
0 1 3 87
0 2 6 151
0 3 9 233
0 4 12 415
65 5 15 520
0 6 18 617
0 7 21 766
67 8 25 850
76 9 28 893
231 10 33 1094
375 11 38 1358
435 12 41 1476
419 13 43 1487
561 14 48 1691
485 15 50 1743
489 16 52 1765
624 17 60 2092
722 18 63 2218
977 19 66 2476
1030 20 68 2589
780 21 70 2618
844 22 73 2703
803 23 75 2726
752 24 77 2733
798 25 80 2781
736 26 82 2803
657 27 85 2833
400 28 87 2850
267 29 89 2929
185 30 91 2938
```

Рис. 8: Файл Queue\_Delay.log

## График значений задержки в очереди

```
#!/usr/bin/gnuplot -persist
# задаём текстовую кодировку,
# тип терминала, тип и размер шрифта

set encoding utf8
set term pngcairo font "Helvetica,9"

# задаём выходной файл графика
set out 'window_1.png'
plot "Queue_Delay.log" using ($4):($1) with lines
```

## График значений задержки в очереди

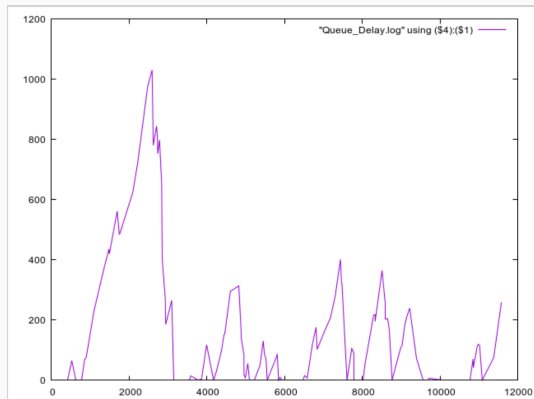


Рис. 9: График изменения задержки в очереди

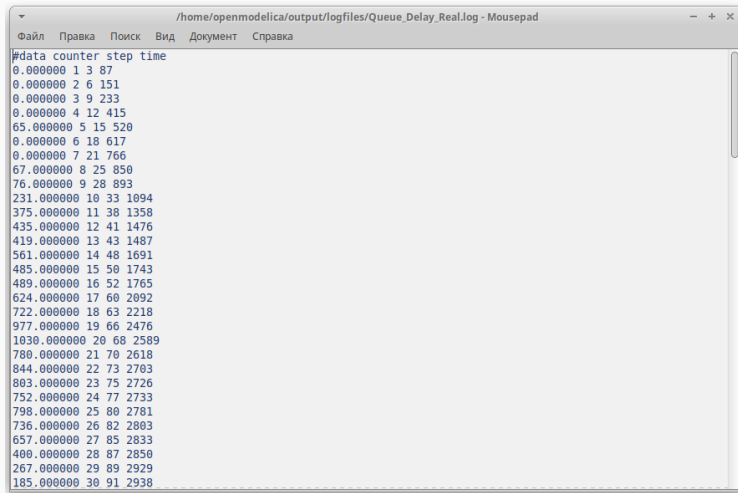


## Посчитаем задержку в действительных значениях

```
Binder 0
System fun obs <Queue Delay Real>
fun obs (bindelem) =
let
  fun obsBindElem (Server'Start (1, {job,jobs,proctime})) = Real.fromInt(intTime() - (#AT job))
  | obsBindElem _ = ~1.0
in
  obsBindElem bindelem
end
```

Рис. 10: Функция Observer монитора Queue Delay Real

## Содержимое Queue\_Delay\_Real.log

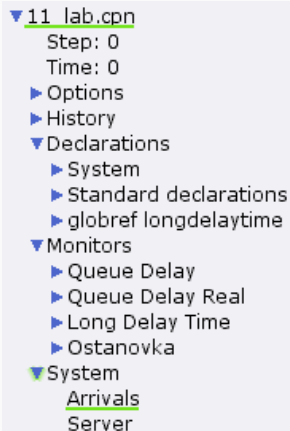


```
#data counter step time
0.000000 1 3 87
0.000000 2 6 151
0.000000 3 9 233
0.000000 4 12 415
65.000000 5 15 520
0.000000 6 18 617
0.000000 7 21 766
67.000000 8 25 850
76.000000 9 28 893
231.000000 10 33 1094
375.000000 11 38 1358
435.000000 12 41 1476
419.000000 13 43 1487
561.000000 14 48 1691
485.000000 15 50 1743
489.000000 16 52 1765
624.000000 17 60 2092
722.000000 18 63 2218
977.000000 19 66 2476
1030.000000 20 68 2589
780.000000 21 70 2618
844.000000 22 73 2703
803.000000 23 75 2726
752.000000 24 77 2733
798.000000 25 80 2781
736.000000 26 82 2803
657.000000 27 85 2833
400.000000 28 87 2850
267.000000 29 89 2929
185.000000 30 91 2938
```

Рис. 11: Содержимое Queue\_Delay\_Real.log

```
Binder 0  
System fun obs <Long Delay Time>  
fun obs (bindelem) =  
  if IntInf.toInt(Queue_Delay.last()) >= (!longdelaytime)  
  then 1  
  else 0
```

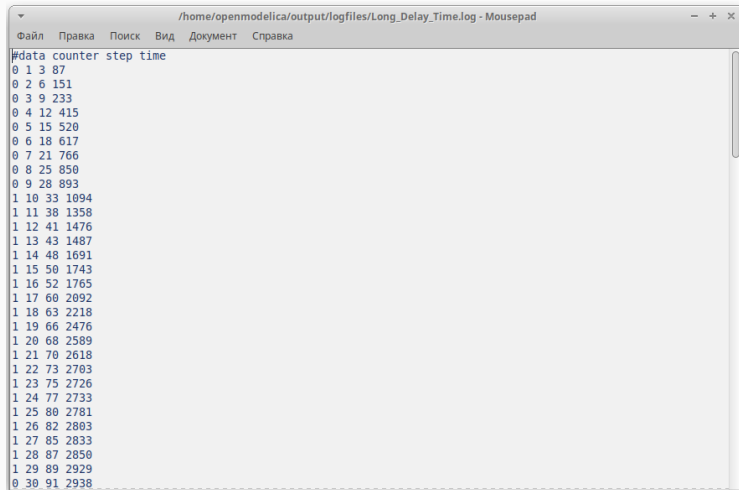
Рис. 12: Функция Observer монитора Long Delay Time



- ▼ 11 lab.cpn
  - Step: 0
  - Time: 0
  - ▶ Options
  - ▶ History
  - ▼ Declarations
    - ▶ System
    - ▶ Standard declarations
    - ▶ globref longdelaytime
  - ▼ Monitors
    - ▶ Queue Delay
    - ▶ Queue Delay Real
    - ▶ Long Delay Time
    - ▶ Ostanovka
  - ▼ System
    - Arrivals
    - Server

Рис. 13: Определение longdelaytime в декларациях

## Содержимое Long\_Delay\_Time.log



```
#data counter step time
0 1 3 87
0 2 6 151
0 3 9 233
0 4 12 415
0 5 15 520
0 6 18 617
0 7 21 766
0 8 25 850
0 9 28 893
1 10 33 1094
1 11 38 1358
1 12 41 1476
1 13 43 1487
1 14 48 1691
1 15 50 1743
1 16 52 1765
1 17 60 2092
1 18 63 2218
1 19 66 2476
1 20 68 2589
1 21 70 2618
1 22 73 2703
1 23 75 2726
1 24 77 2733
1 25 80 2781
1 26 82 2803
1 27 85 2833
1 28 87 2850
1 29 89 2929
0 30 91 2938
```

Рис. 14: Содержимое Long\_Delay\_Time.log

```
#!/usr/bin/gnuplot -persist
# задаём текстовую кодировку,
# тип терминала, тип и размер шрифта

set encoding utf8
set term pngcairo font "Helvetica,9"

# задаём выходной файл графика
set out 'window_2.png'
set style line 2
plot [0:] [0:1.2] "Long_Delay_Time.log" using ($4):($1) with lines
```

Периоды времени, когда значения задержки в очереди превышали заданное значение.  
График

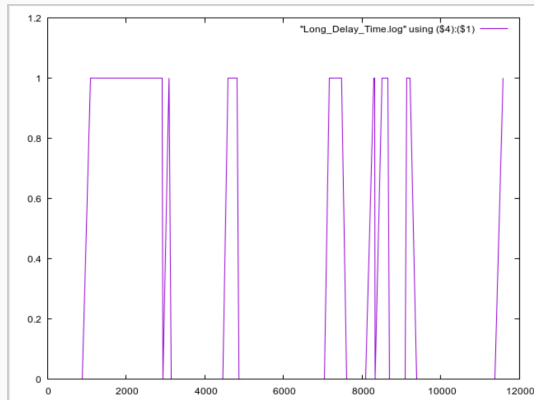


Рис. 15: Периоды времени, когда значения задержки в очереди превышали заданное значение

## Результаты

---



В процессе выполнения данной лабораторной работы я реализовала модель системы массового обслуживания  $M|M|1$  в CPN Tools.