# Проект А: Модель обслуживания клиентов в отделении банка.

# Описание предметной области

Клиенты посещают банковское отделение, чтобы воспользоваться банкоматом или услугами банковских клерков. Часть банковских операций клиенты совершают с помощью банкомата, а более сложные операции с помощью сотрудников банка (клерки). В случае, когда банкомат, либо клерки заняты, клиент встаёт в очередь. Однако со временем, в обслуживании были замечены проблемы, например, чрезмерно большие очереди, что повлекло, повышенную напряженность в офисе, продолжительное время нахождение клиентов в офисе, снижение общего числа обслуженных клиентов, конфликтность клиентов и отказы в обслуживании. Требуется построить имитационную модель функционирования офиса для того, чтобы получить ответы на ряд вопросов, например: какова статистика занятости банкомата и средняя длина очереди перед банкоматом; сколько времени клиент проводит в банковском отделении и сколько времени он теряет, ожидая обслуживания; количество отказов; какова средняя пропускная способность офиса за день? За неделю?; произвести оценку затрат операций и определить, сколько денег тратится на обслуживание одного клиента и какую часть этой суммы составляют расходы на оплату работы персонала банка, а какую – на содержание банкомата.

После построения модели необходимо выполнить эксперименты, содержащиеся в индивидуальном варианте. Полученные результаты следует оформить в соответствие с Методические рекомендациями по подготовке отчетов.

Банковское отделение следует рассматривать как систему массового обслуживания (СМО). Предполагается, что время между появлениями клиентов в отделении является случайной величиной, распределённой по закону Пуассона. Имеется очередь к банкомату, ограниченной вместимости. Время обслуживания банкоматом также случайная величина, распределённая по заданному закону. Рекомендуется составить блок-схему процесса, составить список входных и выходных параметров.

#### Задание А.1. Модель обслуживания клиента банкоматом.

**Цель**: Знакомство с интерфейсом системы AnyLogic. Построение простейшей модели процесса обслуживание клиента банкоматом.

- Запустить AnyLogic. Провести диалог «Новая модель». Определиться с названием и местоположением. Выбрать единицу модельного времени.
- 2. Ознакомиться с элементами интерфейса AnyLogic, перечнем и расположением основных окон.
- 3. Построить диаграмму модели, перетаскивая нужные блоки из **Биб- лиотеки моделирования процессов** на холст (графический редактор).
- 4. Дать краткие пояснения к объектам, использованным в модели.
- 5. Установить параметры эксперимента. Проведите эксперимент длительностью в 1 неделю=7 дней=7\*600=4200 мин. Установите модельное время. Освойте окно инспекта.
- 6. Отредактировать титульную страницу презентации эксперимента, с указанием названия проекта и фамилии исполнителя. Разместите на титульной странице произвольное изображение.
- 7. На какие вопросы можно получить ответы с помощью построенной модели? Выявите критичные режимы работы модели. Каким образом можно решить проблему с переполнением очереди?
- 8. Постройте вариант модели, использующей дополнительные порты объекта **queue.**

**Пояснение:** Заявка может покинуть объект queue различными способами:

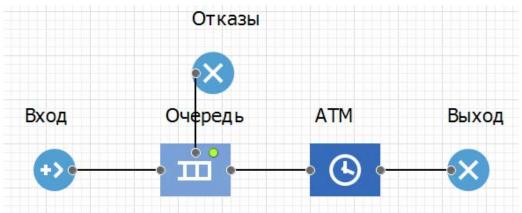
- 1. обычным способом через порт *out*, когда объект, следующий в блоксхеме за этим объектом, готов принять заявку;
- 2. через порт *outTimeout*, если заявка проведет в очереди заданное количество времени (если включен режим таймаута);
- 3. через порт *outPreempted*, будучи вытесненной другой поступившей заявкой при заполненной очереди (если включен режим вытеснения);
- 4. «вручную», путем вызова функции remove() или removeFirst(). Предлагается использовать 3-й способ. Поступающие заявки помещаются в очередь в определённом порядке: либо согласно правилу FIFO (в порядке поступления в очередь), либо согласно приоритетам заявок. Приоритет может быть либо явно храниться в заявке, либо вычисляться согласно свойствам заявки и каким-то внешним условиям. Оч

ередь с приоритетами всегда примет новую входящую заявку, вычислит её приоритет, и поместит в очередь в позицию, соответствующую её приоритету. Если очередь будет заполнена, то приход новой заявки вынудит последнюю хранящуюся в очереди заявку покинуть объект через порт outPreempted. Но если приоритет новой заявки не будет превышать приоритет последней заявки, то тогда вместо неё будет вытеснена именно эта новая заявка.

Пусть все заявки, вырабатываемые объектом source, имеют один и тот же приоритет. Поэтому при полном заполнении накопителя будет теряться по-

следний запрос. Для реализации предложенного способа, выделите объект queue. На панели **Свойства** установите **Вместимость,** например, 15 заявок. Здесь же установите **Разрешить вытеснение**. Для уничтожения потерянных запросов можно добавить второй объект sink1.

9. Установите режим модели, допускающий отказы.



Как подсчитать количество отказов?

- 10.С помощью окна инспекта определите пропускную способность модели.
- 11.Подготовить отчет с выводами, продемонстрировать работу модели, ответить на вопросы преподавателя.

## Задание А.2. Визуализация модели

**Цель**: Освоить возможности AnyLogic создания 3D анимаций.

- 1. Откройте предыдущую модель. Переименуйте объект Delay в ATM (Automated Teller Machines).
- 2. Задайте фигуру банкомата на холсте. Используйте палитру *Размет- ка пространства*, объект **point**.
- 3. Нарисуйте путь, обозначающий очередь к банкомату. Свяжите объекты *queue* и *path* через свойства *queue*, *место агента*. Запустите модель, чтобы увидеть простейшую анимацию и изменение цвета банкомата.
- 4. Добавьте *3D Окно* на холст, из палитры *Презентация*. Запустите модель и освойте сцену 3-х мерной анимации: показать область; переместить сцену; повернуть; масштабировать.
- 5. Освойте добавление 3D объектов. Создать **новый** (нестандартный) тип объекта с помощью **Библиотеку моделирования процессов**, элемент **Тип агента 3**. Диалоговое окно **Мастера создания** агентов, **название типа агента** (Customer) и выбор фигурки человека.

Свяжите объекты *source* и *Customer*. Запустите модель, чтобы увидеть клиентов в очереди.

6. Постройте вариант модели с разными фигурками.

Добавьте на страницу «Customer» параметр из палитры Агент. Назовите параметр «figura».

Откройте свойства параметра и, в значении по умолчанию пропишем строку «uniform discr(0,число объектов – 1)».

Далее добавим объекты (фигурки) на страницу «Customer». Объекты будем брать из палитры «3D модели». Ставим объекты на пересечении координатных осей. Первому объекту в строке *Видимость* нужно прописать figura == 0. Далее, каждому последующему объекту нужно повысить порядковый номер, т.е у следующего объекта в строке *Видимость* будет написано figura == 1 и т.д.

- 7. Добавьте фигуру *Банкомат* из Палитры «*Супермаркет*». Отредактируйте его положение, «лицом к клиентам».
- 8. Подготовить отчет с выводами, продемонстрировать работу модели, ответить на вопросы преподавателя.

# Задание А.З. Добавление клерков

**Цель:** Моделирование клерков с помощью ресурсов, как альтернативы очереди.

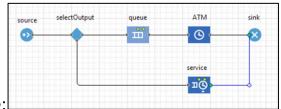
**Уточнение.** Предполагается, что известна вероятность, с которой всякий новый клиент обращается либо к услугам банкомата, либо к услугам клерков.

#### Пояснение.

Ресурс — это специальный объект Библиотеки моделирования процессов, который может потребоваться клиенту (агенту) для выполнения какой-то задачи. В каждый момент времени ресурс может быть занят только одним агентом. В данной модели ресурсы это банковские служащие (клерки), которые предоставляют услуги клиентам банковского отделения.

- 1. Перетащите из **Библиотеки моделирования процессов** на холст Main объект **Service**.
- 2. Задайте параметры объекта *Service* следующим образом:
  - Ко всем клеркам будет вести одна общая очередь. Максимальное количество человек в этой очереди: 20.
  - Время обслуживания имеет треугольное распределение со значениями, согласно варианту в таблице А.
- 3. Перетащите Блок принятия решения **SelectOutput.** В свойствах укажите вероятность выбора услуги, согласно варианту из таблицы.

4. Соедините блоки selectOutput и service с другими блоками так, как



показано на рисунке:

- 5. Перетащите из **Библиотеки моделирования процессов** блок **ResourcePool** на холст агента Main. Объект ResourcePool задает ресурсы определенного типа банковские клерки, разместите объект под блоком **service** и назовите **Клерки**. Количество укажите в соответствие с вариантом из таблицы A.
- 6. В свойствах объекта *service* выберите опцию *Pecypcы одного типа* в параметре *Захватить ресуры, затем* укажите блок *Клерки*.
- 7. Нарисуйте место ожидания клиентами, используя прямоугольный узел. В палитре «Разметка пространства». Назовите «*Место ожидания*» и свяжите с параметром *Место агентов (queue)* объекта service
- 8. Аналогично постройте объект «Место обслуживания», используйте Аттракторы и свяжите объект с параметром Место агентов (delay)объекта service
- 9. Аналогично постройте объект **«Места клерков»**, используйте **Ат- тракторы**, поверните лицом к клиентам. и свяжите объект с параметром **Базовое расположение (узлы)** объекта **Клерки**
- 10. Добавьте 3D фигуры клерки с помощью элемента *Тип ресурса* 🙃
- 11. Добавьте столы для клерков. Откройте палитру 3D Объекты. Перетащите четыре 3D фигуры Стол из секции палитры Офис в графический редактор (холст) и поместите их в узел *Места клерков*. Расположите столы на аттракторах. При необходимости поверните и подровняйте.
- 12.Подготовить отчет с выводами, продемонстрировать работу модели, ответить на вопросы преподавателя.

### Задание А.4. Добавление статистики

**Цель:** Освоить средства для сбора статистики по работе блоков диаграммы процесса.

Определим статистику занятости банкомата и длины очереди перед банкоматом. Вычислим, сколько времени клиент проводит в банковском отделении и сколько времени он теряет, ожидая своей очереди

- 1. Создайте на холсте диаграмму средней занятости банкомата. Перетащите элемент **Столбиковая диаграмма** из палитры **Статистика** на холст.
- 2. В секции Данные свойств столбиковой диаграммы, щелкните кнопку Добавить элемент данных, измените Заголовок на ATM utilization. В поле Значение введите ATM.statsUtilization.mean().Здесь ATM имя объекта Delay. У каждого объекта Delay есть встроенный набор данных statsUtilization, занимающийся сбором статистики использования этого объекта. Функция mean() возвращает среднее из всех измеренных этим набором данных значений. Вы можете использовать и другие методы сбора статистики, такие, как min() или max(). Статистика использования это числовой коэффициент, равный отношению времени занятости объекта обслуживанием заявки к общему времени работы.
- 3. Отформатируйте вертикально, легенду расположите справа.
- 4. Аналогично постройте столбиковую диаграмму для средней длины очереди. Заголовок: Queue length и задайте Значение: queue.statsSize.mean(). Отформатируйте горизонтально, легенду расположите сверху.
- 6. Добавьте элементы сбора статистики по времени ожидания клиентов waitTimeDistr и времени пребывания клиентов в системе timeInSystemDistr. Эти элементы будут запоминать соответствующие значения времен для каждого клиента и предоставят пользователю стандартную статистическую информацию: среднее, минимальное, максимальное из измеренных значений, среднеквадратичное отклонение, доверительный интервал для среднего и т.д.). Эти объекты находятся в библиотеки Статистика и их следует добавить на холст main.
- 7. Откройте свойства объекта source. Введите agent.enteredSystem = time(); в поле действия При выходе в секции Действия.

- 8. Измените свойства объекта queue: Введите agent.startWaiting = time(); в поле действия При входе в секции Действия. Введите waitTimeDistr.add(time() agent.startWaiting); в поле действия При выходе. Этот код добавляет время, в течение которого клиент ожидал обслуживания, в объект сбора данных waitTimeDistr.
- 9. Измените свойства объекта sink: Введите timeInSystemDistr.add(time()-agent.enteredSystem); в поле действия При входе в секции Действия. Этот код добавляет полное время пребывания клиента в банковском отделении в объект сбора данных гистограммы timeInSystemDistr.
- 10.Добавьте в модель две гистограммы для отображения распределений времен ожидания клиента и пребывания клиента в системе.
- 11. Проведите эксперименты длительностью неделя. Проведите анализ для разных значений параметра.
- 12.Оцените затраты, если клерку платят 6,5 долларов в час, когда он занят обслуживанием клиентов и 4 доллара в час иначе. Одна операция банкомата обходится в 0,3 доллара.
- 13.Подготовить отчет с выводами, продемонстрировать работу модели, ответить на вопросы преподавателя.

# Указания к проекту А.

- 1. Модель должна разрешать отказы. Отказом считается выход заявки через порт вытеснения.
- 2. Для изменения цвета банкомата можно использовать скрипт: ATM.size() > 0 ? red : green
- 3. Скрипт для счётчика отказов: agent.col\_otkaz=sink1.count() количество заявок, вошедших в sink1
- 4. Отчет должен быть подготовлен в соответствии с документом *Meтодические рекомендации по подготовке отчета* (cm.eluniver.ugrasu.ru)