Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное

учреждение высшего образования

«Севастопольский государственный университет»

Институт информационных технологий и управления в технических системах

Кафедра ИС

ОТЧЁТ

По лабораторной работе №4

«Исследование подходов к моделированию вычислительных систем»

Выполнил:

Ст. гр. ИС/б-20-1-о

Хроменко Д.А.

Проверил:

Хохлов В.В.

Севастополь

2022

**4.1 Цель работы**

Исследование технологий моделирования вычислительных систем на примере имитационного моделирования алгоритмов.

**4.2 Ход работы**

Была создана аналитическая модель процесса сортировки пузырьком в форме сети Петри. Была создана имитационная модель процесса сортировки пузырьком в среде AnyLogic на основе сети Петри. Была создана имитационная модель процесса.

Создана имитационная модель заданного по варианту процесса в среде Anylogic с использованием диаграммы действий на основе схемы алгоритма. Также, была организована визуализация работы алгоритма. Полученная имитационная модель представлена на рисунке 1.

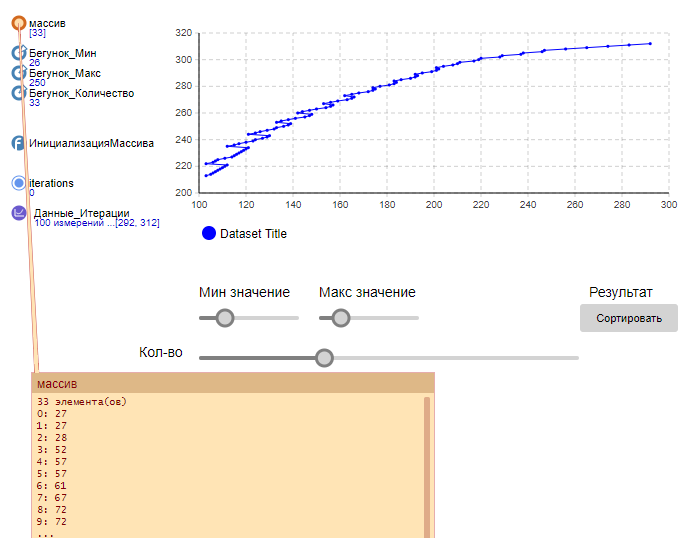


Рисунок 1 – Имитационная модель с помощью диаграммы действий

**Выводы**

В ходе выполнения данной лабораторной работы были исследованы характеристики одноканальной системы массового обслуживания, использовав аналитический и имитационный методы моделирования; изучены особенности работы, отладки с помощью пакета моделирования Anylogic.

**Ответы на контрольные вопросы**

1. Основные достоинства:

• имитационная модель позволяет описать моделируемый процесс с большей наглядностью, чем другие;

• имитационная модель обладает гибкостью варьирования структуры, алгоритмов и параметров системы;

• применение ЭВМ существенно сокращает продолжительность испытаний по сравнению с натурным экспериментом (если он возможен), а также их стоимость.

Основные недостатки:

• решение, полученное на имитационной модели, всегда носит частный характер, так как оно соответствует фиксированным элементам структуры, алгоритмам поведения и значениям параметров системы;

• большие трудозатраты на создание модели и проведение экспериментов, а также обработку их результатов;

• если использование системы предполагает участие людей при проведении машинного эксперимента, на результаты может оказать влияние так называемый хоторнский эффект (заключающийся в том, что люди, зная (чувствуя), что за ними наблюдают, могут изменить свое обычное поведение).

2. Классические сети Петри ввел Карл Адам Петри в 60-х гг. XX в. С тех пор их использовали для моделирования и анализа самых разных систем с приложениями от протоколов, аппаратных средств и внедренных систем до гибких производственных систем, пользовательского взаимодействия и бизнес-процессов. Сети Петри предназначены для моделирования систем, которые состоят из множества взаимодействующих друг с другом компонент. При этом компонента сама может быть системой. Примерами таких систем могут служить вычислительные системы, в том числе и параллельные, компьютерные сети, программные системы, обеспечивающие их функционирование, а также экономические системы, системы управления дорожным движением, химические системы, и т. д.

3. Наиболее наглядным представлением сети Петри является ее графическое изображение в виде (рисунок 1) двудольного, ориентированного мультиграфа. Двудольный граф – это такой граф, множество вершин которого разбивается на два подмножества и не существует дуги, соединяющей две вершины из одного подмножества. Мультиграф – граф, в котором может существовать пара вершин, которая соединена более чем двумя дугами.

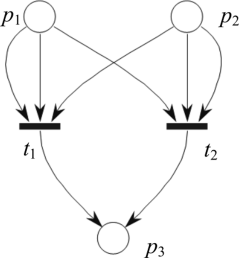


Рисунок 1 — Графическое отображение сети Петри

Граф сети Петри обладает двумя типами узлов: кружок, представляющий позицию сети Петри, и планка, представляющая переход сети Петри.

Ориентированные дуги этого графа (стрелки) соединяют переход с его входными и выходными позициями. При этом дуги направлены от входных позиций к переходу и от перехода к выходным позициям. Кратным входным и выходным позициям перехода соответствуют кратные входные и выходные дуги.

4. Сеть Петри формально определяется как четверка вида:

S = <P, T, E, M0>,

где Р – конечное множество позиций (иначе—мест или условий);

T – конечное множество переходов (событий);

Е – конечное множество дуг, E Í (P ´ T) È (Т ´ Р);

M0 – начальная маркировка (разметка), M0: P ® N, N = {0, 1, 2, ...}- множество натуральных чисел.

5.

Таблица 1 – Элементы сетей Петри

|  |  |
| --- | --- |
| Название элемента | Изображение элемента |
| Позиция (P) |  |
| Переход (T) |  |
| Дуга |  |

Переходы в сети Петри являются событиями, которые изменяют состояния в реальной системе. Каждому условию в сети Петри соответствует определенная позиция. Совершению события соответствует срабатывание (возбуждение или запуск) перехода, при котором маркеры из входных позиций этого перехода перемещаются в выходные позиции. Последовательность событий образует моделируемый процесс. Перемещаемые по сети маркеры часто называют фишками.

6. Сеть Петри может быть смоделирована в AnyLogic при помощи использования семантически соответствующих ей элементов из библиотеки моделирования процессов. Позиции и переходы в таком случае, будут реализованы, как при помощи элементов агентов.

7. Диаграммы действий используются для визуального задания функций. Каждый элемент диаграммы действий задаёт её основные свойства, а именно тип возвращаемого значения, аргументы, уровень доступа и т.д.

8. Запуск визуализации работы диаграммы действий осуществляется таким же образом, что и для остальных видов моделей, а именно при помощи нажатия на кнопку запуска (зелёный круг с белым треугольником) в верхней панели инструментов или же при помощи горячей клавиши F5.

9. Модель, разработанная с помощью диаграммы действий в рамках четвёртой лабораторной работы, использует класс коллекции LinkedList для хранения элементов списка, в котором будет производится алгоритм бинарного поиска.

10. События в диаграмме действий реализуются при помощи блоков Код, которые приводят к изменению состояний системы. Каждому условию в диаграмме действий соответствует блок ветвления или определённый цикл. Совершению события соответствует срабатывание блока Код, при котором маркеры из входных позиций этого блока перемещаются в выходные позиции. Последовательность событий образует моделируемый процесс.