

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Информационные системы в строительстве»

МОДЕЛИ И МЕТОДЫ ИССЛЕДОВАНИЯ ИНФОРМАЦИОННЫХ  
ПРОЦЕССОВ и СИСТЕМ

Методические указания по выполнению контрольной работы  
для магистрантов направления подготовки  
«Информационные системы и технологии» (заочная форма)

Ростов-на-Дону  
ДГТУ  
2020

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	3
Общие сведения о среде AnyLogic и установка приложения.....	5
Постановка задачи «Магистраль передачи данных».....	6
Начало работы в AnyLogic .....	7
Диаграмма процессов модели .....	8
Описание моделирования профилактики или сбоя в канале связи.....	10
Описание моделирования передачи сообщений .....	16
Анимация в модели .....	22
Запуск модели на выполнение .....	23
Задания контрольной работы (базовый уровень) .....	26
Задания контрольной работы (повышенный уровень).....	28
ЛИТЕРАТУРА .....	38

## ВВЕДЕНИЕ

Особым видом математических моделей являются имитационные модели. Имитационная модель – это компьютерная программа, которая описывает структуру и воспроизводит поведение реальной системы во времени. Можно сказать, что имитационная модель – это набор правил, согласно которым система переходит из одного состояния в другое. Правила могут задаваться самыми различными способами, например, дифференциальными уравнениями, диаграммами состояний, диаграммами процессов, расписаниями. Выходные данные модели всегда можно проанализировать прямо по ходу моделирования.

Имитационные модели разрабатываются с помощью специализированного программного обеспечения, в котором используются различные языки моделирования.

Имитационное моделирование – разработка компьютерных моделей и постановка экспериментов на них.

Можно выделить основные преимущества имитационного моделирования по сравнению с аналитическим.

1. Имитационные модели позволяют анализировать системы и находить решения в тех случаях, когда такие методы, как аналитические вычисления не справляются с задачей.

2. Разрабатывать имитационную модель гораздо проще, чем аналитическую, поскольку процесс создания модели будет инкрементальным (пошаговый, усложняющийся постепенно) и модульным.

3. Структура имитационной модели естественным образом отображает структуру моделируемой системы.

4. Имитационная модель позволяет отслеживать все объекты системы, учтенные в выбранном уровне абстракции, добавлять метрики и проводить статистический анализ.

5. Имитационная модель дает возможность проигрывать модель во времени и анимировать ее поведение. Анимация будет неоспоримым преимуществом при демонстрации модели и может оказаться полезной для верификации (тестирование на предмет соответствия проекта техническому заданию – замыслу проектировщика).

В имитационном моделировании под методом понимается некая основа, которую мы используем, чтобы «перевести» систему из реального мира в мир моделей. Метод предполагает определенный язык, положения и условия для

разработки модели. На данный момент, существует три метода (концепции) моделирования:

- системная динамика и динамические системы;
- дискретно-событийное моделирование;
- агентное моделирование.

Каждый метод применяется в некотором диапазоне уровней абстракции. Системная динамика предполагает очень высокий уровень абстракции и, как правило, используется для стратегического моделирования. Дискретно-событийное моделирование поддерживает средний и низкий уровни абстракции. Между ними находятся агентные модели, которые могут быть как очень детализированными, когда агенты представляют физические объекты, так и предельно абстрактными, когда с помощью агентов моделируются конкурирующие компании или правительства государств.

В пособии рассматривается создание модели системы массового обслуживания (дискретно-событийное моделирование). Динамика таких систем представляется как последовательность операций (прибытие, задержка, захват ресурса, разделение, ...) над агентами (транзактами), представляющими клиентов, документы, звонки, пакеты данных, транспортные средства и т.п. Эти агенты сами не контролируют свою динамику, но могут обладать определёнными атрибутами и переменными, влияющими на процесс их обработки (например, тип звонка, сложность работы) или накапливающими статистику (общее время ожидания, стоимость).

Данное пособие поможет овладеть на начальном уровне навыками создания имитационных моделей в системе моделирования Anylogic. Для самостоятельного углубленного изучения методом моделирования в этой системе рекомендуется использовать учебную литературу [1,2]. Знание принципов и возможностей имитационного моделирования, умение создавать и применять модели являются в настоящее время необходимыми требованиями к инженеру, менеджеру, бизнес-аналитику.

## Общие сведения о среде AnyLogic и установка приложения

Среда имитационного моделирования AnyLogic основана на Java и базируется на платформе Eclipse – современном стандарте для бизнес-приложений. Благодаря Eclipse AnyLogic работает на всех распространенных операционных системах (Windows, Mac, Linux и т.д.).

Для выполнения практических и лабораторных заданий по моделированию в системе AnyLogic нужно скачать и установить на компьютере свободно распространяемую версию Personal Learning Edition (время ее использования не ограничено) с официального сайта разработчика

<https://www.anylogic.ru/downloads/> . Объем < 1Гб.

При первоначальном запуске программы Anylogic отображается Начальная страница, на которой пользователю предлагается ознакомиться с многочисленными примерами моделей, разработанных в этой системе. Попасть на эту страницу в ходе дальнейшей работы можно из меню Справка. Нужно сказать, что даже в бесплатной версии AnyLogic разработчики предоставляют огромное количество материалов для обучения.

На рисунке (Рисунок 1) показан интерфейс среды имитационного моделирования AnyLogic. Сверху располагаются Меню, панели инструментов, по центру – рабочая область (графический редактор или область структурной диаграммы), справа – окно свойств объектов, слева – Окно панелей Проекты и Палитра, эти панели переключаются щелчком по заголовку. В нижней части размещаются окно ошибок и др. Отображение тех или иных окон можно настроить с помощью кнопки Вид в главном меню AnyLogic.

Окно панели Проекты (слева), навигатор системы, отображает имена проектов, открываемых в последнее время. При ненужности их можно закрыть и окно станет пустым. Каждый проект организован иерархически в виде дерева: сам проект образует верхний уровень, эксперименты, классы активных объектов и Java классы образуют следующий уровень. Элементы, входящие в состав активных объектов, вложены в соответствующую подветвь дерева класса активного объекта и т. д.

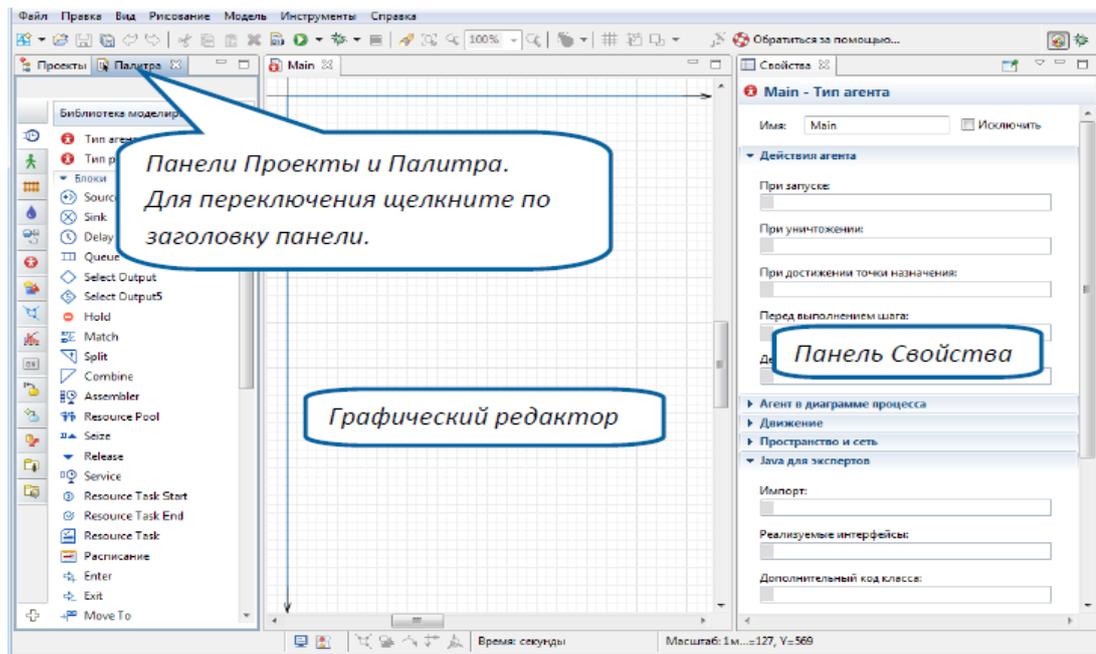


Рисунок 1. Интерфейс среды имитационного моделирования AnyLogic

Окно панели Палитра (слева) включает элементы (графические объекты), которые могут быть добавлены на структурную диаграмму активного объекта. Элементы разбиты по группам, отображаемым на разных вкладках Палитры. Количество отображаемых вкладок можно увеличить (уменьшить), используя «+» в нижнем левом углу окна Палитра. Вкладки переключаются по щелчку. Объекты палитры добавляются на диаграмму модели путем перетаскивания их в окно графического редактора.

### Постановка задачи «Магистраль передачи данных»

Магистраль передачи данных состоит из двух каналов (основного и резервного) и общего накопителя. При нормальной работе сообщения передаются по основному каналу за  $7 \pm 3$  мкс. В основном канале происходит профилактика (или сбой) через интервалы времени  $200 \pm 35$  мкс. Если профилактика происходит во время передачи, то за 2 мкс запускается запасной канал, который передает прерванное сообщение с самого начала в течение  $8 \pm 3$  мкс. Восстановление основного канала занимает  $23 \pm 7$  мкс. После восстановления резервный канал выключается и основной канал продолжает работу с очередного сообщения. Сообщения поступают через  $9 \pm 4$  мкс. и остаются в накопителе до окончания передачи. Если в накопителе в очереди оказывается более 10 сообщений, то очередное сообщение оказывается потерянным. В случае профилактики передаваемое сообщение передается повторно по запасному каналу.

Смоделировать работу магистрали передачи данных в течение 1 мкч. Определить загрузку основного и запасного каналов, частоту профилактики канала и число потерянных сообщений.

### Начало работы в AnyLogic

рассмотрим решение задачи «Магистраль передачи данных» средствами AnyLogic.

Для создания модели выберите в главном меню AnyLogic: Файл > Создать > Модель. Откроется диалоговое окно Новая модель (Рисунок 2), в котором нужно задать имя модели, в нашем случае это Магистраль, указать путь сохранения модели или принять тот путь, который создается по умолчанию. В нашем случае путь изменен. Имя Java пакета можно поменять, но оно обязательно начинается со строчной буквы. Созданный проект не забывайте периодически сохранять по ходу наполнения данными.

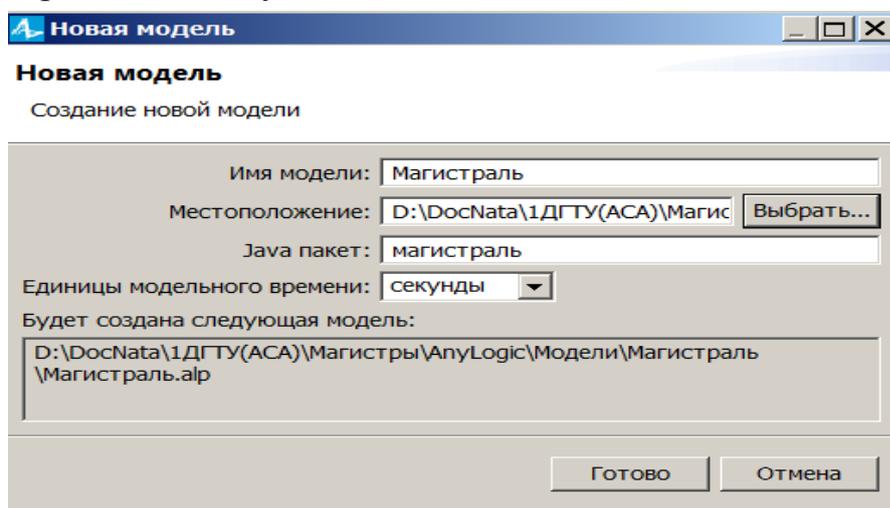


Рисунок 2

В результате создания модели по сути первоначально создается структура проекта в виде иерархического дерева, которую можно увидеть, открыв окно панели Проекты.

В окне Проекты может быть несколько открытых моделей, на рисунке (Рисунок 3) их две: Магистраль3 и Магистраль. Модель Магистраль только что создана, пока не заполнялась данными, на ней можно проиллюстрировать структуру вновь создаваемого проекта. По умолчанию в каждой модели создается один тип агента – **Main** и один простой эксперимент **Simulation: Main**, содержащий настройки запуска этой модели, встроенную **Базу данных** и **Презентацию**. Для создания в модели новых типа агента, эксперимента, базы дан-

ных, Java-класса и др. нужно в окне Проекты щелкнуть по имени модели правой кнопкой, далее в меню Создать выбрать из раскрывающегося списка создаваемый класс.

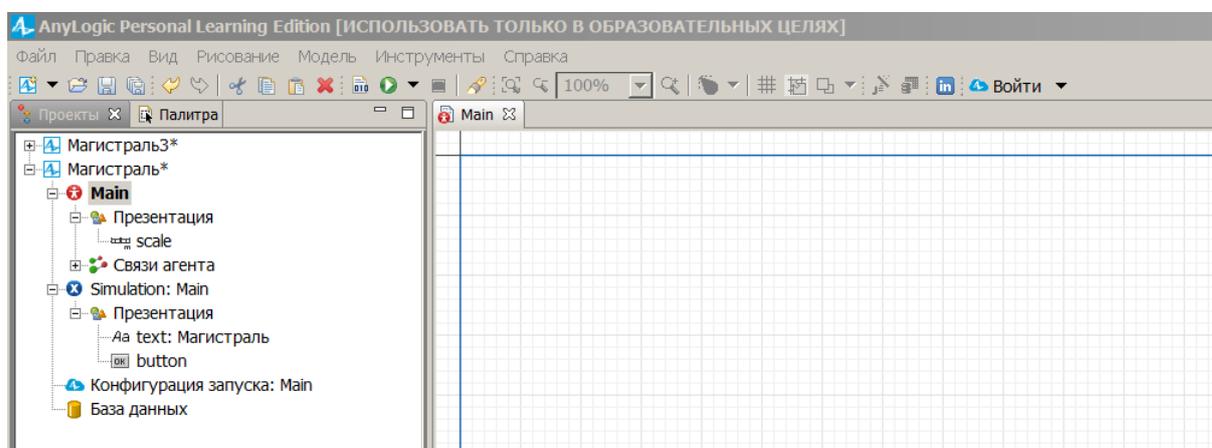


Рисунок 3

Если дважды щелкнуть по иконке  **Main** в окне Проекты, то справа или по центру, если открыто окно свойств, открывается графический редактор диаграммы агента верхнего уровня **Main** (Рисунок 3). Это главное графическое окно проекта, в нем будет создаваться структурная схема, размещаться анимационная картинка, которая отображается при проигрывании модели. В принципе, имя Main можно заменить на другое, но тогда так же нужно изменить имя простого эксперимента  **Simulation: Main**. Пока этого делать не будем.

Следует заметить, что термин «Агент» в AnyLogic используется в широком смысле и применяется к различным по смыслу и действиям объектам. Здесь в качестве агентов могут быть транзакты и ресурсы, но, кроме этого, узлы иерархического дерева, проекты, идеи и т.д.

### Диаграмма процессов модели

В диаграмме процессов будем использовать те блоки библиотеки моделирования процессов, которые позволяют описать логику решения поставленной задачи. Заметим, что предложенная ниже реализация не является единственной для данной модели, возможны и другие варианты, которые при желании, можно осуществить.

При создании более сложных моделей, чем рассматриваемая, могут возникнуть сложности с описанием логики задачи, поскольку нужно использовать конструкции на языке Java. Тем, кто владеет этим языком, безусловно, будет проще. Однако, предоставляемые справочные материалы позволяют

преодолеть эти трудности и тем, кто имеет о Java весьма поверхностное представление.

Пусть диаграмма процессов, имеет две ветви, которые моделируют процессы передачи сообщений и выполнения профилатик. Обе эти ветви нужно разместить в окне графического редактора агента Main. Свяжем с этими процессами сущности Message и Trouble, которые как и Main, должны являться агентами верхнего уровня.

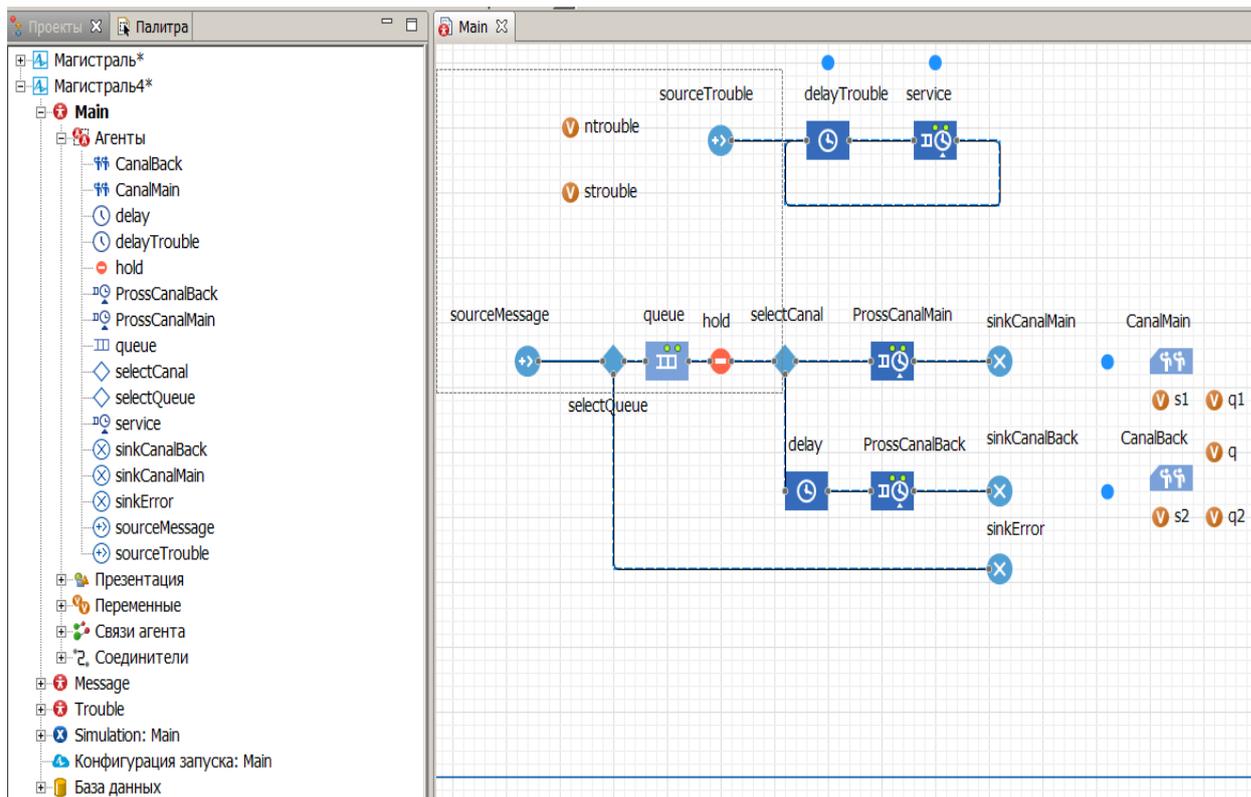


Рисунок 4

На рисунке (Рисунок 4) показана уже созданная диаграмма агента Main для модели Магистраль4 в окне графического редактора (вызывается окно двойным щелчком по иконке **Main** в окне Проекты).

Опишем этапы создания диаграммы. Откройте пустое окно графического редактора агента Main в создаваемой модели, дважды щелкнув по иконке **Main** в окне Проекты. Переключитесь на панель Палитры, откройте в ней Библиотеку моделирования процессов, если она сразу не открылась, щелчком по иконке  (Рисунок 5). Для добавления объекта на диаграмму надо щёлкнуть его мышью и перетащить в графический редактор. При перетаскивании объектов автоматически появляются соединительные линии между объектами. Эти линии должны соединять только порты, находящиеся с

правой или левой стороны иконок. Если не получится автоматическое соединение нужных объектов, дважды щёлкните начальный порт. Он станет зелёным. Протащите курсор к конечному порту. Он также станет зелёным. Для завершения процесса соединения щёлкните конечный порт.

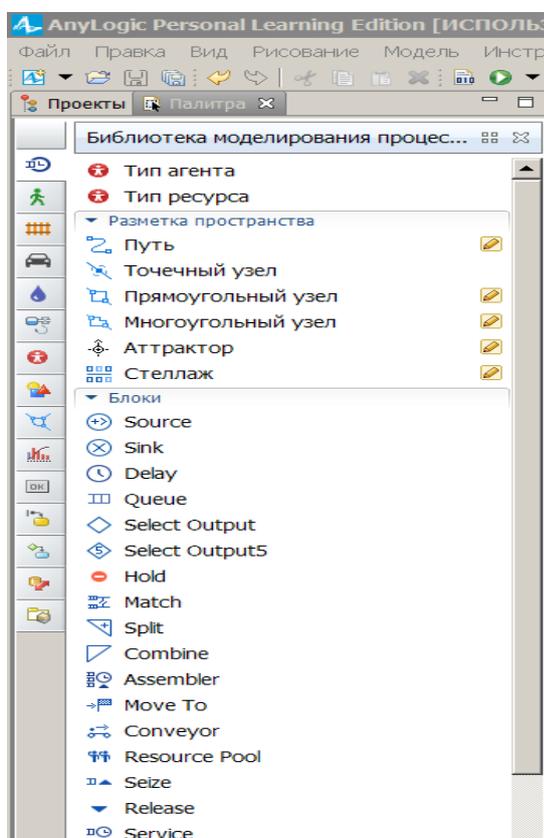


Рисунок 5

### Описание моделирования профилактики или сбоя в канале связи

На рисунке (Рисунок 6) показана ветвь диаграммы процессов, соответствующая агенту Trouble, создающему профилактику (или сбой) в системе, которую нужно создать. Здесь использованы объекты Source (вход), Delay (задержка), Service (захват-задержка-освобождение ресурса). Свойства объекта (как и любого другого элемента AnyLogic) можно изменить в панели Свойства, расположенной справа. Для изменения свойств объекта нужно будет предварительно щелчком мыши выделить его в графическом редакторе или в панели Проекты.

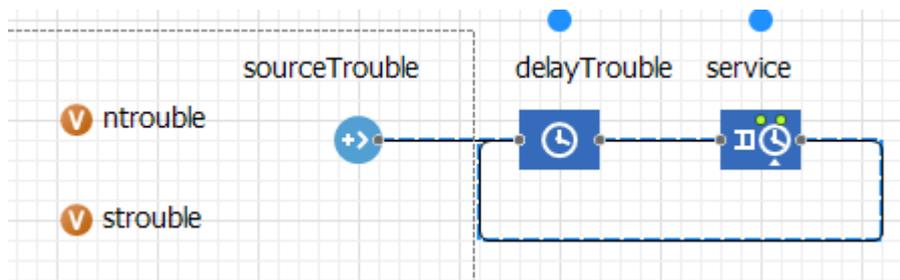


Рисунок 6

Создайте диаграмму процессов (синие блоки) первоначально без настройки свойств, как показано на рисунке (Рисунок 6). Далее переходим к описанию свойств объектов.

На рисунке (Рисунок 7) показано уже настроенное окно свойств объекта sourceTrouble (источник сбоев или профилактик) в соответствии с принятой логикой задачи. Чтобы вызвать это окно, щелкните по объекту в графическом редакторе. Заполните окно по образцу. Первоначально в поле Новый Агент (Рисунок 8) отображается стандартное имя Агент. Здесь нужно выбрать [создать другой тип](#). Появится окно мастера (Рисунок 9) создания агентов, в котором следует задать имя нового агента Trouble и далее следовать указаниям.

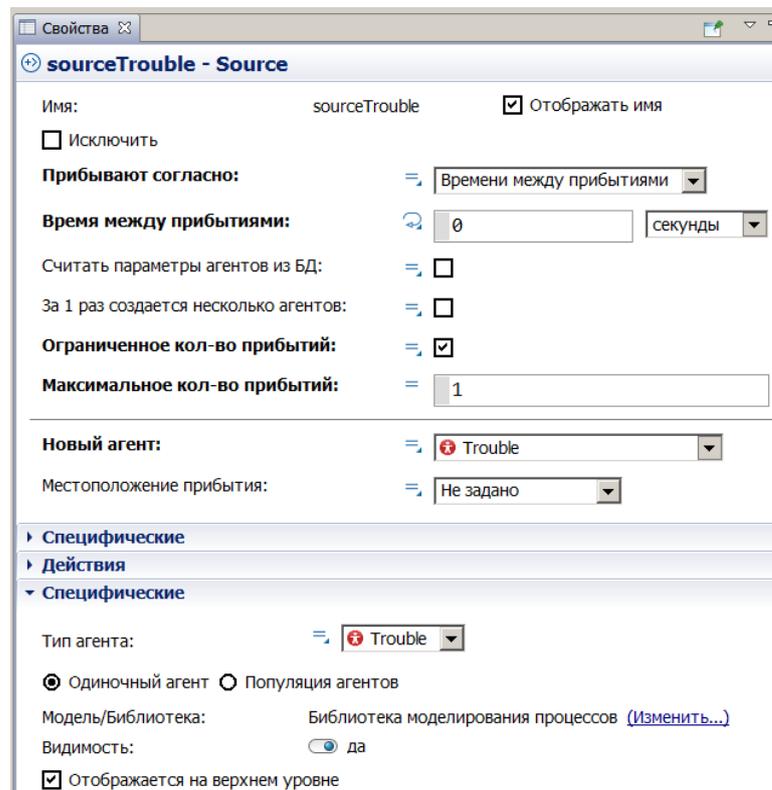


Рисунок 7

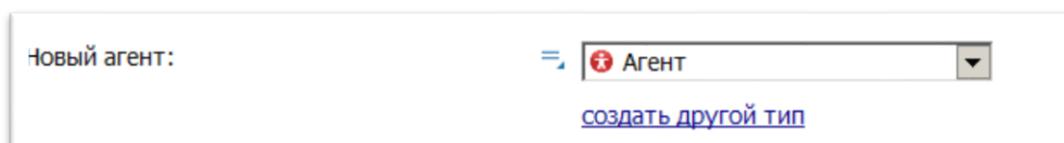


Рисунок 8

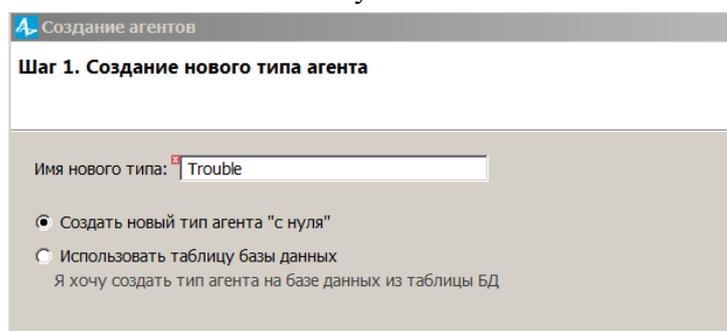


Рисунок 9

Для вновь созданного агента Trouble создается новое окно графического редактора. Если в ходе создания агента задавалась его анимационная картинка, то она будет отображаться в этом окне. При создании более сложных моделей в этом окне располагаются переменные и параметры агента и т.д.

Окно мастера создания агентов можно вызвать из окна Проекты, правой кнопкой мыши щелкнув по имени проекта, в котором создается новый агент, и выбрать Создать Тип агента.

Для продолжения работы нужно снова переключиться на окно Main в Графическом редакторе. Не забываем периодически сохранять модель.

Далее необходимо задать свойства остальных объектов диаграммы

На рисунке (Рисунок 10) показано уже настроенное окно свойств объекта delayTrouble (ожидание профилактики) в соответствии с условием задачи. Заполните это окно по образцу.

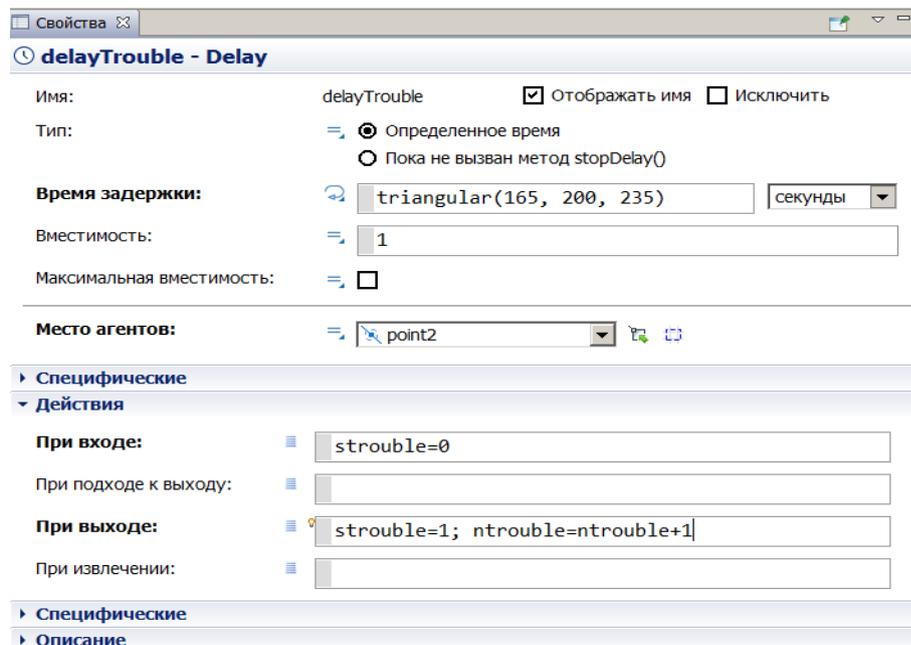


Рисунок 10

Для времени задержки выбран треугольный закон. В AnyLogic возможно задать и другие стандартные законы распределения. Рассмотрим как заполнить поле **Время задержки**. Чтобы не набирать вручную функцию `triangular(165, 200, 235)`, а выбрать ее из раскрывающегося списка, подведите курсор в начало поля ввода функции и нажмите комбинацию **Ctrl+пробел** (выпадающий список может появиться спустя некоторое время). Выберите из списка нужную функцию и измените ее аргументы в соответствии с условием.

Место агентов `point2` оставьте незаполненным. Это поле может быть заполнено только после добавления элементов разметки пространства на диаграмму для настройки анимации. В данной модели анимацию не будем создавать.

В свойстве Действия (здесь записываются операторы на Java) задайте значения переменной `strouble` сбоя в канале `CanalMain` и переменной `ntrouble`, используемой для подсчета числа сбоев, как показано на рисунке (Рисунок 10). Но этого для дальнейшей работы недостаточно. Переменные должны быть предварительно созданы. Это делается следующим образом: на панели Палитра выбрать вкладку Агент, щелкнув по иконке . Среди компонент агента следует выбрать компоненту  Переменная (Рисунок 11), перетащить ее в область диаграммы агента `Main` и расположить в каком-либо месте

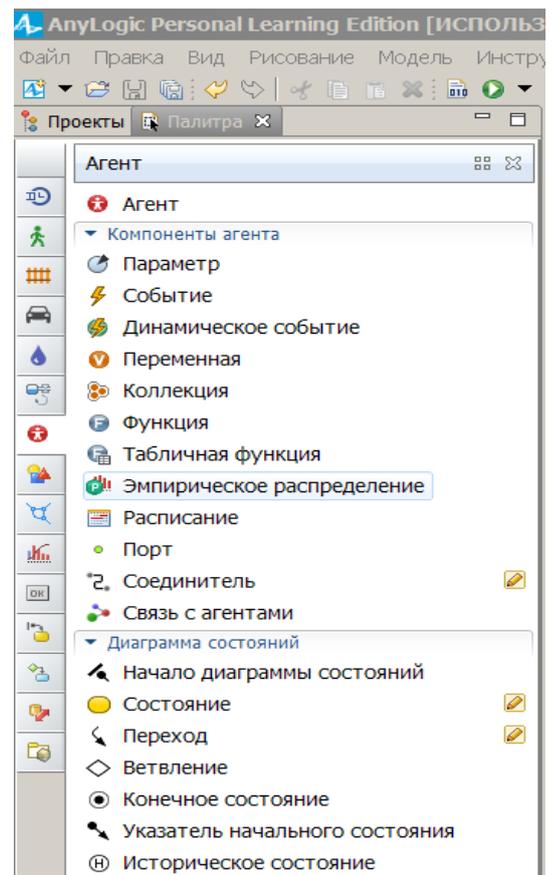


Рисунок 11

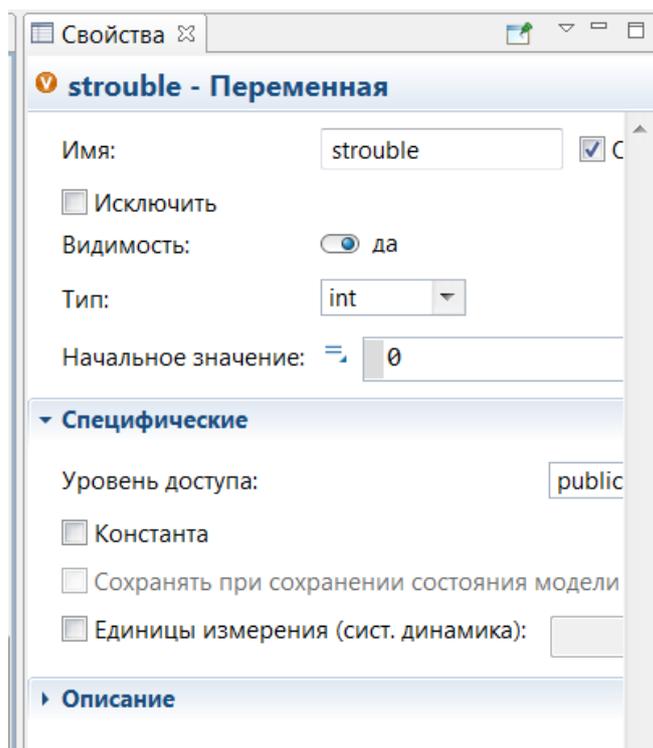


Рисунок 12

Далее нужно щелкнуть по объекту  `variable` в области диаграммы агента `Main`. В появившемся справа окне свойств (Рисунок 12) присвоить переменной имя `strouble`, тип и начальное значение `0`.

Аналогично создайте переменную `ntrouble`.

Не забываем периодически сохранять модель.

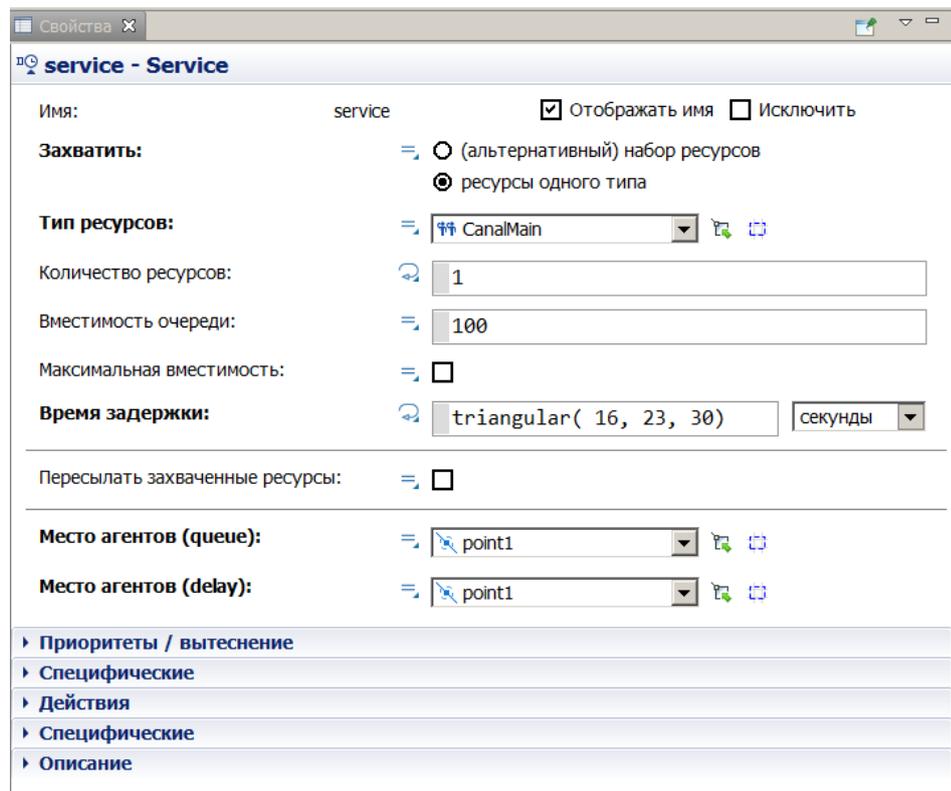


Рисунок 13

На рисунке (Рисунок 13) показано уже настроенное окно свойств объекта `service` (профилактика) в соответствии с условием задачи. Чтобы вызвать это окно, щелкните по объекту. Чтобы выбрать Тип ресурсов (из раскрывающегося списка), нужно предварительно эти ресурсы создать. В строке Количество ресурсов нужно указать, сколько ресурсов задействованы в процессе `service` для обслуживания транзакта.

Опишем, как создаются ресурсы. В нашем случае это ресурсы `CanalMain` и `CanalBack`. В библиотеке моделирования процессов выбираем объект  `Resource Pool`, перетаскиваем его в графическую область диаграммы агента `Main`. В графической области появится объект , размещаем его в произвольном месте.

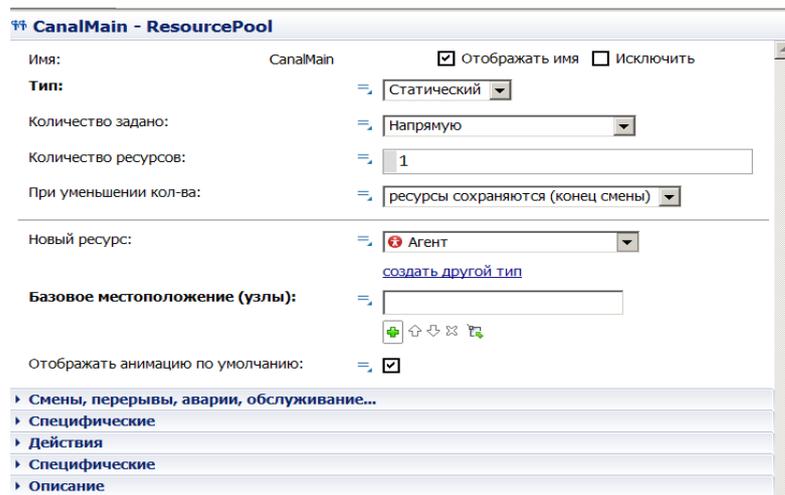


Рисунок 14

Щелчком по этому объекту вызываем окно свойств ресурса (Рисунок 14), в котором вводим имя CanalMain и количество ресурсов – 1 (один канал). Если имеется несколько ресурсов данного типа, например, несколько компьютеров или несколько рабочих для параллельного обслуживания транзакта, то нужно указывать это число. В окне ресурса можно добавить новый ресурс-агент, но мы этого не будем делать, в поле Новый ресурс оставляем стандартное имя Агент.

Аналогично создаем второй ресурс CanalBack, он понадобится во второй диаграмме.

Поля Место агентов, которые нужны для анимации, не заполняем.

Так как профилактики периодически повторяются, то рассматриваемая ветвь диаграммы процессов циклическая. Это задается соединением выходного порта объекта service с входным портом объекта delayTrouble. Чтобы изменить форму коннектора, следует предварительно создать на нем дополнительные узлы (по щелчку мыши).

### Описание моделирования передачи сообщений

На следующем рисунке показана ветвь диаграммы процессов, описывающая процесс обработки сообщений. С этой диаграммой связан агент Message. Создайте диаграмму, пока без настройки свойств, как показано на рисунке (**Ошибка! Источник ссылки не найден.**), используя блоки Библиотеки моделирования процессов.

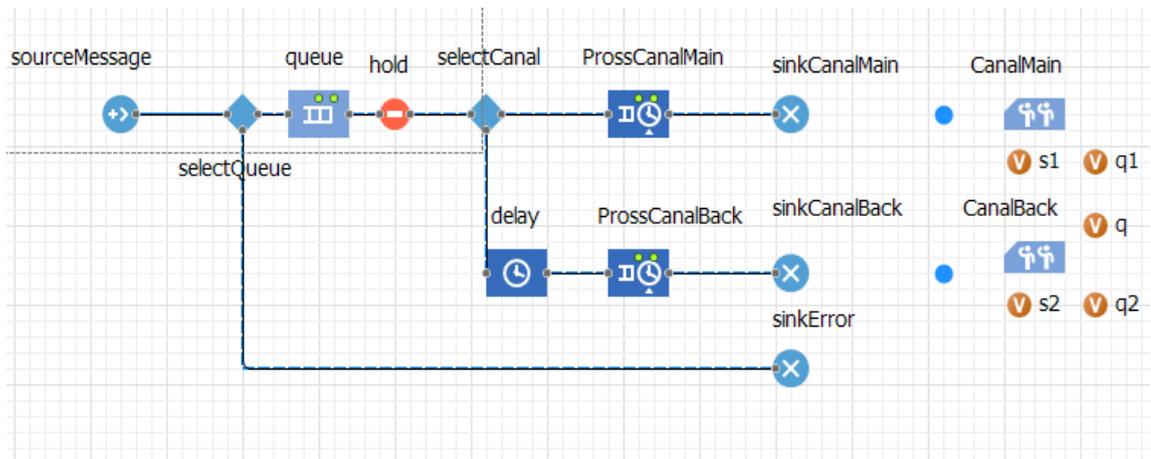


Рисунок 15

Далее заполняются окна свойств для каждого объекта диаграммы. На рисунке (Рисунок 16) представлено уже заполненное окно свойств объекта sourceMessage (источник сообщений).

В AnyLogic блок-шлагбаум Hold не имеет настроек очереди, хотя он способен ее создавать. Для определения длины очереди нужно перед блоком Hold разместить объект Queue (очередь).

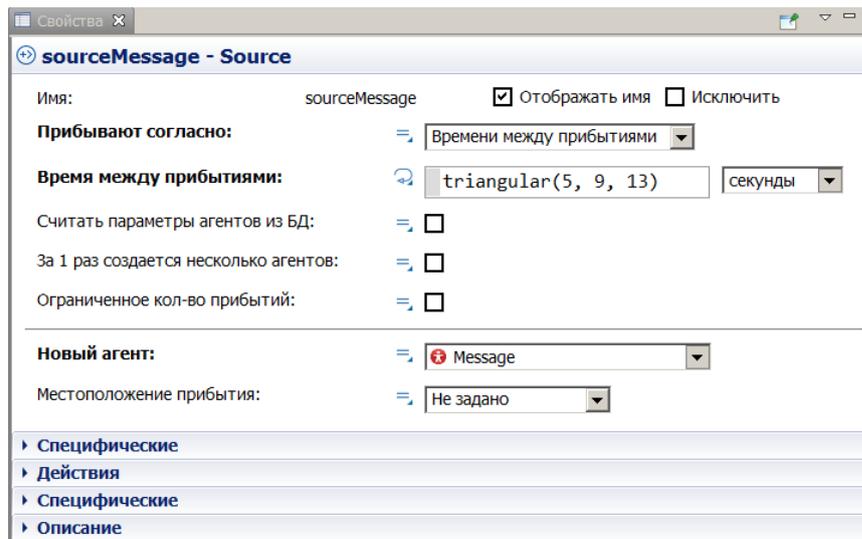


Рисунок 16

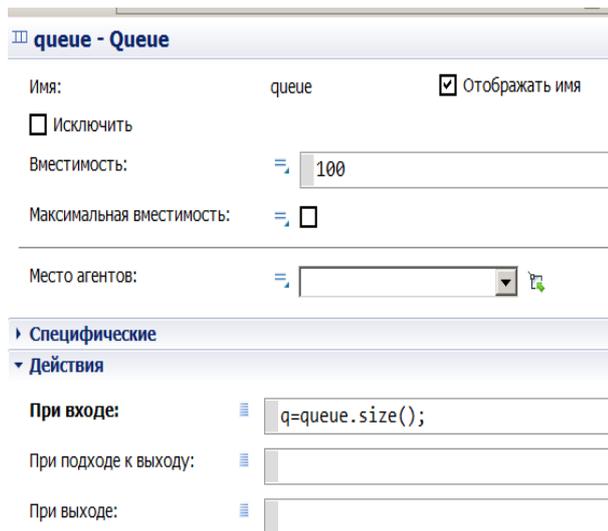


Рисунок 17

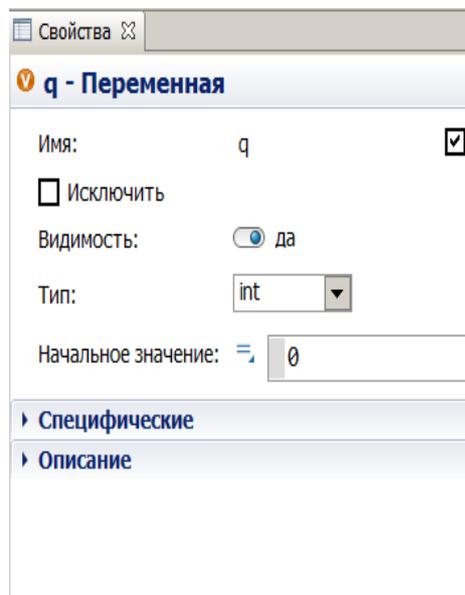
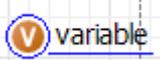


Рисунок 18

На рисунке (Рисунок 17) представлено окно свойств объекта queue. Оставьте допустимую вместимость очереди равной 100. В секции Действия следует переменной q – длина очереди, присвоить значение текущей длины очереди с помощью функции size(). Используйте Ctrl+пробел для вызова подсказки или наберите в поле имя функции. Сведения о функциях AnyLogic можно найти в справочных материалах.

Переменная q должна быть предварительно создана: на область диаграммы агента Main из вкладки Агент Палитры перетащите иконку 

. В окне свойств поменяйте имя, тип, начальное значение переменной (Рисунок 18).

В окне свойств объекта Hold (Рисунок 19) в секции Действия следует задать оператор `hold.setBlocked(true)` блокировки шлагбаума Hold при занятости ресурсов и режим пропуска сообщений по одному. В условиях используются переменные `s1,s2` (создайте эти переменные с помощью , которые отвечают за состояния ресурсов. Значение `s1=1` означает, что ресурс CanalMain занят, `s1=0` означает что ресурс CanalMain свободен и т.д. Значения этим переменным будут присвоены далее в блоках `ProssCanalMain` (обработка сообщения в основном канале), `ProssCanalBack` (обработка сообщения в резервном канале).

Переменная `strouble` отвечает за наличие-отсутствие профилактики или сбоя в основном канале, она получает значение, как описано ранее, в блоке `delayTrouble`.

В условном операторе `if` используются логические операторы `||`, `&&`, `==`, означающие логические операции соответственно или, и, равно.

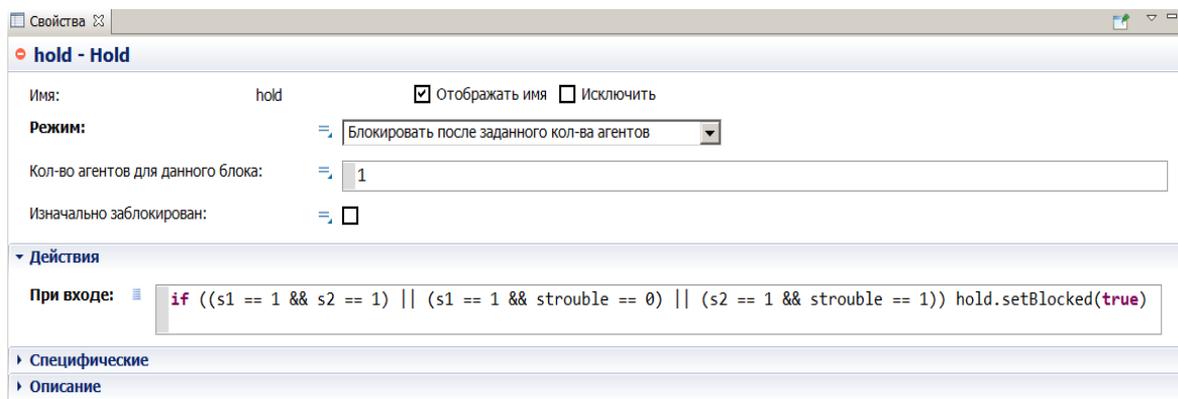


Рисунок 19

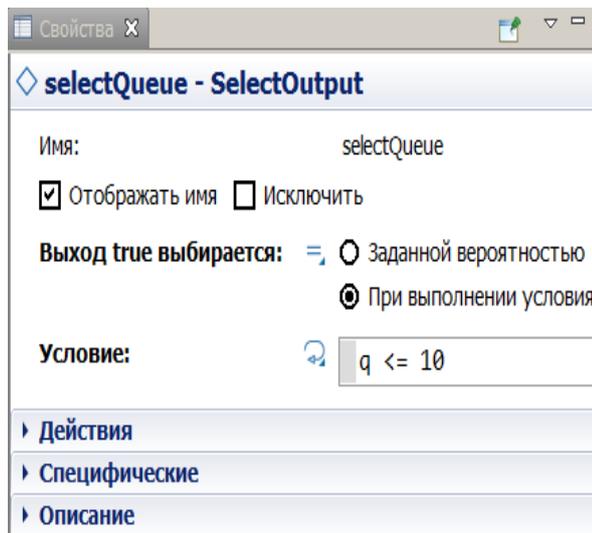


Рисунок 20

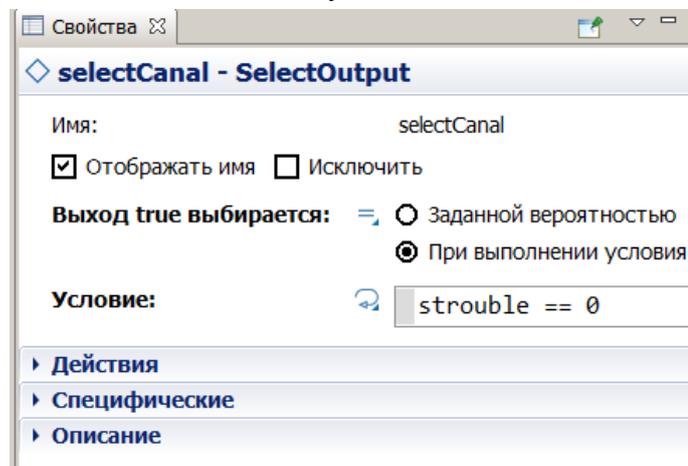


Рисунок 21

На рисунке (Рисунок 20) показано окно свойств объекта `selectQueue`, который обеспечивает разветвление: сообщения теряются, если длина очереди больше 10. На рисунке (Рисунок 21) показано окно свойств объекта `selectCanal`, который обеспечивает разветвление: сообщения обрабатываются основным каналом, если сбоя нет.

На рисунке (Рисунок 22) показано окно свойств объекта `ProssCanalMain`. При захвате ресурса присваивается `s1=1`, при освобождении – `s1=0`, и записывается оператор разблокировки `Hold: hold.setBlocked(false)`. Здесь также определяется переменная `q1` – длины очереди для процесса `ProssCanalMain`.

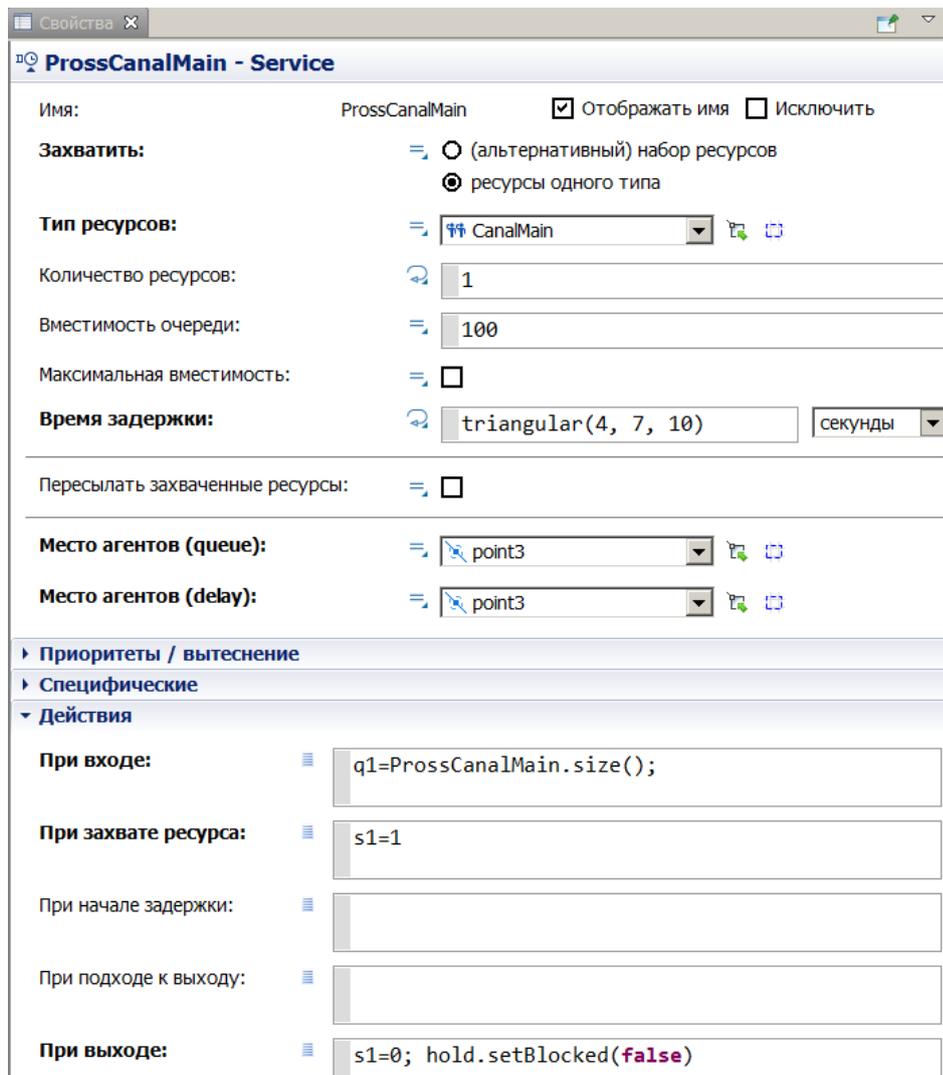


Рисунок 22

Окно свойств объекта ProssCanalBack заполняется аналогично.

Окно свойств объекта delay представлено на Рисунок 23. В окнах свойств модулей завершения процессов изменено стандартное имя соответственно на имена sinkCanalMain, sinkCanalBack, sinkError (Рисунок 24).

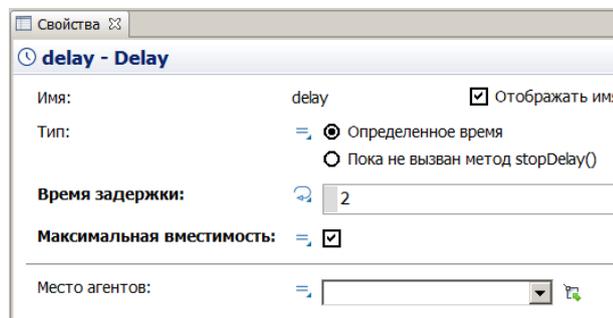


Рисунок 23

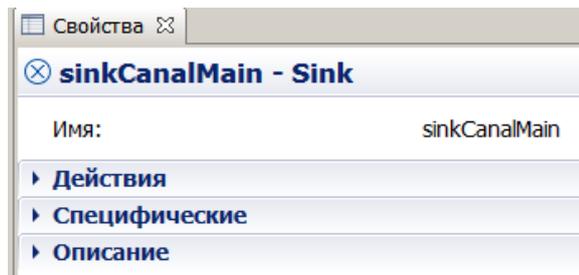


Рисунок 24

### Анимация в модели

В AniLogic предоставляются большие возможности для создания 2D, 3D анимационных моделей. При создании анимации в окне Графического редактора агента Main добавляются компоненты разметки пространства (Рисунок 25). Элементы разметки в проекте отображаются в окне Проекты в классе – презентации агента. Для материальных агентов можно задать анимационные картинки, которые размещаются на диаграммах этих агентов. Встроенная библиотека картинок доступна в Панели на вкладке 3D объекты. Для создания анимации используются также объекты вкладки Презентация Палитры (Рисунок 26). Здесь же можно настроить собственные картинки.

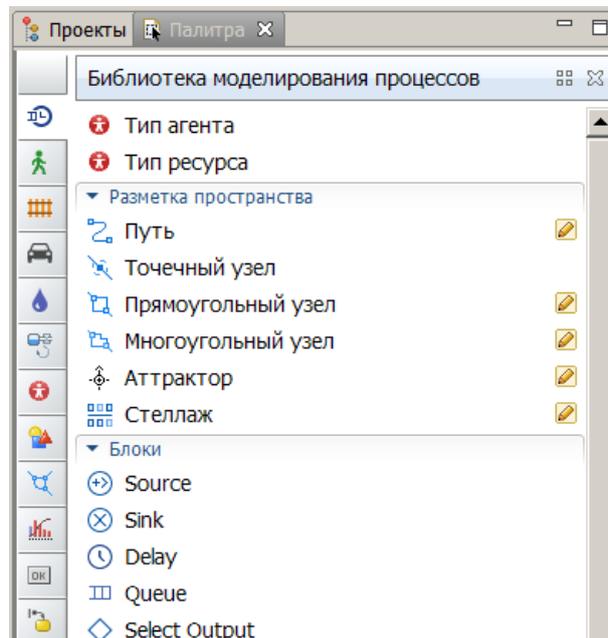


Рисунок 25

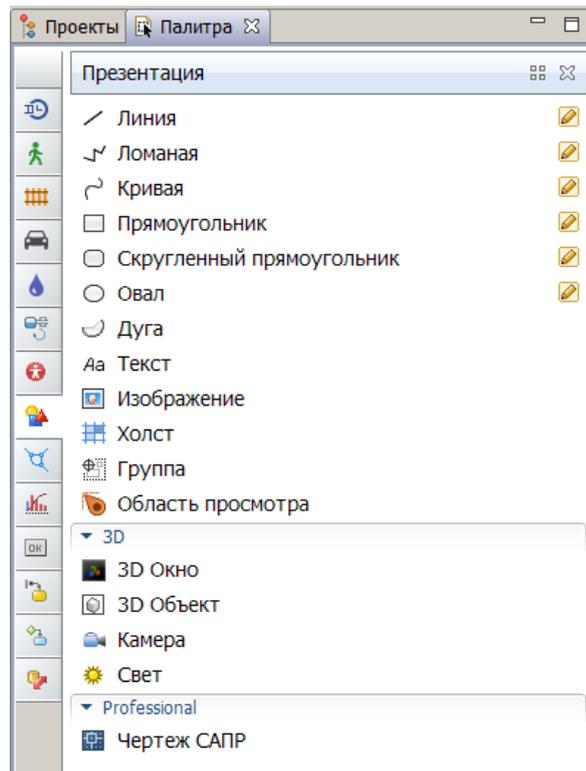


Рисунок 26

В рассматриваемой модели передачи сообщений нет смысла создавать дополнительную анимацию. Уже по созданной диаграмме и переменным, размещенным в области диаграммы, при прогоне модели динамика процесса ясна. В связи с этим в этом пособии создание анимации рассматриваться не будет. Можно самостоятельно ознакомиться с соответствующими инструментами по пособиям и примерам, предоставляемым разработчиками AnyLogic вместе с программным продуктом, а также воспользоваться материалами [1,2].

### Запуск модели на выполнение

Прежде чем запускать модель, нужно настроить окно свойств Простого Эксперимента, которое появляется по щелчку по иконке  Simulation: Main в окне Проекты (Рисунок 27). Обязательно нужно выбрать Агент верхнего уровня Main, так как диаграмма процесса создавалась в окне графического редактора именно этого агента. В настройке, показанной на рисунке, предполагается, что эксперимент длится в течение 3600 сек. модельного времени. Если не указать этого, то по умолчанию время прогона будет не ограничено.

Если единицы модельного времени должны быть другими: минуты, часы, то нужно в окне свойств  Simulation: Main переключиться на вкладку

установки модельного времени (внизу окна свойств) и установить нужные единицы модельного времени (Рисунок 28).

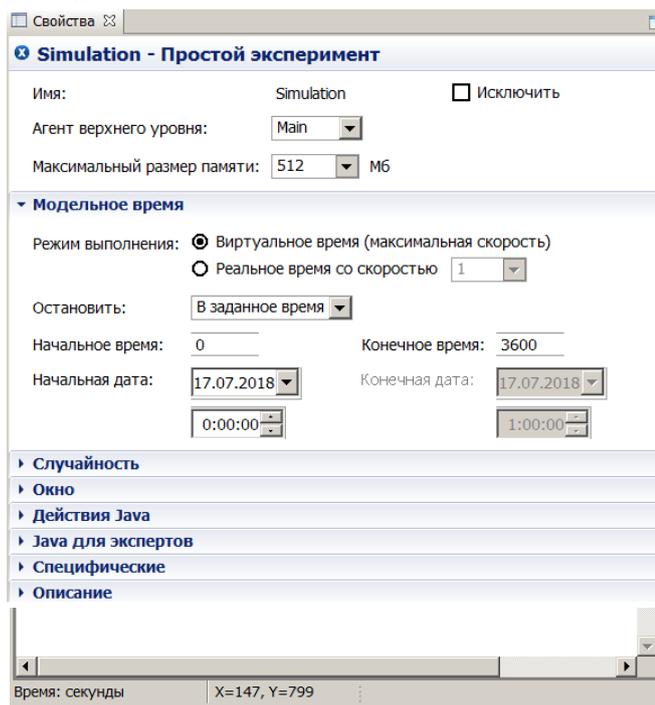


Рисунок 27

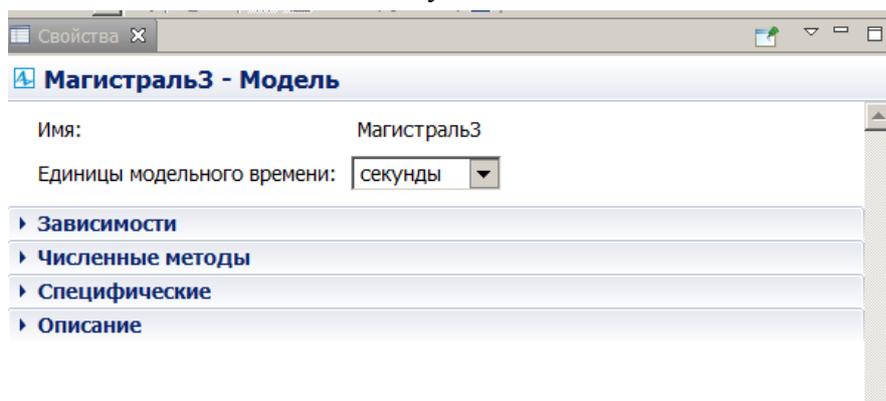


Рисунок 28

Для запуска модели нужно выбрать имя эксперимента в списке, отображаемом при щелчке по черной стрелке  на панели инструментов. По умолчанию по зеленой стрелке запускается эксперимент, который запускался предыдущим. После щелчка по стрелке появляется окно презентации эксперимента (Рисунок 29).

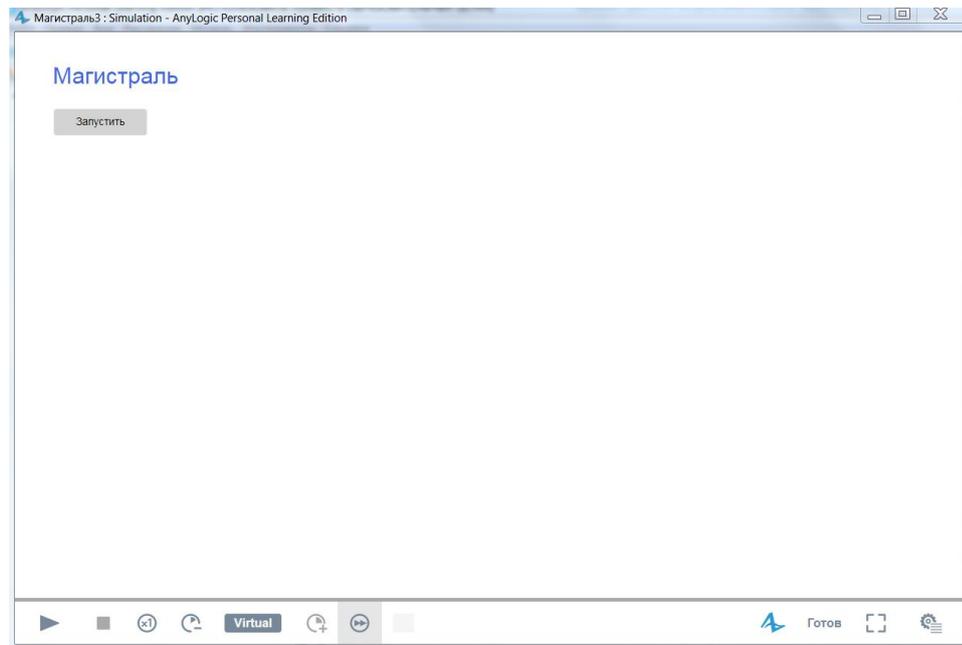
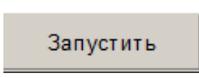


Рисунок 29

Чтобы регулировать скорость эксперимента, нужно нажать кнопку переключения  с виртуального на реальное время и запустить модель, нажав кнопку  или . Скорость выполнения (замедлить, ускорить) модели можно регулировать с помощью кнопок    на нижней панели окна презентации.

На рисунке (Рисунок 30) представлены результаты одного прогона с параметрами, определенными в постановке задачи. Здесь использовались анимационные картинки, создание которых подробно не описывалось. Загрузка основного канала по времени –79%, резервного – 7%, количество профилактик – 15, количество потерянных сообщений 0.

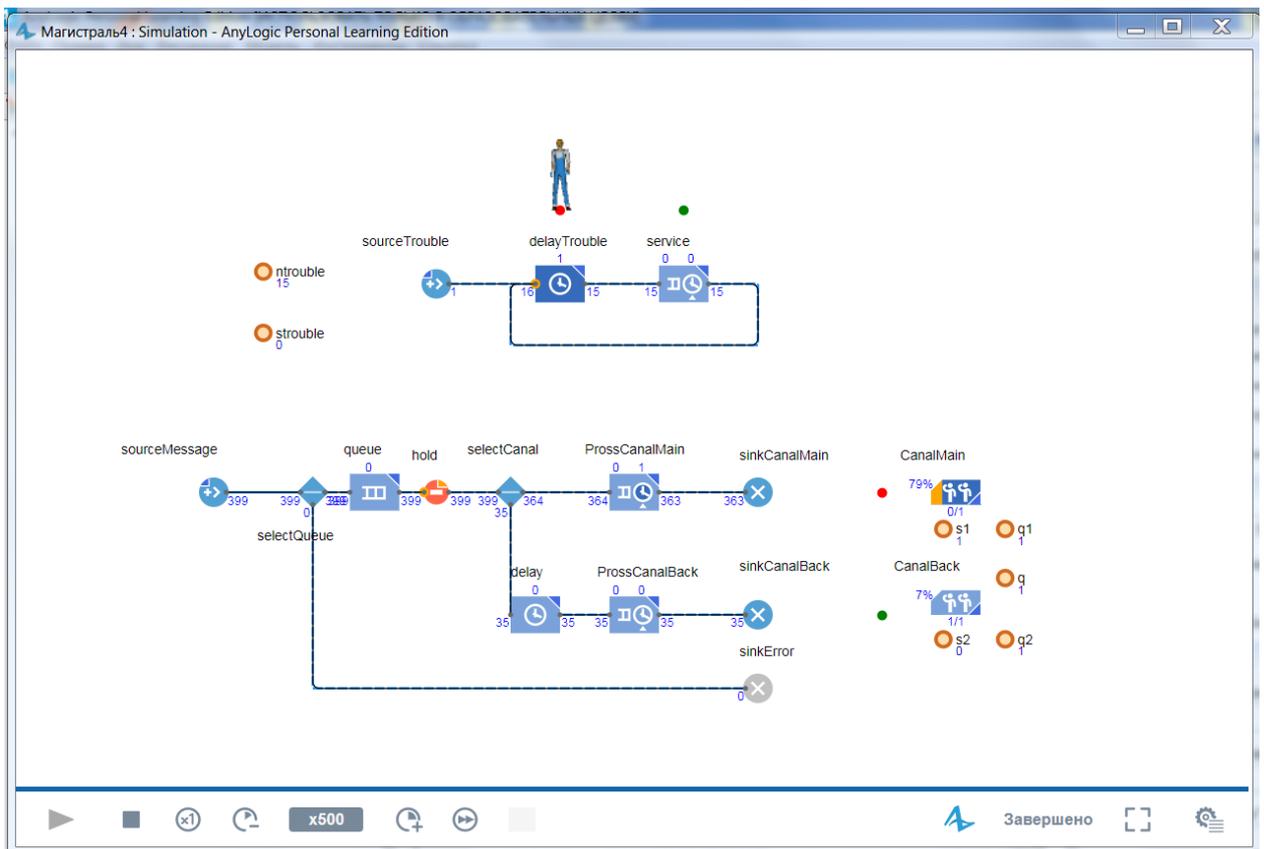


Рисунок 30

### Задания контрольной работы (базовый уровень)

Выполните моделирование информационного процесса «Магистраль передачи данных» в среде AnyLogic. Проанализируйте результаты работы модели, изменяя произвольно параметры и переменные модели.

Внесите изменения в модель в соответствии со своим вариантом (галочками отмечены объекты, в которые вносятся изменения). В качестве изменений могут быть: другие законы распределения (показательный, нормальный, равномерный и др.), значения параметров и переменных, названия блоков, добавление новых блоков, графиков, диаграмм, элементов сбора статистики в окне презентации и т.д.

Выполните анализ полученных результатов в соответствии с внесенными изменениями, оформите контрольную в виде отчета (15-20 стр. А4), включающего титульный лист, лист задания (номер варианта и исходные данные), введение, основную часть (этапы разработки модели), анализ результатов работы модели, список используемых источников.

№ по журналу	sourceTrouble	delayTrouble	service	sourceMessage	selectQueue	queue	hold	selectCanal	ProssCanalMain	sinkCanalMain	delay	ProssCanalBack	sinkCanalBack	sinkError
1	✓		✓			✓		✓			✓			✓
2	✓	✓		✓			✓		✓			✓		
3		✓	✓		✓			✓		✓			✓	
4		✓		✓		✓			✓		✓			✓
5	✓		✓		✓		✓			✓		✓		
6			✓	✓		✓		✓			✓		✓	
7	✓	✓		✓					✓			✓		✓
8		✓	✓		✓			✓		✓	✓			
9	✓			✓			✓	✓	✓		✓			
10	✓	✓				✓		✓		✓		✓		
11		✓				✓			✓	✓	✓			
12	✓		✓		✓		✓		✓		✓			
13			✓	✓		✓		✓		✓			✓	
14	✓			✓					✓		✓	✓		✓
15	✓		✓		✓			✓			✓			✓
16	✓	✓		✓			✓		✓			✓		
17		✓			✓		✓	✓		✓			✓	
18				✓		✓			✓		✓	✓		✓
19	✓				✓		✓		✓	✓		✓		
20			✓	✓	✓	✓		✓					✓	
21		✓		✓	✓				✓			✓		✓
22			✓		✓		✓	✓		✓	✓			
23	✓			✓				✓	✓		✓	✓		
24	✓	✓				✓				✓	✓	✓		
25		✓		✓	✓			✓	✓	✓				

## **Задания контрольной работы (повышенный уровень)**

Разработать в среде AnyLogic математическую модель из выбранной самостоятельно предметной области или из перечня приведенного ниже. Оформить отчет в соответствии с указанными требованиями.

### *Примеры постановок задач*

#### **1. Имитационное моделирование работы распределенного банка данных**

Распределенный банк данных системы сбора информации организован на базе ЭВМ, соединенных дуплексным (возможна одновременная передача данных в обоих направлениях) каналом связи. Поступающий запрос обрабатывается на первой ЭВМ, и с вероятностью  $p_1$  % необходимая информация обнаруживается на месте. В противном случае необходима посылка запроса во вторую ЭВМ.

Запросы имеют разные приоритеты. Запросы с большим приоритетом обрабатываются (первичная обработка) в два раза быстрее. Запросы поступают через  $t_1 \pm 0,3t_1$  с, первичная обработка запроса занимает  $t_2$  с для первых  $n_1$  запросов, для остальных  $t_5$ , выдача ответа требует  $t_3 \pm 0,3t_3$  с, передача по каналу связи занимает  $t_4$  с. Временные характеристики второй ЭВМ аналогичны первой.

Смоделировать прохождение запросов в течение  $t_5$  часов.

Определить количество обработанных запросов в зависимости от приоритета, среднее время обработки и ожидания обработки, необходимую емкость накопителей перед ЭВМ, обеспечивающую безотказную работу системы, и функцию распределения времени обслуживания заявки.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы банка данных. Определить оптимальные значения параметров из условия улучшения его работы. При реализации проекта использовать средства визуализации.

#### **2. Имитационное моделирование обработки сигналов**

Система обработки сигналов содержит мультиплексный канал и три мини-ЭВМ. Сигналы от датчиков поступают на вход канала через интервалы времени  $t_1 \pm 0,3t_1$  мкс. В канале они буферизируются и предварительно обрабатываются в течение  $t_2 \pm 0,3t_2$  мкс. Затем они поступают на обработку в ту мини-ЭВМ, где имеется наименьшая очередь.

Каждая из ЭВМ может останавливаться на профилактику один раз в сутки по расписанию. Время ее восстановления –  $t_3$  часа. В случае остановки одной из ЭВМ, информация обрабатывается той ЭВМ, в которой очередь

меньше. Время обработки сигнала в любой мини ЭВМ равно  $t_4$  мкс.

Емкости входных накопителей во всех ЭВМ рассчитаны на хранение величин  $n_1$  сигналов. В случае переполнения появляется сигнал «переполнение».

Смоделировать процесс обработки сигналов и работу системы в течение 1000 мкс. Определить количество обработанных сигналов, среднее время обработки и время ожидания обработки сигнала. Определить необходимые емкости накопителей из условия отсутствия переполнения.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы системы. Определить оптимальные значения параметров из условия улучшения ее работы. При реализации проекта использовать средства визуализации.

### ***3. Имитационное моделирование обработки заданий вычислительной системой***

Вычислительная система включает три ЭВМ. В систему в среднем через  $t_1$  с поступают задания, которые попадают в очередь на обработку к первой ЭВМ, где они обрабатываются около  $t_2$  с. После этого задание поступает или во вторую, или в третью ЭВМ, где меньше очередь. Вторая ЭВМ может обработать задание за  $t_3 \pm 0,3t_3$  с, а третья – за  $t_4 \pm 0,3t_4$  с.

Каждая из ЭВМ может останавливаться на профилактику один раз в сутки по расписанию. Время ее восстановления –  $t_5$  часов. В случае остановки одной из ЭВМ, информация обрабатывается на подключаемой резервной ЭВМ. Время обработки на резервных ЭВМ в 3 раза дольше, чем на основной.

Смоделировать  $t_6$  ч работы системы.

Определить количество обработанных заданий, среднее время обработки и ожидания, необходимую емкость накопителей перед всеми ЭВМ, коэффициенты загрузки каждой ЭВМ и функцию распределения времени обслуживания заданий.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы системы. Определить оптимальные значения параметров из условия улучшения ее работы. При реализации проекта использовать средства визуализации.

### ***4. Имитационное моделирование обслуживания пользователей машинного зала***

В машинный зал с интервалом времени  $t_1 \pm 0,3t_1$  мин (нормальный закон) заходят пользователи, желающие произвести расчеты на ЭВМ. В зале имеется одна ЭВМ, работающая в однопрограммном режиме. Время, необходимое для решения задач, характеризуется интервалом  $t_2 \pm 0,3t_2$  мин.

ЭВМ может останавливаться на профилактику один раз в сутки по расписанию. Время ее восстановления –  $t_3$  часа. В случае остановки ЭВМ, информация обрабатывается на резервной ЭВМ, время обработки на которой в 2 раза дольше.

В машинном зале не допускается, чтобы более  $n_1$  пользователей ожидали своей очереди на доступ к ЭВМ.

Смоделировать процесс обслуживания  $n_2$  пользователей. Подсчитать число пользователей, не нашедших свободного места в очереди. Определить среднее число обслуженных пользователей, коэффициент загрузки ЭВМ, среднее время обслуживания.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы ЭВМ. Определить оптимальные значения параметров из условия улучшения ее работы. При реализации проекта использовать средства визуализации.

### ***5. Имитационное моделирование управления технологическим процессом***

В вычислительную машину, работающую в системе управления технологическим процессом, через каждые  $t_1 \pm 0,3t_1$  (Закон показательный) с поступает информация от датчиков. До обработки на ЭВМ сигналы накапливаются в памяти. Продолжительность обработки сообщений на ЭВМ –  $t_2 \pm 0,3t_2$  с.

ЭВМ может останавливаться на профилактику один раз в сутки. Время ее восстановления –  $t_3$  часа. В случае остановки ЭВМ, информация обрабатывается на резервной ЭВМ, время обработки на которой в 2 раза дольше.

Сообщения считаются потерянными, если они ожидали обработки более  $t_4$  с.

Смоделировать процесс поступления в ЭВМ в течение времени  $t_4$ . Подсчитать число потерянных сообщений и обработанных, определить коэффициент загрузки ЭВМ (основной ЭВМ и резервной).

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы ЭВМ. Определить оптимальные значения параметров из условия улучшения ее работы. При реализации проекта использовать средства визуализации.

### ***6. Имитационное моделирование работы информационно-поисковой библиографической системы***

Информационно-поисковая библиографическая система построена на базе двух ЭВМ и имеет один терминал для ввода и вывода информации.

Вероятность обращения к первой ЭВМ –  $p_1$ , ко второй –  $p_2$ .

Каждая из ЭВМ может останавливаться на профилактику один раз в течение 8 часов. Время ее восстановления -  $m$  часов. В случае остановки одной из ЭВМ, информация обрабатывается другой ЭВМ.

Пользователи обращаются к терминалу каждые  $t_1 \pm 0,3t_1$  мин. Если в очереди к терминалу ожидают  $n_1$  пользователей, то вновь прибывший пользователь получит отказ. Поиск информации на первой ЭВМ продолжается  $t_2 \pm 0,3t_2$  мин, на второй –  $t_3 \pm 0,3t_3$  мин. Смоделировать процесс работы системы за 8 часов.

Определить среднюю и максимальную длину очереди к терминалу, а также коэффициенты загрузки ЭВМ.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы системы. Определить оптимальные значения параметров из условия улучшения работы системы. При реализации проекта использовать средства визуализации.

### ***7. Имитационное моделирование работы вычислительного устройства***

Вычислительное устройство имеет в своем составе два процессора, соединенные с общей оперативной памятью. В режиме нормальной эксплуатации задания выполняются на первом процессоре, а второй считается резервным. Первый процессор характеризуется низкой надежностью и работает безотказно лишь в течение  $t_1 \pm 0,3t_1$  мин. Если отказ происходит во время решения задания, то производится включение второго процессора, который продолжает решение задание и всех последующих заданий до восстановления первого процессора.

Это восстановление происходит за  $t_2 \pm 0,3t_2$  мин, после чего начинается решение очередного задания на первом процессоре, а резервный выключается. Задания поступают на устройство каждые  $t_3 \pm 0,3t_3$  мин и решаются за  $t_4 \pm 0,3t_4$  мин.

Смоделировать процесс работы устройства в течение  $t_5$  ч. Определить коэффициент загрузки резервного процессора, максимальную длину очереди заданий, время обслуживания.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы устройства. Определить оптимальные значения параметров из условия улучшения работы системы. При реализации проекта использовать средства визуализации.

### ***8. Имитационное моделирование работы устройства обслуживания терминалов***

ЭВМ обслуживает три терминала по круговому циклическому алгоритму, предоставляя каждому терминалу  $t_1$  с. Если в течение этого времени задание обрабатывается, то обслуживание завершается, если нет, то остаток задачи становится в специальную очередь, которая использует свободные циклы терминалов, т.е. задача обслуживается, если на каком-либо терминале нет заявок. Заявки на терминалы поступают через  $t_1 \pm 0,3t_1$  с и имеют длину  $n_1 \pm 0,3n_1$  знаков. Скорость обработки заданий ЭВМ составляет  $n_2$  знаков/с.

Смоделировать  $t_2$  ч работы ЭВМ. Определить загрузку ЭВМ, параметры очереди неоконченных заданий.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы устройства. Определить оптимальные значения параметров из условия улучшения работы системы. При реализации проекта использовать средства визуализации.

### ***9. Имитационное моделирование процесса предпродажной подготовки компьютера***

В салон по сборке компьютеров со средним временем в  $t_1$  минут приходит 1 клиент (что определяется по экспоненциальному закону) чтобы выбрать компьютер (ПК). Выбор ПК осуществляется в течении  $t_1 \pm 0,3t_1$  минут. Далее клиенту предлагается подтвердить свой выбор. По статистике  $p_1\%$  удовлетворенны своим выбором и готовы купить ПК. После появления заявки на сборку ПК, обязанности делятся между 2мя отделами: по сборке системного блока подбора, соответствующих монитора и периферийных устройств.

Отдел по сборке системного блока осуществляет единичный заказ в течение  $t_2 \pm 0,3t_2$  минут, в то время как, отдел по подбору соответствующих монитора и периферийных устройств делает свой единичный заказ в течение  $t_3 \pm 0,3t_3$  минут. Далее из этих составляющих на складе собираются компьютеры, которые в последствии будут проданы их владельцам.

Определить необходимое минимальное количество сотрудников в каждом отделе, необходимых для исполнения максимального количества заказов за  $t_4$  мин. (изначально в каждом отделе может одновременно работать только один сотрудник).

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на процесс подготовки компьютера. Определить оптимальные значения параметров из условия улучшения процесса. При реализации проекта использовать средства визуализации.

### ***10. Имитационное моделирование работы сборочного цеха***

В сборочный цех поступают детали трех видов. Детали первого типа (Д1) поступают через  $t_1 \pm 0,3t_1$  мин. Детали второго типа (Д2) через  $t_2 \pm 0,3t_2$  мин. Детали третьего типа (Д3) –  $t_3$  мин. Как только сборщику поступают три

детали (создать модели в случае случаи объединения трех любых деталей и трех различных), он производит монтаж готового изделия за  $t_4$  мин. Из собранных изделий  $p_1$  % бракованные.

Если изделие бракуется в первый раз, то оно поступает на повторный монтаж к сборщику. Если изделия бракуются 2 раза, то они идут в отходы (10 мин). Не бракованные изделия упаковываются по  $n_1$  штук за  $t_5$  минуты упаковщиком и складировются. Один раз в  $t_6$  часов из гаража выезжает грузовик и забирает со склада все упаковки.

Смоделировать 8-часовой рабочий день.

Построить модель согласно заданию и выполнить следующие задания:

Определить каждому типу деталей свою анимационную картинку. Определить анимационную картинку готовому изделию и упакованному изделию. Задать анимационную картинку ресурсам «Сборщик» и «Упаковщик», когда они свободны и заняты.

Собрать статистику по бракованным изделиям (отходы и один раз бракованные), количеству упаковок на складе, по загруженности ресурсов «Сборщик» и «Упаковщик».

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы сборочного цеха. Определить оптимальные значения параметров из условия улучшения работы. При реализации проекта использовать средства визуализации.

## ***11. Имитационное моделирование работы компании***

В компанию поступают запросы  $t_1 \pm 0,3t_1$  Поступающий запрос обрабатывается двумя сотрудниками, причем первый сотрудник обрабатывает  $p_1$  % запросов, второй обрабатывает остальные запросы. Первичная обработка запроса занимает  $t_2$  минуты, выдача ответа требует  $t_2 \pm 0,3t_2$  мин., как у первого, так и у второго сотрудника. Каждый из сотрудников имеет перерыв по расписанию

Смоделировать прохождение запросов в течение рабочего дня.

Определить количество запросов, обработанных каждым сотрудником за день, среднее время обработки запроса, длину очереди.

Дополнительные задания

1. В систему первоначально поступают сущности в виде телефонных звонков, а затем к первому сотруднику приходят в виде отчетов, а ко второму сотруднику – в виде дискет.

2. Измените модель: первые  $n_1$  запросов идут к первому сотруднику на обработку, остальные – ко второму.

3. Измените модель: первые  $t_4$  часа все запросы идут ко второму сотруднику на обработку, остальные – к первому.

4. Измените модель: первые  $n_2$  запросов проходят первичную обработку  $t_5$  мин., остальные –  $t_6$  минут.

5. Измените модель: на обработку поступают 2 вида запросов (телефонные звонки и письма). Причем при первичной обработке у телефонных звонков приоритет выше, чем у писем.

6. Создайте анимационные картинки ресурсам, когда они свободны и заняты.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы компании. При реализации проекта использовать средства визуализации.

## ***12. Имитационное моделирование полетов рейсов самолетов***

Клиенты, желающие приобрести билет на данный рейс, приходят в кассу аэропорта в среднем через  $t_1 \pm 0,3t_1$ , причем  $p_1$  % из них приобретают билеты в первый класс,  $p_2$  % – во второй класс, а остальные вообще отказываются приобретать билеты и уходят.

Время вылета самолета определяется его полной загрузкой, т. е. самолет вылетит только при наличии  $n_1$  пассажиров первого класса и  $n_2$  пассажиров второго класса. Время ожидания рейса  $p_2$  % пассажирами ограничено  $t_7$ , остальные пассажиры могут ждать неограниченно.

Самолеты данного направления прибывают в аэропорт в среднем раз в  $t_2$ - $t_3$  часов, максимальное количество самолетов в аэропорту –  $n_2$ . После прилета самолет выгружает пассажиров  $t_6 \pm 0,3t_6$  мин., затем проходит  $t_5 \pm 0,3t_5$  мин техобслуживание.

Время полета до станции назначения занимает в среднем  $(t_4 \pm 0,3t_4)$  часов. Смоделировать полеты в данном направлении в течение суток.

Определить среднее число отправленных самолетов, минимальное и максимальное время ожидания полета, время, потраченное на полет вместе с ожиданием в аэропорту, количество пассажиров, отказавшихся от полета.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы аэропорта. Определить оптимальные значения параметров из условия улучшения работы. При реализации проекта использовать средства визуализации.

## ***13. Имитационное моделирование работы станции техобслуживания***

Участок ремонта кузовов автомобилей состоит из двух рабочих мест: первое рабочее место – это кузовной ремонт автомобиля, второе рабочее место – окраска кузова. На каждом рабочем месте 2 человека. После восстановления кузова автомобиля поступают в окрасочную камеру.

Время поступления на ремонт поврежденных автомобилей первой модели – случайная величина, равномерно распределенная на интервале от  $t_1$  до  $t_2$  часов, второй модели – от  $t_3$  до  $t_4$  часов.

На кузовной ремонт автомобилей первой модели тратится от  $t_5$  до  $t_6$  часов, второй модели – от  $t_7$  до  $t_8$  часов.

Время окраски любого автомобиля равномерно распределено на интервале ( $t_9 - t_{10}$ ) минут.

Модели первого типа при обслуживании имеют более высокий приоритет.

В случае, если ремонтная мастерская и покрасочная камера заняты, автомобили дожидаются обслуживания в очередях, длины которых не ограничена.

За 12 часов оценить отдельно для первой и второй модели:

- среднее время, которое тратится на ремонт автомобилей;
- среднее время ожидания в очередях;
- количество отремонтированных автомобилей;
- максимальный размер очереди «ожидания» начала обслуживания и очереди перед операцией окраски.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы станции. Определить оптимальные значения параметров из условия улучшения работы. При реализации проекта использовать средства визуализации.

#### ***14. Имитационное моделирование работы магазина***

В магазин за покупками приходят клиенты. Для работников магазина клиенты классифицируются на постоянных и обычных. Продавцы (менеджеры) затрачивают в среднем  $t_1$  минуты на человека для разъяснения информации по товарам и ответа на вопросы. Приоритетное право на обслуживание без очереди имеют постоянные клиенты;  $p_1$  % посетителей уходят без покупок, а остальные в среднем  $t_2$  времени тратят на выбор товаров, затем встают в очередь в кассу (два кассира), в которой меньше очередь. Кассир в первую очередь он обслуживает постоянных клиентов, а потом обычных посетителей, время обслуживания клиента занимает  $t_3 \pm 0,3t_3$  минут. Кассы закрываются на перерыв по расписанию.

Постоянные клиенты в основном приходят утром (с 9 до 11 часов) и в конце рабочего дня (с 16 до 18 часов), а обычные посетители – в основном в середине дня. Известна средняя стоимость покупки для постоянных клиентов и для обычных.

Смоделировать работу магазина в течение дня. Определить количество обслуженных клиентов (постоянных и обычных), среднюю длину очереди, выручку кассы, количество ушедших без покупок клиентов.

Построить график, отображающий уровень посещаемости магазина покупателями в течение дня и график загруженности кассира.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы магазина. Определить оптимальные значения параметров из условия улучшения работы. При реализации проекта использовать средства визуализации.

### ***15.Имитационное моделирование работы почтового отделения***

Люди приносят на почту письма, которые могут быть двух видов: заказные и обычные. Затем почтовые работники их обрабатывают.

Заказные письма поступают круглосуточно раз в  $t_1-t_2$  минут, а простые письма принимаются только с 8.00 до 20.00 (8.00–12.00 их количество увеличивается от  $n_1$  до  $n_2$ , наибольшее их количество  $n_3$  поступает между 12.00 и 14.00, а затем их количество плавно убывает от  $n_2$  до  $n_1$ ).

Оба вида писем обрабатываются одним работником почтовой службы, причем заказные письма обрабатываются вне очереди, т. к. они важнее. Время обработки заказных писем –  $t_2 \pm 0,3t_2$  минуты, а время обработки простых писем –  $t_3 \pm 0,3t_3$ , чаще  $t_3$  минуты.

Затем все эти письма поступают в отдел подготовки к отправлению второй работник почты), где заказные письма обслуживаются также вне очереди. Время подготовки писем к отправке –  $t_4 \pm 0,3t_4$  минут.

Смоделировать работу почты в течение дня. Определить количество обычных и заказных писем, среднее время работы с письмом, коэффициенты загрузки работников почты, среднее количество необработанной корреспонденции.

Создать анимацию работы сотрудников почты и отразить процесс обработки писем на гистограмме.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы почтового отделения. Определить оптимальные значения параметров из условия улучшения работы. При реализации проекта использовать средства визуализации.

### ***16.Имитационное моделирование работы производственного участка***

На участке термической обработки выполняются цементация и закаливание шестерен, поступающих через  $t_1 \pm 0,3t_1$  мин.

Цементация занимает  $t_2 \pm 0,3t_2$  мин., а закаливание –  $t_3 \pm 0,3t_3$  мин. Качество определяется суммарным временем обработки.

Шестерни со временем обработки:

– больше  $t_5$  мин. – покидают участок;

– от  $t_4$  до  $t_5$  мин. – передаются на повторную закалку;

– меньше  $t_4$  мин. – должны пройти повторную полную обработку.

Детали с суммарным временем обработки меньше  $t_4$  мин. считаются вторым сортом.

Смоделировать процесс обработки в течение рабочего дня.

Определить общее количество обработанных деталей, число повторений закалки и полной обработки, длину очереди. В случае длины очереди больше  $n_1$  предусмотреть автоматическую переналадку станков (уменьшение времени цементации и закалки)

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы производственного участка. Определить оптимальные значения параметров из условия улучшения работы. При реализации проекта использовать средства визуализации.

### ***17. Имитационное моделирование работы автостоянки***

Автостоянка имеет  $n$  мест. Автомобили приезжают и уезжают в течение всего дня с разной интенсивностью (наиболее интенсивно уезжают утром с 7 до 9, наиболее интенсивно приезжают вечером с 18 до 21). В ночное время интенсивность наименьшая. На автостоянке в дневное время работают 2 человека (получение оплаты за стоянку, определение на место стоянки), в ночные часы – 1 человек. Стоимость стоянки в течение суток – «с» рублей. В случае отсутствия мест автомобиль ожидает  $t_1 \pm 0,3t_1$  и уезжает.

Определить выручку в течение дня, график загрузки автостоянки в течение дня, число автомобилей, которые уехали по причине отсутствия мест, среднюю длину очереди.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы автостоянки. Определить оптимальные значения параметров из условия улучшения работы. При реализации проекта использовать средства визуализации.

### ***18. Имитационное моделирование работы взлетно-посадочной полосы аэропорта***

Самолеты прибывают для посадки каждые  $t_1 \pm 0,3t_1$  минут. Если взлетно-посадочная полоса свободна, то самолет получает разрешение на посадку. Если полоса занята, самолет выполняет полет по кругу и возвращается в район аэропорта каждые  $t_2$  мин. Если после пятого круга не получено разрешение на посадку, то самолет отправляется на запасной аэродром.

В аэропорту каждые  $t_3 \pm 0,3t_3$  минут на взлетно-посадочную полосу

выруливают самолеты и получают разрешение на взлет, в случае свободной полосы. Для взлета и посадки самолеты занимают полосу ровно  $t_4$  мин. Приоритет имеет взлетающая машина.

Смоделировать работу полосы в течение суток. Подсчитать количество взлетевших, посаженных, рейсов, отправленных на запасной аэродром. Определить коэффициент загрузки полосы.

Числовые данные (входные параметры) задать самостоятельно. Исследовать влияние изменения входных параметров на показатели качества работы взлетной полосы. Определить оптимальные значения параметров из условия улучшения ее работы. При реализации проекта использовать средства визуализации.

## ЛИТЕРАТУРА

1. Григорьев И. AnyLogic за три дня. Практическое пособие по имитационному моделированию. СПб., 2017.
2. Боев В. Д. Моделирование в AnyLogic. Пособие для практических занятий. СПб.: ВАС, 2016.