МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное автономное бюджетное образовательное учреждение высшего образования

Севастопольский государственный университет

кафедра Информационных систем

**Лисянский Александр Игоревич**

Институт информационных технологий и управления в технических системах

курс 2 группа ИC/м-21(о)

09.04.02 Информационные системы и технологии

Лабораторная работа №8

по дисциплине «МЕТОДЫ ИССЛЕДОВАНИЯ И МОДЕЛИРОВАНИЯ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ И ТЕХНОЛОГИЙ»

«СРЕДА ИМИТАЦИОННОГО МОДЕЛИРОВАНИЯ ANYLOGIC. МОДЕЛЬ ПРЕДОСТАВЛЕНИЯ РЕМОНТНЫХ УСЛУГ»

Отметка о зачёте \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_

(дата)

Руководитель практикума

\_\_\_\_\_проф.\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_Ю.В. Доронина\_\_\_\_\_\_\_\_\_\_

(должность) (подпись) (инициалы, фамилия)

Севастополь 2017

1. **Цель работы**

Закрепления навыков работы со средой имитационного моделирования Anylogic.

1. **Постановка задачи**

В фирму предоставления ремонтных услуг поступают заявки n типов с вероятностями р1, р2, ..., рn соответственно. Интервалы времени Тп между двумя очередными поступлениями одного типа заявок случайные. Каждый любой тип заявки может требовать одного из а1, а2, ..., аk видов ремонта с вероятностями рa1, рa2, ..., рak соответственно.

В фирме имеются n1, n2, ..., nn мастеров для выполнения заявок каждого типа соответственно. Мастера n1 выполняют заявки первого типа. Если их нет и мастера n2, ..., nn групп заняты, они выполняют заявки этих типов. При этом поступающие заявки первого типа ожидают их освобождения. Мастера n2 выполняют заявки второго типа. Если их нет и мастера n3, n4, ..., nn групп заняты, они выполняют заявки этих типов. При этом поступающие заявки второго типа ожидают их освобождения. Аналогичные обязанности и у мастеров остальных групп. Только мастера nn выполняют заявки одного n-го типа.

Время выполнения заявки n-го типа случайное, не зависит от мастера, а зависит только от вида ремонта: Т11, Т12, Т13 – для СС первого типа, Т21, Т22, Т23 – для СС второго типа, ..., Tn1, Tn2, ..., Tnn – для СС n-го типа.

Прием и распределение заявок между группами мастеров осуществляется d диспетчерами. Время, затрачиваемое одним диспетчером на одну заявку, Т1, случайное. Диспетчерами не принимаются к ремонту q заявок всех типов.

Тп = 30; n = 4;

p1 = 0.2, p2 = 0.3, p3 = 0.25, p4 = 0.25;

p11 = 0.5, p12 = 0.25, p13 = 0.25;

n1 = 2; T11 = 30; T12 = 40; T13 = 50;

n2 = 1; T21 = 20; T22 = 30; T23 = 40;

n3 = 1; T31 = 15; T32 = 25; T33 = 35;

n4 = 1; T41 = 25; T42 = 35; T43 = 45;

d = 2; T1 = 15; q = 2 %.

Интервалы времени распределены по экспоненциальному закону.

Разработать имитационную модель предоставления ремонтных услуг. Исследовать зависимость количества выполненных заявок и вероятностей выполнения заявок всех типов от интервала Тп поступления их в ремонт и вероятностей p1, p2, p3, p4.

1. **Разработка модели**

В ходе выполнения лабораторной работы была разработана модель AnyLogic. Графическое изображение модели представлено на рисунке 1.

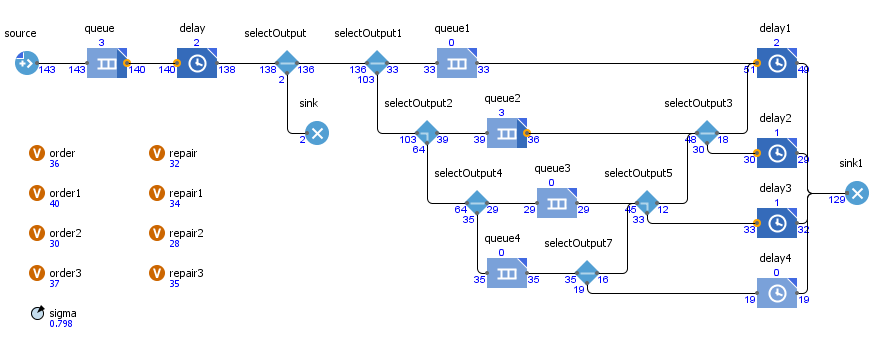


Рисунок 1 – Графическое изображение модели

Заявки вводятся в систему с помощью объекта источника. При выходе из блока увеличивается значение переменной количество заявок соответствующего типа. Параметры источника заявок представлены на рисунке 2.

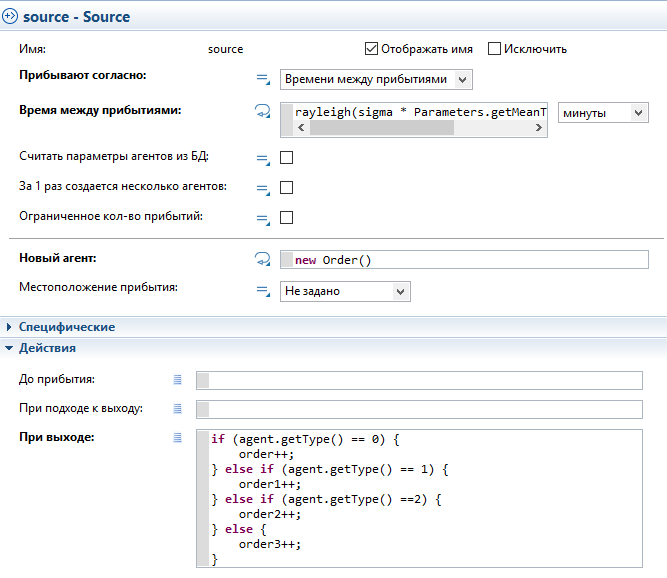


Рисунок 2 – Основные параметры источника заявок

Далее заявка попадает к диспетчерам. Подача заявки через диспетчера моделируется тремя блоками – очередью к диспетчерам, задержки (принятия заказа) и блока выбора определяющего, моделирующего принятие или отказ. В случае отказа заявка выводится из системы. Параметры этих трёх блоков представлены на рисунках 3-5.

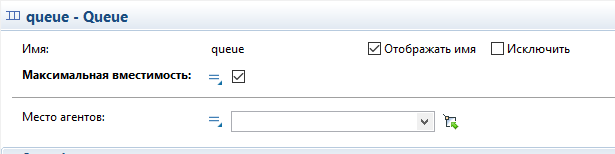


Рисунок 3 – Параметры очереди к диспетчерам

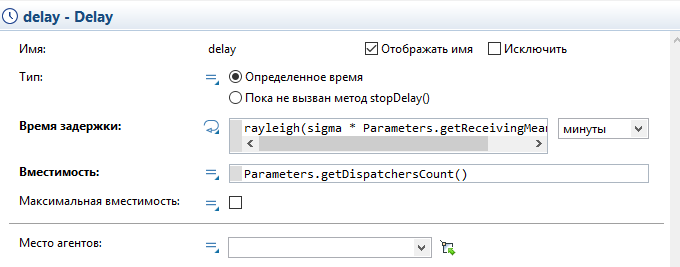


Рисунок 4 – Параметры блока задержки, моделирующего приём устройства

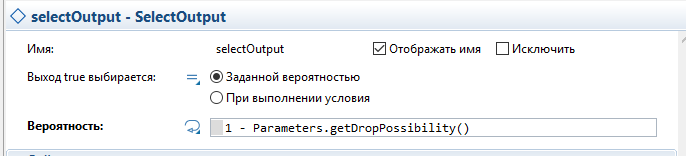


Рисунок 5 – Параметры блока выбора между приёмом или отказом от приёма заявки

Далее заявки при помощи блоков выбора сортируются по типам и направляются в соответствующие очереди. Параметры одного из блоков выбора для сортировки заявок представлены на рисунке 7.

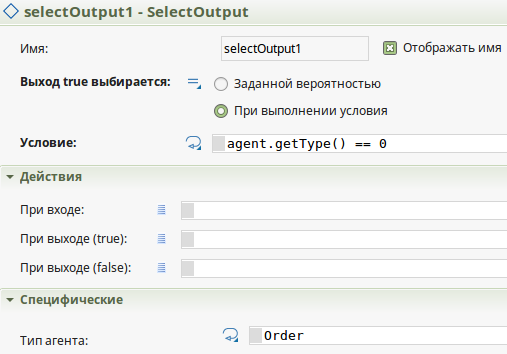


Рисунок 7 – . Параметры одного из блоков выбора для сортировки заявок

После того, как заявка вынимается из очереди, она направляется на блок выбора, проверяющий все ли мастера заданного типа заняты и свободны ли мастера и очереди заявок для тех типов, мастера которых могут обслужить данную заявку. Пример параметров блока выбора для заявки последнего типа представлен на рисунке 8.

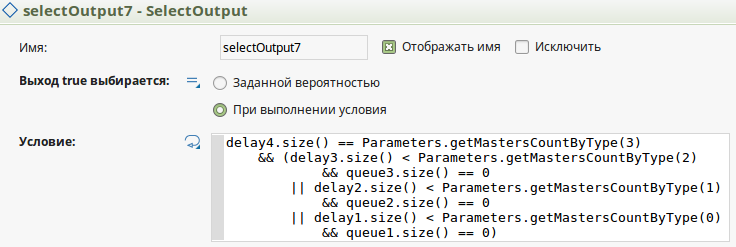


Рисунок 8 – Пример параметров блока выбора для заявки последнего типа

Далее заявка направляется к мастерам определённого типа и выполняется их обслуживание, моделируемое блоком задержки. Его параметры представлены на рисунке 9.

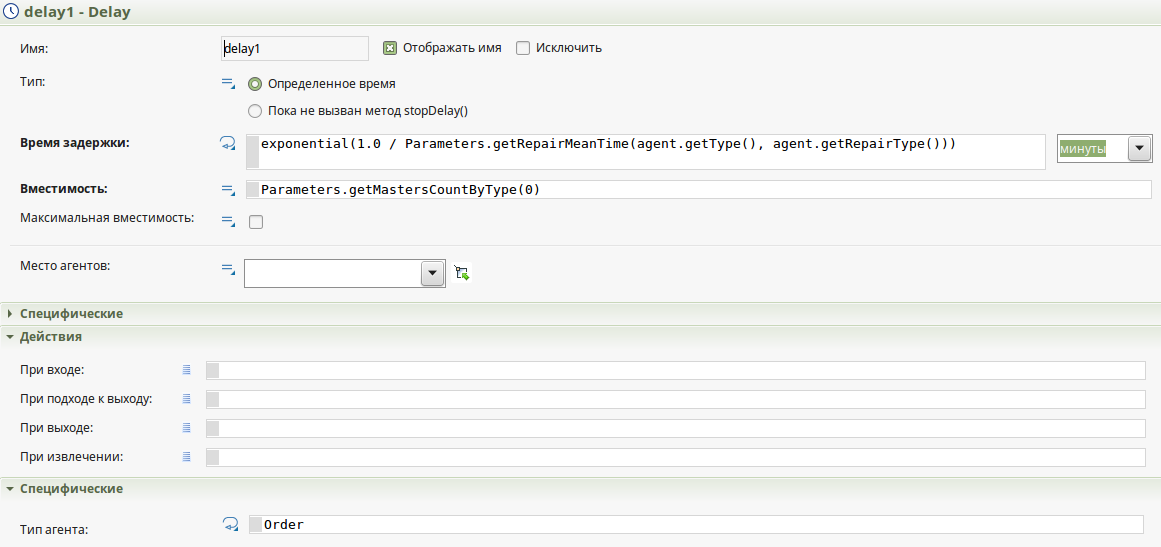


Рисунок 9 – Параметры блока задержки, моделирующего обслуживание

После обслуживания заявки выводятся из системы. В момент попадения в блок вывода увеличивается значение соответствующей переменной количества обслуженных заявок. Параметры блока представлены на рисунке 10.



Рисунок 10 – Параметры блока вывода заявок из системы.

Также разработаны два класса – Order, представляющий заявку, и Parameters, содержащий параметры модели.

Текст класса Order.java:

public class Order extends Agent {

private int type;

private int repairType;

public Order() {

Random random = new Random();

double typeDefinition = random.nextDouble();

for (int i = 0; typeDefinition > 0; i++) {

if (typeDefinition < Parameters.getPossibilityByType(i)) {

type = i;

}

typeDefinition -= Parameters.getPossibilityByType(i);

}

double repairTypeDefinition = random.nextDouble();

for (int i = 0; repairTypeDefinition > 0; i++) {

if (repairTypeDefinition < Parameters.getPossibilityByRepairType(i)) {

repairType = i;

}

repairTypeDefinition -= Parameters.getPossibilityByRepairType(i);

}

}

public int getType() {

return type;

}

public int getRepairType() {

return repairType;

}

}

Текст класса Parameters.java:

public class Parameters implements Serializable {

private static double[] TYPE\_POSSIBILITY = {0.3, 0.2, 0.25, 0.25};

private static double[] REPAIR\_TYPE\_POSSIBILITY = {0.5, 0.25, 0.25};

private static int[] MASTERS\_COUNT = {2, 1, 1, 1};

private static double[][] REPAIR\_TYPE\_BY\_TYPE\_MEAN\_TIME = {

{30, 40, 50},

{20, 30, 40},

{15, 25, 35},

{25, 35, 45},

};

private static int DISPATCHERS\_COUNT = 2;

private static double RECEIVING\_MEAN\_TIME = 15;

private static double DROP\_POSSIBILITY = 0.02;

private static double MEAN\_TIME\_BETWEEN\_ORDERS = 10;

public static double getPossibilityByType(int typeNumber) {

return TYPE\_POSSIBILITY[typeNumber];

}

public static double getPossibilityByRepairType(int repairTypeNumber) {

return REPAIR\_TYPE\_POSSIBILITY[repairTypeNumber];

}

public static int getMastersCountByType(int typeNumber) {

return MASTERS\_COUNT[typeNumber];

}

public static double getRepairMeanTime(int repairTypeNumber, int typeNumber) {

return REPAIR\_TYPE\_BY\_TYPE\_MEAN\_TIME[repairTypeNumber][typeNumber];

}

public static int getDispatchersCount() {

return DISPATCHERS\_COUNT;

}

public static double getReceivingMeanTime() {

return RECEIVING\_MEAN\_TIME;

}

public static double getDropPossibility() {

return DROP\_POSSIBILITY;

}

public static double getMeanTimeBetweenOrders() {

return MEAN\_TIME\_BETWEEN\_ORDERS;

}

private Parameters() { }

}

1. **Проведение эксперимента**

В ходе первого эксперимента было исследовано влияние среднего времени между поступлениями заявок на вероятность их обработки. Результаты эксперимента представлены в таблице 1.

Таблица 1 – Результаты исследования влияния среднего времени между поступлениями заявок на вероятность их обработки

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Среднее время между поступлением заявок | Количество поступивших заявок | Количество обработанных заявок | Вероятность обработки заявки | Количество поступивших заявок 1-го типа | Количество обработанных заявок 1-го типа | Вероятность обработки заявки 1-ого типа | Количество поступивших заявок 2-го типа | Количество обработанных заявок 2-го типа | Вероятность обработки заявки 2-ого типа | Количество поступивших заявок 3-го типа | Количество обработанных заявок 3-го типа | Вероятность обработки заявки 3-ого типа | Количество поступивших заявок 4-го типа | Количество обработанных заявок 4-го типа | Вероятность обработки заявки 4-ого типа |
| 1 | 1465 | 170 | 0,13 | 295 | 34 | 0,12 | 440 | 53 | 0,12 | 332 | 41 | 0,12 | 398 | 42 | 0,10 |
| 2 | 697 | 178 | 0,25 | 148 | 45 | 0,30 | 207 | 54 | 0,26 | 167 | 34 | 0,20 | 175 | 45 | 0,26 |
| 4 | 359 | 168 | 0,47 | 78 | 37 | 0,47 | 113 | 58 | 0,51 | 90 | 32 | 0,35 | 78 | 41 | 0,53 |
| 6 | 233 | 179 | 0,77 | 43 | 31 | 0,72 | 58 | 45 | 0,78 | 68 | 53 | 0,78 | 64 | 50 | 0,78 |
| 8 | 191 | 174 | 0,91 | 33 | 31 | 0,94 | 59 | 50 | 0,85 | 49 | 45 | 0,92 | 50 | 48 | 0,96 |
| 10 | 142 | 137 | 0,96 | 23 | 26 | 1,00 | 42 | 42 | 1,00 | 40 | 36 | 0,90 | 34 | 33 | 0,97 |

Как видно из результатов, система рассчитана на среднее время между поступлением заявок не меньше 9 минут. В случае уменьшения начинает происходить накопление заявок, не успевающих обработкаться. из-за чего уменьшается общая вероятность обработки.

В ходе второго эксперимента было исследовано влияние комбинаций вероятностей поступления заявок определённых типов на вероятность их обработки. Результаты эксперимента представлены в таблице 2.

Таблица 2 – Результаты исследования влияния комбинаций вероятностей поступления заявок определённых типов на вероятность их обработки

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Вероятность поступления заявки 1-го типа | Вероятность поступления заявки 2-го типа | Вероятность поступления заявки 3-го типа | Вероятность поступления заявки 4-го типа | Количество поступивших заявок 1-го типа | Количество обработанных заявок 1-го типа | Количество поступивших заявок 2-го типа | Количество обработанных заявок 2-го типа | Количество поступивших заявок 3-го типа | Количество обработанных заявок 3-го типа | Количество поступивших заявок 4-го типа | Количество обработанных заявок 4-го типа |
| 0,25 | 0,25 | 0,25 | 0,25 | 32 | 31 | 41 | 39 | 45 | 44 | 31 | 29 |
| 0,3 | 0,2 | 0,25 | 0,25 | 37 | 31 | 30 | 30 | 40 | 39 | 41 | 38 |
| 0,1 | 0,2 | 0,3 | 0,4 | 20 | 19 | 33 | 31 | 35 | 35 | 61 | 58 |
| 0,4 | 0,3 | 0,2 | 0,1 | 60 | 56 | 39 | 36 | 26 | 26 | 17 | 16 |
| 0,1 | 0,3 | 0,2 | 0,4 | 11 | 10 | 38 | 37 | 23 | 22 | 71 | 67 |
| 0,2 | 0,4 | 0,3 | 0,1 | 33 | 29 | 51 | 51 | 35 | 32 | 22 | 21 |
| 0,3 | 0,2 | 0,4 | 0,1 | 34 | 33 | 33 | 31 | 57 | 55 | 19 | 17 |

По результатам эксперимента сделан вывод, что вероятность обслуживания заявки тем больше, чем больше типов мастеров могут её выполнить. Кроме того установлено, что заявка, имеющая большую вероятность появления как правило имеет меньшую вероятность обслуживания.

**Вывод:**

В ходе выполнения лабораторных работ закреплены навыки работы со средой имитационного моделирования Anylogic. Построена модель предоставления ремонтных услуг. Исследовано влияние среднего времени между поступлениями заявок на вероятность их обработки. Необходимое среднее время между поступлением заявок не должно быть меньше 9 минут. Кроме того исследовано влияние комбинаций вероятностей поступления заявок определённых типов на вероятность их обработки. Вероятность обслуживания заявки тем больше, чем больше типов мастеров могут её выполнить. Заявка, имеющая большую вероятность появления как правило имеет меньшую вероятность обслуживания.