Санкт-Петербургский политехнический университет Петра Великого Институт компьютерных наук и технологий

Высшая школа программной инженерии

КУРСОВАЯ РАБОТА

«Методы имитационного моделирования»

по дисциплине «Архитектура программных систем»

|  |  |
| --- | --- |
| Выполнил  студент гр. 3530904/00105 | Бутняков С.А. |
| Руководитель | Гончаров А.В. |

Санкт-Петербург 2023

ОГЛАВЛЕНИЕ

[Постановка задачи 3](#_Toc127895835)

[Принцип ∆t 3](#_Toc127895836)

[Принцип особых состояний 3](#_Toc127895837)

[Диаграммы 4](#_Toc127895838)

[Формализованная схема и описание СМО 5](#_Toc127895839)

[Вариант СМО, рассматриваемой в данной работе 5](#_Toc127895840)

[Вывод законов распределения 6](#_Toc127895841)

[Пуассоновский закон распределения 6](#_Toc127895842)

[Равномерный закон распределения 6](#_Toc127895843)

[Пример технической системы (ВС или части ВС), удовлетворяющей формализованному описанию 7](#_Toc127895844)

[Ограничения и требуемые характеристики 7](#_Toc127895845)

[Стоимость компонентов системы 7](#_Toc127895846)

[ПО для моделирования системы 8](#_Toc127895847)

[Блок-схема 8](#_Toc127895848)

[Структура 8](#_Toc127895849)

[Пользовательский интерфейс 9](#_Toc127895850)

[Результаты работы имитационной модели 12](#_Toc127895851)

[Анализ результатов, выводы и рекомендации по выбору конфигурации системы 12](#_Toc127895852)

[Выводы 16](#_Toc127895853)

# Постановка задачи

Целью курсовой работы является создание модели вычислительной системы (ВС) или ее части на некотором уровне детализации, описывающей и имитирующей ее структуру и функциональность. Каждый реальный объект (реальная ВС) обладает бесконечной сложностью, множеством характеристик, внутренних и внешних связей. Модель есть приближенное описание объекта с целью получения требуемых результатов с определенной точностью и достоверностью.

При необходимости исследования поведенческих характеристик ВС в процессе исследования выгодно использовать не сам объект, а его модель. Степень приближения модели к описываемому объекту может быть различной и зависит от требований задачи.

Существуют различные типы моделей:

1. Аналитические (математические) модели
2. Аналоговые модели
3. Физические модели
4. Имитационные модели

Последний тип моделей является предметом нашего изучения. Одним из подходов к построению имитационной модели является построение ее в виде системы массового обслуживания (СМО), с характерной для СМО терминологией: источник, буфер, прибор, диспетчер, заявка (требование). Существуют два подхода к построению моделирующего алгоритма:

## Принцип ∆t

Принцип ∆t – универсальный метод построения моделирующего алгоритма, когда состояние объекта проверяется через фиксированный интервал модельного времени. Суть его заключается в следующем: в каждый момент времени t получают приближенные значения характеристик исследуемого объекта. ∆t можно получить детерминированным способом. Основной критерий выбора ∆t – он должен быть настолько мал, чтобы не пропустить событие в моделируемой системе, которое должно быть учтено при выбранной детальности моделирования. Метод неэффективен, т. к. постоянно проверяет состояние объектов моделирования, не изменяющихся при этом, особенно при малых ∆t.

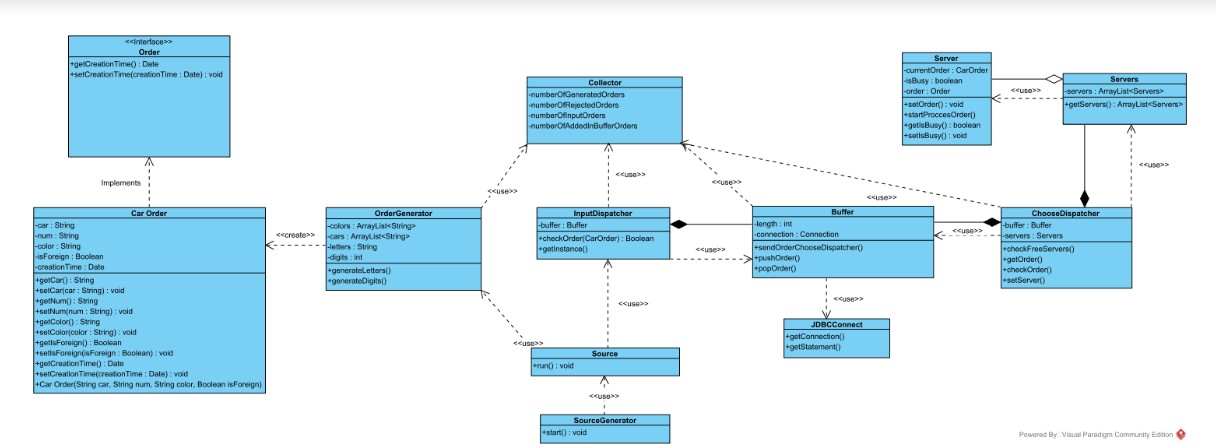
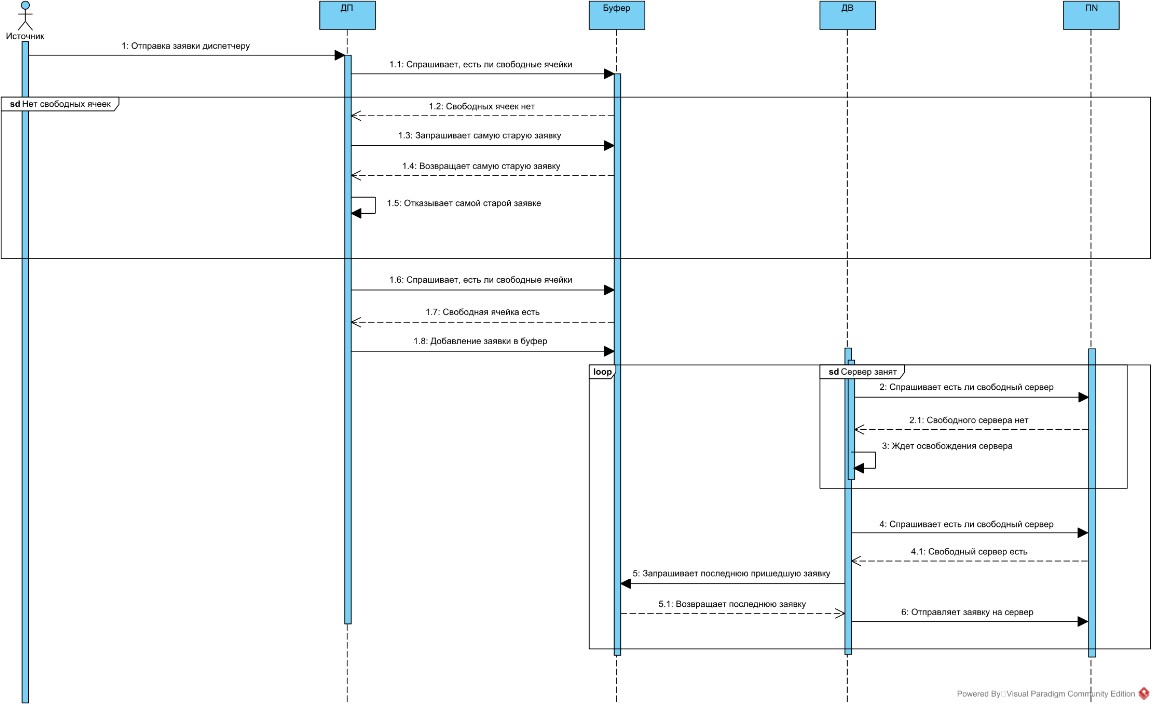
## Принцип особых состояний

При исследовании реальной системы интервалы, в которых состояние ее не меняется, не представляют интереса. Имеют значение только переходы системы из одного состояния в другое в некоторые моменты времени. Эти переходы определяются особыми состояниями или событиями. Рассмотрим некоторые типы особых событий, которые изменяют состояние системы:

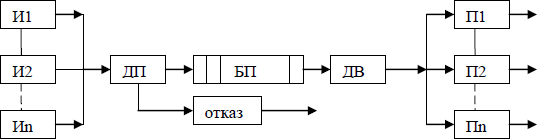
1. Поступление заявки в СМО (момент генерации заявки источником);
2. Освобождение прибора (готовность прибора взять заявку на обслуживание);
3. Окончание процесса моделирования.

Использование принципа особых событий для построения имитационной модели наиболее эффективно. В настоящей курсовой работе предлагается использовать именно этом принцип.

# Диаграммы



# Формализованная схема и описание СМО



*Рис 1. Формализованная схема СМО*

Здесь Иi (i = 1..n) – источник заявок, который генерирует заявки, а все вместе n источников создают входной поток заявок в систему. Каждая заявка приходит в СМО со своими характеристиками. Это входное T – время генерации заявки (время поступления ее в СМО) и номер заявки, составленный из номера источника, сгенерировавшего заявку, и порядкового номера заявки от этого источника. Например, (2.3) – третья заявка от второго источника.

П – приборы, которые обслуживают заявки и создают выходной поток заявок после обслуживания.

БП – буферная память (место для хранения очереди заявок). В общей памяти хранятся заявки от различных источников. Порядок их записи в БП определяется только дисциплиной буферизации.

ДП – диспетчер постановки заявок. ДВ – диспетчер выбора заявок.

## Вариант (25) СМО, рассматриваемой в данной работе

* ИБ – бесконечный источник.
* И31 – пуассоновский закон распределения.
* ПЗ2 – равномерный закон распределения времени обслуживания.
* Д1ОЗ3 – дисциплина буферизации; запись в буфер, если есть место – на свободное место.

Заявка встанет в очередь на первое от начала свободное место, если такое найдется. Сдвига очереди в этом случае не происходит.

* Д1ОО3 – дисциплина отказа; самая старая в буфере.

Эта дисциплина рассматривает только время прихода заявок в систему (момент генерации заявок источником). Заявка, раньше других вставшая в буфер, получает отказ, уходит из системы и на ее место встает пришедшая заявка.

* Д2П1 – дисциплина постановки на обслуживание; выбор прибора; приоритет по номеру прибора. Приоритеты приборов, так же, как и приоритеты источников, определяются номерами приборов. Поэтому поиск свободного прибора ведется последовательным перебором, каждый раз начиная с самого приоритетного.
* Д2Б2 – дисциплина постановки на обслуживание; выбор заявки из буфера; LIFO.

В этом случае раньше других будет выбрана из буфера на обслуживание та заявка, которая пришла последней.

* ОР1 – отображение результатов в автоматическом режиме; сводная таблица результатов. Шаг в этом случае — интервал модельного времени от одного особого события до другого ближайшего по времени особого события.
* ОД1 – отображение результатов в пошаговом режиме; календарь событий, текущее состояние.

# Вывод законов распределения

## Пуассоновский закон распределения



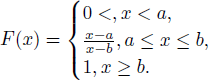
где γ – заданное значение.



В программе данное выражение записано следующим образом:

(-1 / lambda) \* Math.log(Math.random());

## Равномерный закон распределения



где a и b – заданные значения.



В программе выражение записано следующим образом:

Math.random() \* (b - a) + a;

# Пример технической системы (ВС или части ВС), удовлетворяющей формализованному описанию

|  |  |
| --- | --- |
| Система | Автомастерская |
| Источники | Источником является сайт с авиабилетами, который отсылает данные на обработку в виде пакета размером 16 Кбайт. |
| Приборы | Приборами являются сервера, которые обрабатывают полученную заявку и отправляют результат авиакомпаниям. |
| Буфер | Буфером является буфер коммутатора, который имеет объем 16 Кбайт (т. е. может хранить 1 заявку)  и может быть наращен до 128 кб для хранения 8 заявок. |
| Дисциплина постановки в буфер | Заявка встает на ближайшее от начала свободное  место в буфере. Сдвига в таком случае не происходит. |
| Дисциплина выборки из буфера | Выбирается последняя добавленная в буфер заявка. |
| Дисциплина отказа | Самая старая заявка в буфере. |
| Дисциплина постановки на обслуживание | Приоритет по номеру прибора. |

# Ограничения и требуемые характеристики

Вероятность отказа должна составлять не более 10%. Загрузка приборов более 90%.

Время пребывания заявки в системе не более 5 мс. Рассматриваемый диапазон характеристик системы и доступные типы процессоров и характеристики программно-аппаратного комплекса, построенного на данном типе процессора приведены ниже в таблицах компонентов системы.

|  |  |
| --- | --- |
| Количество сайтов-источников | 4 штуки |
| Вес заявки | 16 Кб |
| Объем буфера | От 16Кб до 128 Кб |
| Количество приборов | От 1 до 7 |

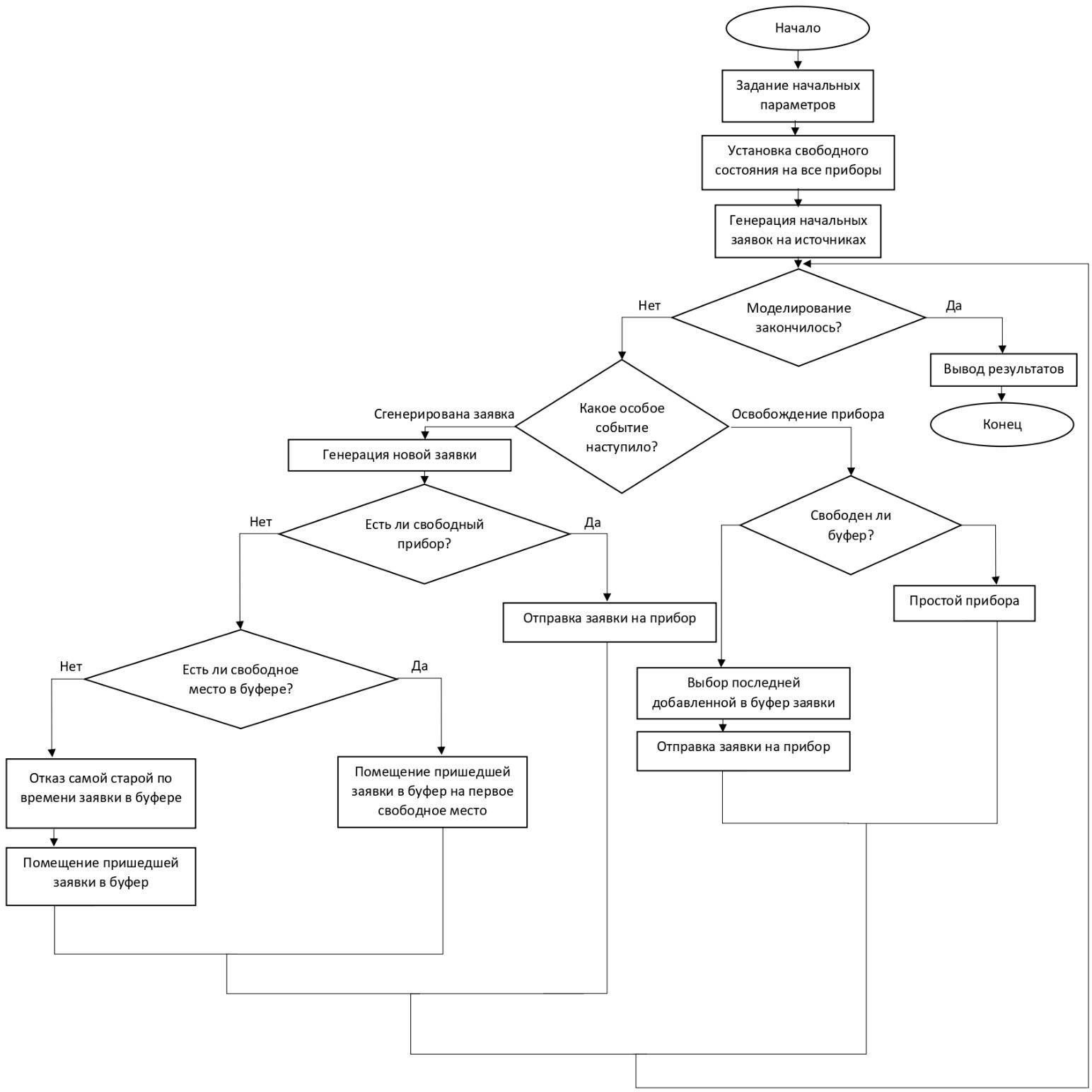
## Стоимость компонентов системы

Условимся, что стоимость одного прибора для обработки составляет – 10000 рублей.

Стоимость расширения буфера на 1 слот – 500 рублей.

# ПО для моделирования системы

## Блок-схема



*Рис 3. Блок-схема ПО для моделирования системы*

## Структура

Для моделирования системы была написана программа на языке Java с использованием платформы для создания приложений с графическим пользовательским интерфейсом Java FX. Приложение использует принцип ООП и содержит следующие классы:

• Ticket — содержит описание заявки, реализует методы для получения этих описаний;

• Controller — реализует функционирование системы в автоматическом и пошаговом режимах;

• Source — содержит описание источника: его номер, параметры закона распределения, текущую заявку;

• SourceMetrics – реализует сбор статистики по каждому источнику: количество сгенерированных, обработанных и выбитых заявок одного источника, а также время нахождения в системе, время ожидания обслуживания и время обработки заявок данного источника;

• SourceController – осуществляет управление источниками системы: хранит все источники, регулирует отправку заявок в буфер по кольцу;

• Device — содержит описание прибора: его номер, коэффициент закона распределения, время окончания обработки предыдущей заявки, текущую заявку;

• DeviceMetrics – реализует сбор статистики по каждому прибору: время работы прибора и количество обработанных заявок;

• DeviceController – осуществляет управление приборами системы: хранит все приборы, регулирует получение заявок из буфера и их обработку;

• Buffer - содержит описание буфера: его номер и текущую заявку;

• BufferController — реализует методы управления буфером: выбирает какой из буферов освободить и какой занять;

• MetricsController – формирует общую статистику по всей системе, объединяя информацию, полученную от SourceMetrics и DeviceMetrics;

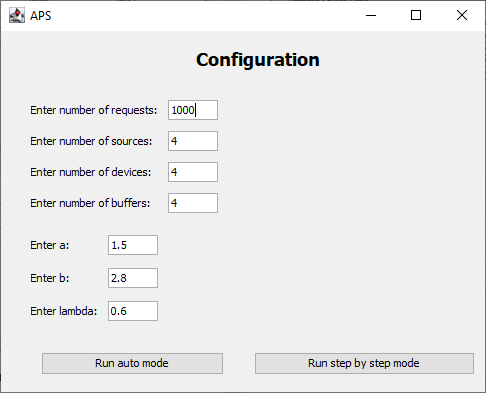
• AutoMode — класс графического интерфейса, отображающий автоматический режим модели;

• StepByStepMode — класс графического интерфейса, отображающий пошаговый режим модели;

• MainMenu - класс графического интерфейса, отображающий общее меню программы для настройки параметров

## Пользовательский интерфейс

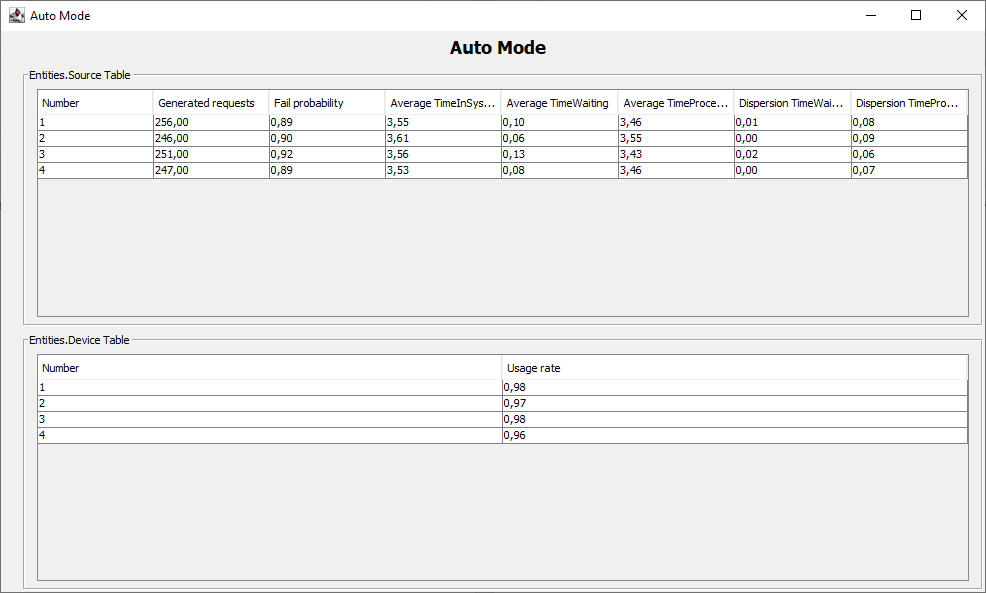
При запуске приложения первоначально открывается окно настроек:



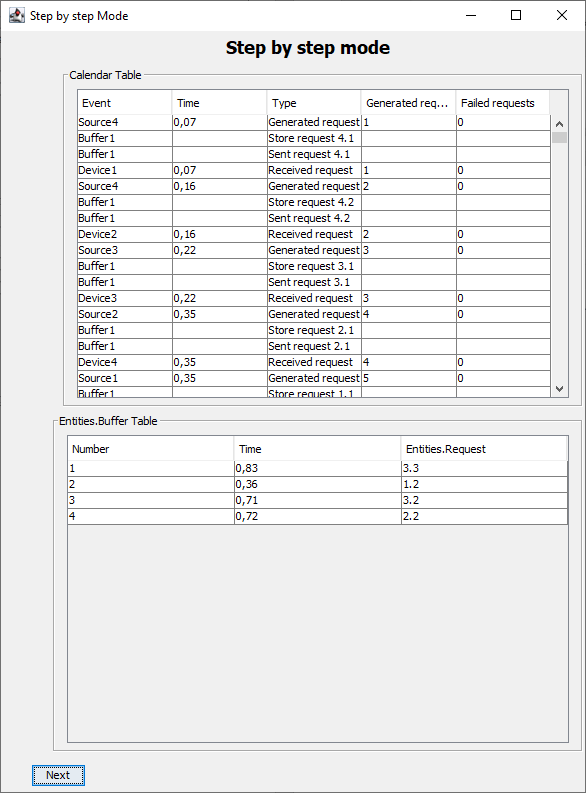
*Рис 4. Окно настроек приложения*

Приложение ожидает от пользователя ввода количества источников, количества устройств буфера, количества приборов, интенсивности Пуссановского потока λ, границы равномерного потока, а также количества заявок, которые будут сгенерированы в автоматическом режиме.

Здесь пользователь может выбрать использование автоматического или пошагового режима.



*Рис 5. Автоматический режим приложения*



*Рис 6. Автоматический режим приложения*

# Результаты работы имитационной модели

Количество реализаций, необходимое для получения нужной точности при заданной доверительной вероятности, можно оценивать по формуле:

*𝑡*2(1 *− 𝑝*)

*α*

*𝑁* =

*𝑝𝛿*2

где p – вероятность отказа заявкам в обслуживании, tα = 1.643, для α = 0.9,

σ = 0.1 - относительная точность.

По результатам работы программы получено, что в большинстве случаев для достижения заданной точности необходимо около 1000 заявок

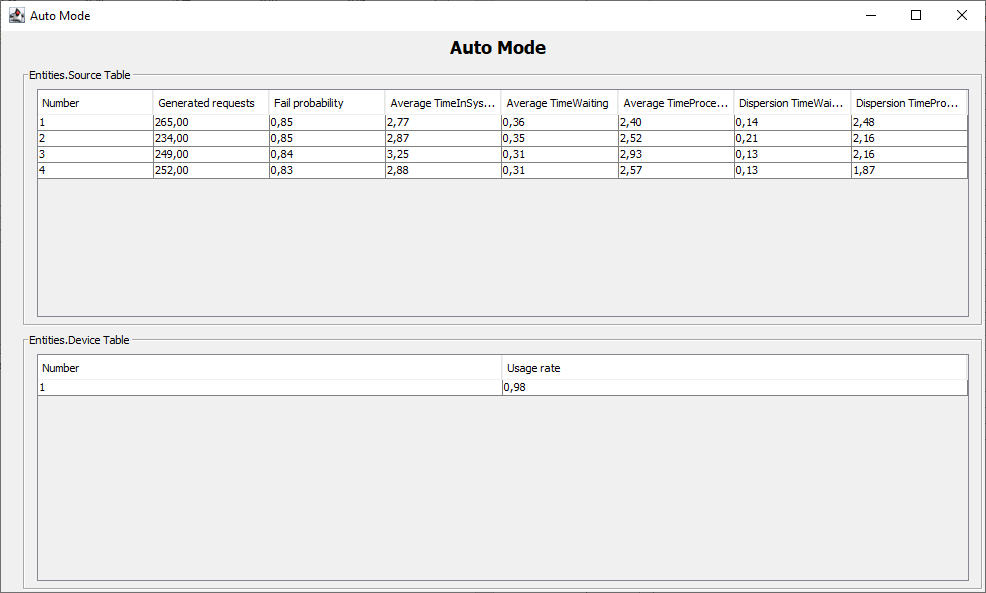
## Анализ результатов, выводы и рекомендации по выбору конфигурации системы

Рассмотрим вариант конфигурации, когда имеется всего 4 источника. Т. к. целью

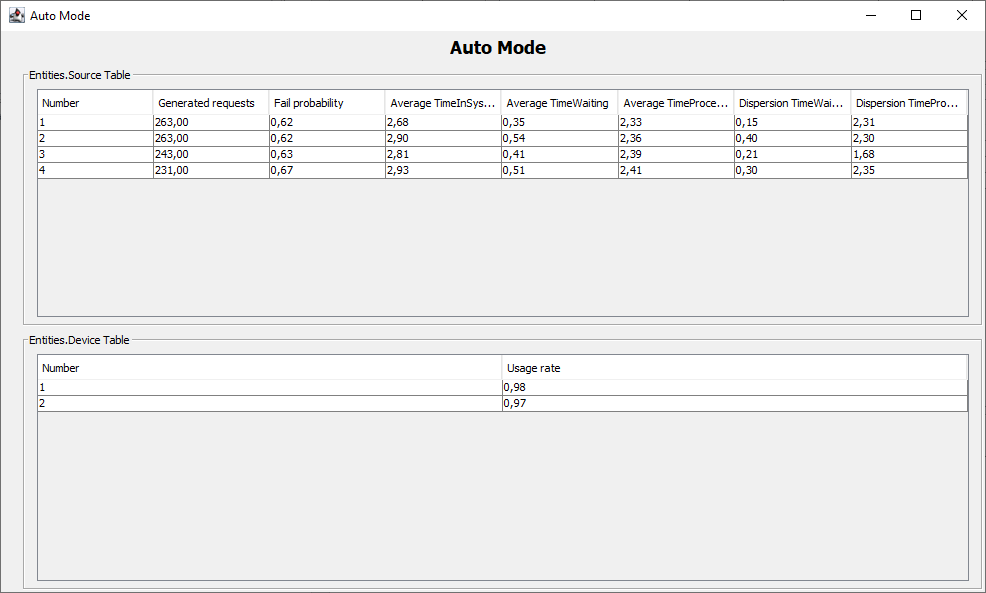
моделирования является выбор конфигурации системы, требующей наименьшее количество ресурсов и обрабатывающей максимальный поток информации, то начнем с проверки конфигурации с минимальным числом приборов и минимальным размером буфера.

Также ограничим время пребывания заявки в буфере – 5 мс.

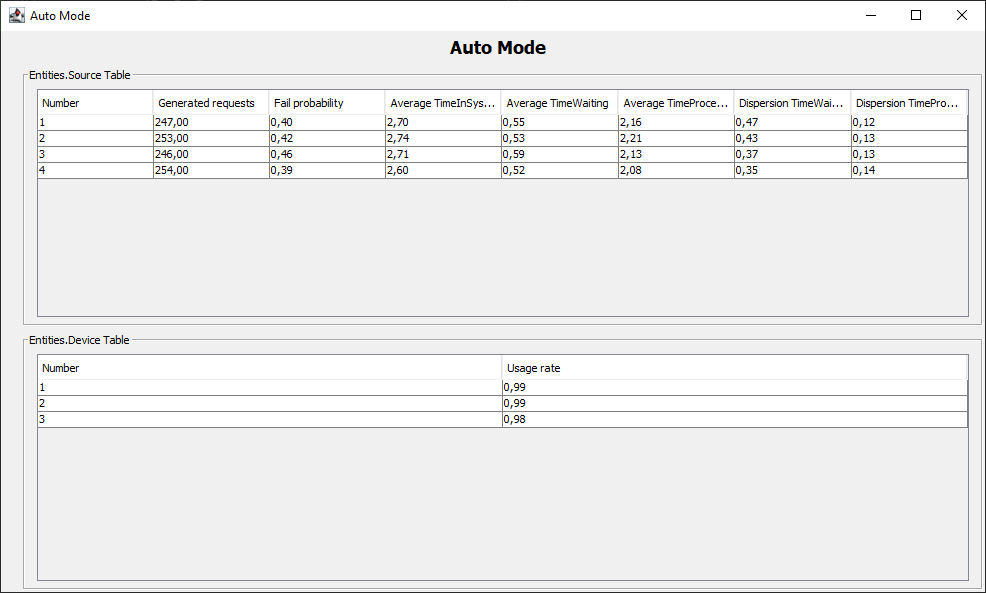
Количество источников во всех опытах возьмем равным 4.



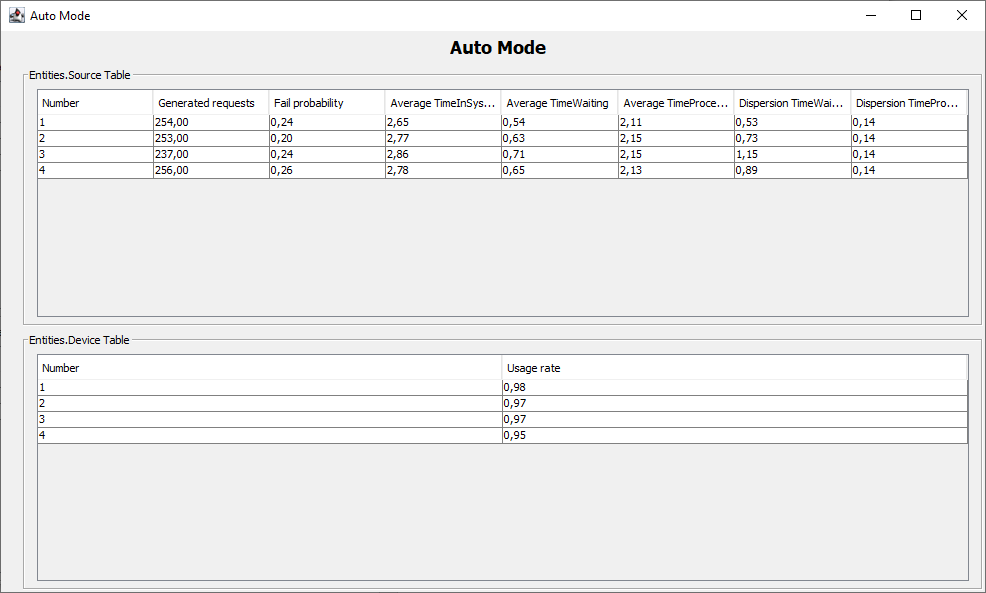
В первом опыте прибор эффективно используется, время нахождения в системе допустимо, но 84% заявок уходят в отказ, что является недопустимым, поэтому добавляем ещё один прибор и расширяем буфер на 1.



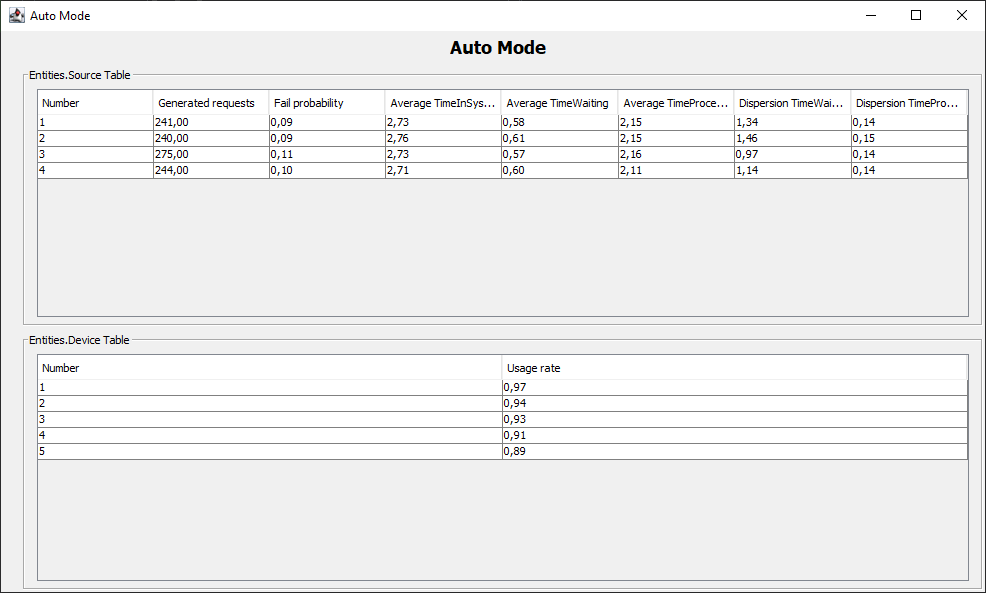
Вероятность отказа снизилась, но до сих пор не является удовлетворительной. Время нахождения в системе допустимо. Возьмем для обработки еще один прибор и еще одно место в буфере.



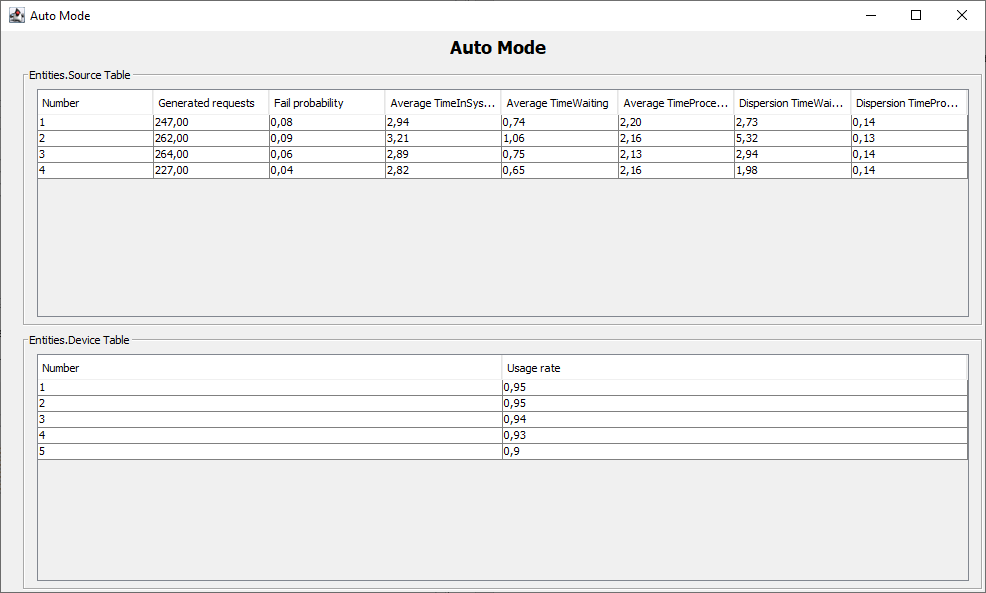
Вероятность отказа снизилась, но до сих пор не является удовлетворительной. Время нахождения в системе допустимо. Возьмем для обработки еще один прибор и еще одно место в буфере.



Вероятность отказа упала, но все еще выше нормы, возьмем для обработки еще один прибор и один слот для буфера.



Вероятность отказа существенно упала, но при этом существуют случаи превышения нормы отказа, а также один прибор загружен менее, чем на 90 процентов. Увеличим количество слотов в буфере до 7.



Время нахождения в системе допустимо. Вероятность отказа снизилась до 4–9%, что нас полностью

устраивает. При уменьшении количества приборов (даже при улучшении их качества) система не будет удовлетворять нашим требованиям.

Стоимость системы: 10000 \* 5 + 500 \* 7 = 53500 рублей.

# Выводы

В ходе выполнения курсовой работы мной была написана система массового обслуживания на язык Java с использованием платформы Java FX. С помощью получившейся программы была проанализирована реальная система и подобрана максимально выгодная комплектация данной системы.