

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Космических и информационных технологий

институт

Вычислительная техника

кафедра

УТВЕРЖДАЮ

Заведующий кафедрой

О.В. Непомнящий

инициалы, фамилия

«__» _____ 2018 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.01 Информатика и вычислительная техника

код и наименование направления

Программный комплекс для автоматизации проверки выполнения
практической работы «Языки описания конечных автоматов» по дисциплине
«Прикладная теория цифровых автоматов»

тема

Руководитель

подпись, дата

доцент, канд.техн.наук

должность, ученая степень

А.И. Постников

инициалы, фамилия

Выпускник

подпись, дата

М. А. Совков

инициалы, фамилия

Нормоконтролер

подпись, дата

доцент, канд.техн.наук

должность, ученая степень

В.И. Иванов

инициалы, фамилия

Красноярск 2018

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Космических и информационных технологий

институт

Вычислительная техника

кафедра

УТВЕРЖДАЮ

Заведующий кафедрой

О.В. Непомнящий

инициалы, фамилия

«__» _____ 2018 г.

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
в форме _____ бакалаврской работы

бакалаврской работы, дипломного проекта, дипломной работы, магистерской диссертации

Студенту _____ Совкову Максиму Андреевичу
фамилия, имя, отчество

Группа КИ14-07Б Направление (специальность) 09.03.01
номер код

«Информатика и вычислительная техника»

наименование

Тема выпускной квалификационной работы Программный комплекс для автоматизации проверки выполнения практической работы «Языки описания конечных автоматов» по дисциплине «Прикладная теория цифровых автоматов».

Утверждена приказом по университету № _____ от _____

Руководитель ВКР А.И. Постников, к.т.н., доцент, ИКИТ СФУ
инициалы, фамилия, должность, учебное звание и место работы

Исходные данные для ВКР Задание на выпускную квалификационную работу

Перечень разделов для ВКР Задание на выпускную квалификационную работу, анализ технического задания на выпускную квалификационную работу, структура программного комплекса, формирование отчета, руководство пользователя

Перечень графического материала: Презентация доклада выступления

Руководитель ВКР _____ А.И. Постников
подпись инициалы и фамилия

Задание принял к исполнению _____ М.А. Совков
подпись инициалы и фамилия

« _____ » _____ 2018 г.

РЕФЕРАТ

Выпускная квалификационная работа по теме «Программный комплекс для автоматизации проверки выполнения практической работы «Языки описания конечных автоматов» по дисциплине «Прикладная теория цифровых автоматов». Пояснительная записка содержит 46 страниц текстового документа, 27 иллюстрации, 5 использованных источников.

ПРОГРАММНЫЙ КОМПЛЕКС, ЯЗЫКИ ОПИСАНИЯ КОНЕЧНЫХ АВТОМАТОВ, АВТОМАТ МИЛИ, АВТОМАТ МУРА.

Объектом исследования данной работы является раздел прикладной теории цифровых автоматов (ПТЦА), изучающий языки описания конечных автоматов.

Цель работы – разработать программный комплекс, позволяющий с помощью средств вычислительной техники заполнять базу данных заданиями и ответами на практическую работу «Языки описания конечных автоматов» по дисциплине «Прикладная теория цифровых автоматов» и автоматизировать составление отчёта по практической работе.

В процессе работы был произведен анализ теоретических сведений по заданному вопросу, на основе которых была определена структура программного комплекса.

Результатом работы является реализация указанного выше программного комплекса с подробным описанием структуры и алгоритмов работы.

СОДЕРЖАНИЕ

Введение.....	3
1 Задание на ВКР	4
2 Анализ задания	6
2.1 Анализ предметной области.....	6
2.2 Основные понятия теории автоматов.....	7
2.3 Языки описания конечных автоматов.....	9
2.3.1 Матричное представление автомата.....	9
2.3.2 Графовое представление автомата.....	9
2.3.3 Теоретико-множественное представление автомата	10
2.3.4 Табличное представление автомата.....	11
2.4 Выбор программных средств.....	12
2.4.1 Выбор языка программирования.....	12
2.4.2 Выбор среды разработки	13
3 Структура программного комплекса.....	13
3.1 Структура программы для формирования базы вариантов.....	14
3.2 Структура программы для выполнения практической работы.....	18
4 Формирование отчёта.....	22
5 Руководство пользователя.....	26
5.1 Программа для формирования базы вариантов.....	26
5.2 Программа для выполнения практической работы.....	34
Заключение.....	44
Список сокращений.....	45
Список использованных источников.....	46

ВВЕДЕНИЕ

Технологии в современном мире стремительно развиваются, позволяя автоматизировать различные процессы. Например, автоматизация проверки выполнения практических работ у студентов даст возможность значительно сэкономить время преподавателя и позволит избежать ошибок, допускаемых во время ручной проверки.

Целью данной ВКР является разработка программного комплекса, позволяющего проводить выполнение и проверку практической работы «Язык описания конечных автоматов» и автоматизировать составление отчёта по практической работе.

1 Задание на ВКР

Заданием на выпускную квалификационную работу является разработка программного комплекса, позволяющего проводить выполнение практической работы «Языки описания конечных автоматов» по дисциплине ПТЦА на компьютере, и автоматизация составления отчёта по практической работе.

Состав разрабатываемого программного комплекса:

- программа для формирования базы вариантов;
- программа для выполнения практической работы.

Исходные данные:

- число состояний автомата – от 6 до 8;
- число входных сигналов автомата – от 3 до 6;
- число выходных сигналов автомата – от 3 до 6;
- число вариантов заданий – 60;
- законы функционирования автоматов Мили и Мура.

Программа для формирования базы вариантов должна обеспечивать:

- возможность ввода типа автомата, числа состояний автомата, числа входных и выходных сигналов автомата;
- возможность выбора условия задания (какое именно представление будет отображено в программе для выполнения практической работы);
- возможность ввода матричного представления автоматов Мили и Мура;
- возможность ввода табличного представления автоматов Мили и Мура;
- возможность ввода теоретико-множественного представления автоматов Мили и Мура;
- возможность загрузки изображения графового представления автоматов Мили и Мура;
- шифрование и сохранение введённых данных в базе вариантов;
- возможность редактирования ранее введённых и сохранённых данных в базе вариантов.

Программа для выполнения практической работы должна обеспечивать:

- возможность выбора варианта задания и отображение текста задания для выбранного варианта;
- возможность составления матричного, табличного и теоретико-множественного представлений автомата на основе графового представления автомата;
- возможность составления матричного и табличного представлений автомата на основе теоретико-множественного представления автомата;
- возможность составления теоретико-множественного и табличного представлений автомата на основе матричного представления автомата;
- возможность составления теоретико-множественного и матричного представлений автомата на основе табличного представления автомата;
- возможность запуска стороннего приложения для создания графового представления;
- загрузка данных из базы вариантов, дешифрование и проверка составленных представлений с теми, что сохранены в базе;
- сохранение составленных представлений автомата в файл;
- загрузка ранее сохранённых представлений автомата из файла и вывод в рабочую область для дальнейшей работы с ними;
- возможность ввода студентом данных для формирования отчета (номер практической работы, ФИО преподавателя, ФИО студента, группа, номер зачётной книжки);
- включение окончательных результатов выполнения практической работы в формируемый отчет;
- возможность сохранения отчёта в файл формата «Microsoft Office Word (*.docx)».

Системные требования:

- операционная система Microsoft Windows 7 и выше;

– программный продукт Microsoft Office 2007 и выше (для формирования отчета в формате «*.docx»).

2 Анализ задания

Разработка программного комплекса, позволяющего проводить выполнение практической работы по дисциплине ПТЦА по теме «Языки описания конечных автоматов», сводится к анализу предметной области, пониманию основных понятий теории автоматов, пониманию языков описания конечных автоматов и выбору программных средств для разработки.

2.1 Анализ предметной области

Прикладная теория цифровых автоматов – это раздел теории автоматов, ориентированный на разработку структурных и функциональных схем конечных автоматов, которые должны обладать нужными функциями, т.е. выполнять ряд действий, предписанных законами функционирования автомата.

Предметом теории автоматов является изучение математических моделей преобразователей дискретной информации. В данной теории решаются следующие основные задачи: анализ и синтез автоматов, определение полноты, минимизация и эквивалентные преобразования автоматов [1].

Наиболее тесно теория автоматов связана с теорией алгоритмов. Это объясняется тем, что автомат преобразует дискретную информацию по шагам в дискретные моменты времени и формирует результирующую информацию по шагам заданного алгоритма. Эти преобразования возможны с помощью технических и/или программных средств. Автомат можно представить, как некоторое устройство, на которое подаются входные сигналы и снимаются выходные и которое может иметь некоторые внутренние состояния. При анализе

автоматов изучают их поведение при различных возмущающих воздействиях и минимизируют число состояний автомата для работы по заданному алгоритму.

Целью выполнения практической работы по дисциплине ПТЦА по теме «Языки описания конечных автоматов» является получение навыков в описании конечных автоматов с помощью различных представлений.

2.2 Основные понятия теории автоматов

Термин «автомат» используется в двух аспектах. С одной стороны, автомат – устройство, выполняющее некоторые функции без участия человека (например, ЭВМ). С другой стороны, термин «автомат» – математическое понятие, обозначающее математическую модель реальных технических процессов.

В общем случае автомат представляется как «чёрный ящик» и полностью описывается совокупностью следующих шести объектов:

- множество входных сигналов X ;
- множество выходных сигналов Y ;
- множество состояний автомата A ;
- начальное состояние автомата a_0 как элемент множества A ;
- функция перехода из одного состояния в другое $f(a,x)$;
- функция выходов автомата $\varphi(a,x)$.

В начальный момент времени t_0 автомат находится в состоянии a_0 . В каждый момент времени t , определяемый интервалом дискретности, автомат под воздействием входного сигнала $x(t)$ скачкообразно переходит из состояния $a_i(t)$ в состояние $a_i(t+1)$ и выдает соответствующий выходной сигнал $y(t)$.

Понятие состояние автомата используется для описания систем, выходы которых зависят не только от входных сигналов в данный момент времени, но и от некоторой предыстории, – сигналов, которые поступили на входы системы ранее, т. е. состояние – некоторая память о прошлом.

Цифровые автоматы делят на два класса. В синхронных автоматах моменты времени, в которые фиксируются состояния автомата, задаются специальным устройством – генератором синхроимпульсов. В асинхронных автоматах моменты перехода автомата из одного состояния в другое заранее не определены и зависят от каких-то событий.

В зависимости от закона определения выходных сигналов синхронные автоматы делятся на автоматы Мили и автоматы Мура.

Для автомата Мили закон функционирования определяется следующими соотношениями:

$$\begin{cases} a(t+1) = f(a(t), x(t)), \\ y(t) = \varphi(a(t), x(t)). \end{cases}$$

Следующее состояние автомата Мили (состояние в момент времени $t+1$) зависит от состояния автомата в текущий момент времени t и входного сигнала x в текущий момент времени. Выходной сигнал в момент времени t определяется текущим внутренним состоянием и текущим входным сигналом.

Работа автомата Мура описывается соотношениями:

$$\begin{cases} a(t+1) = f(a(t), x(t)), \\ y(t) = \varphi(a(t)). \end{cases}$$

Следующее состояние автомата Мура есть функция от текущего состояния и текущего входного сигнала, а выходной сигнал в текущий момент времени t зависит только от текущего внутреннего состояния.

2.3 Языки описания конечных автоматов

Для задания конечного автомата S требуется описать все элементы множества $S = \{ A, X, Y, f, \varphi, a_0 \}$.

Для описания элементов множества S используется матричное, графовое, теоретико-множественное и табличное представления автомата. [2]

2.3.1 Матричное представление автомата

Для автомата Мили матричная форма состоит из матрицы $C = | c_{ij} |$ размерностью $M \times M$, где каждый элемент матрицы $c_{ij} = x_k / y_s$, стоящий на пересечении i -ой строки и j -го столбца соответствует входному сигналу x_k , вызывающему переход из состояния a_i в состояние a_j с выработкой выходного сигнала y_s .

Для автомата Мура матричная форма состоит из матрицы $C = | c_{ij} |$ размерностью $M \times M$, где каждый элемент матрицы $c_{ij} = x_k$, стоящий на пересечении i -ой строки и j -го столбца, соответствует входному сигналу x_k , вызывающему переход из состояния a_i в состояние a_j . Так как выходной сигнал y_s в автомате Мура зависит только от состояния, следовательно, выходные сигналы могут быть представлены матрицей-столбцом.

2.3.2 Графовое представление автомата

Более наглядно и удобно автомат можно задать с помощью графа автомата. Граф автомата – ориентированный граф, вершины которого соответствуют состояниям, а дуги – переходам между ними. Дугам автомата Мили приписывается входной сигнал, обеспечивающий данный переход и выходной сигнал, который при этом выдаётся. При описании автомата Мура в виде графа выходной сигнал, соответствующий данному состоянию, записывается внутри вершины или рядом с ней.

В графовой форме автомат $S = \{ A, X, Y, f, \varphi, a_0 \}$ представляется графом, в котором:

- множество A изображено вершинами графа;
- функция f задана дугами графа, причём две вершины графа a_i и a_j , соединяются дугой, если в автомате существует переход из a_i в a_j ;
- множество X изображено метками дуг: x_k ставится на дуге из вершины a_i в вершину a_j , если в автомате существует переход из a_i в a_j под действием входного сигнала x_k ;
- функция φ задана метками дуг или вершин: для автомата Мили дуга из вершины a_i в вершину a_j помечается выходным сигналом y_s , если в автомате существует переход из a_i в a_j и при этом вырабатывается выходной сигнал y_s ; а для автомата Мура выходным сигналом y_s помечается вершина, определяющая $a_j = \varphi(a_i)$.

В графе автомата не должно существовать двух дуг с одинаковыми входными сигналами, выходящих из одной и той же вершины (условие однозначности).

Состояние автомата называется тупиковым, если соответствующая вершина графа не содержит исходящих дуг, но имеет хотя бы одну входящую дугу.

Изолированным состоянием называется такое состояние, которому соответствует вершина графа, не имеющая как входящих, так и исходящих дуг.

2.3.3 Теоретико-множественное представление автомата

Для задания конечного автомата $S = \{ A, X, Y, f, \varphi, a_0 \}$ все элементы множества должны быть заданы явно.

Для автомата Мили:

- $A = \{ a_0, a_1, a_2, \dots, a_m \}$ – алфавит состояний;
- $X = \{ x_1, x_2, \dots, x_n \}$ – входной алфавит;
- $Y = \{ y_1, y_2, \dots, y_k \}$ – выходной алфавит;
- $f: A^*X \rightarrow A \left(a_s = f(a_i, x_j) \mid a_s \in A \right)$;
- $\varphi: A^*X \rightarrow Y \left(y_s = \varphi(a_i, x_j) \mid a_s \in A, y_s \in Y \right)$;
- $a_0 \in A$ – начальное состояние автомата;

Для автомата Мура:

- $A = \{ a_0, a_1, a_2, \dots, a_m \}$ – алфавит состояний;
- $X = \{ x_1, x_2, \dots, x_n \}$ – входной алфавит;
- $Y = \{ y_1, y_2, \dots, y_k \}$ – выходной алфавит;
- $f: A^*X \rightarrow A \left(a_s = f(a_i, x_j) \mid a_s \in A \right)$;
- $\varphi: A^*X \rightarrow Y \left(y_s = \varphi(a_i) \mid a_s \in A, y_s \in Y \right)$;
- $a_0 \in A$ – начальное состояние автомата;

2.3.4 Табличное представление автомата

Таблица переходов (выходов) представляет собой таблицу с двойным входом, строки которой нумерованы входными сигналами, а столбцы – состояниями. На пересечении указывается состояние, в которое переходит автомат (в таблице переходов) или выходной сигнал, выдаваемый им (в таблице выходов).

Строки этих таблиц соответствуют входным сигналам, а столбцы – состояниям, причем крайний левый столбец обозначен начальным состоянием a_0 . На пересечении столбца a_i и строки x_j в таблице переходов ставится функция перехода $f(a_i, x_j)$, то есть состояние, в которое автомат переходит из состояния a_i под действием входного сигнала x_j , а в таблице выходов – выходная функция $\varphi(a_i, x_j)$, то есть соответствующий этому переходу выходной сигнал y_k .

Автомат называется частично заданным, если он определен не для всех пар переходов $f(a_i, x_j)$. Для частично заданного автомата на месте отсутствующего перехода ставится прочерк, как в таблице переходов, так и в таблице выходов.

Иногда при задании автомата Мили используют одну совмещенную таблицу переходов и выходов, в которой на пересечении столбца a_i и строки x_j записывают в виде a_k/y_m следующее состояние и выдаваемый выходной сигнал.

Табличная форма задания автомата Мура представляет собой совмещенную (отмеченную) таблицу, в которой выходной сигнал, соответствующий состоянию в a_i автомате Мура размещен в верхней строке над соответствующими состоянием, а остальная информация аналогична представлению автомата Мили.

2.4 Выбор программных средств

2.4.1 Выбор языка программирования

Для разработки программного комплекса была выбрана платформа .NET и язык программирования C#, поскольку данная платформа предоставляет возможность применения технологии создания графического интерфейса – Windows Forms [3]. Данная технология является довольно удобной в использовании и обладает широким спектром возможностей для создания приложений с графическим интерфейсом. Приложения, написанные на языке программирования C# с использованием технологии Windows Forms поддерживаются во всех версиях ОС Microsoft Windows.

2.4.2 Выбор среды разработки

В качестве среды разработки была выбрана интегрированная среда разработки Microsoft Visual Studio Community 2017. Данная IDE предоставляет

возможность комфортной работы с технологией Windows Forms. В качестве альтернативы рассматривалась среда разработки JetBrains Rider, однако данная IDE не предоставляет возможностей для работы с созданием графического интерфейса.

3 Структура программного комплекса

Программный комплекс состоит из двух программ:

- программа для формирования базы вариантов;
- программа для выполнения практической работы.

Для взаимодействия между собой программы используют базу вариантов, в которой хранятся исходные данные, условие задания и представления автоматов Мили и Мура.

Программа для выполнения практической работы позволяет работать с файлами сохранений (сохранять и загружать результаты работы в программе) и формировать отчёт о проделанной работе.

Перед сохранением в базу вариантов все данные шифруются, и только потом сохраняются. При загрузке из базы вариантов, данные сначала загружаются и только потом дешифруются. Шифрование и дешифрование происходит с использованием класса `DESCryptoServiceProvider` из пространства имён `System.Security.Cryptography` [4].

Обе программы защищены от декомпиляции при помощи специального ПО.

Структурная схема программного комплекса представлена на рисунке 1.

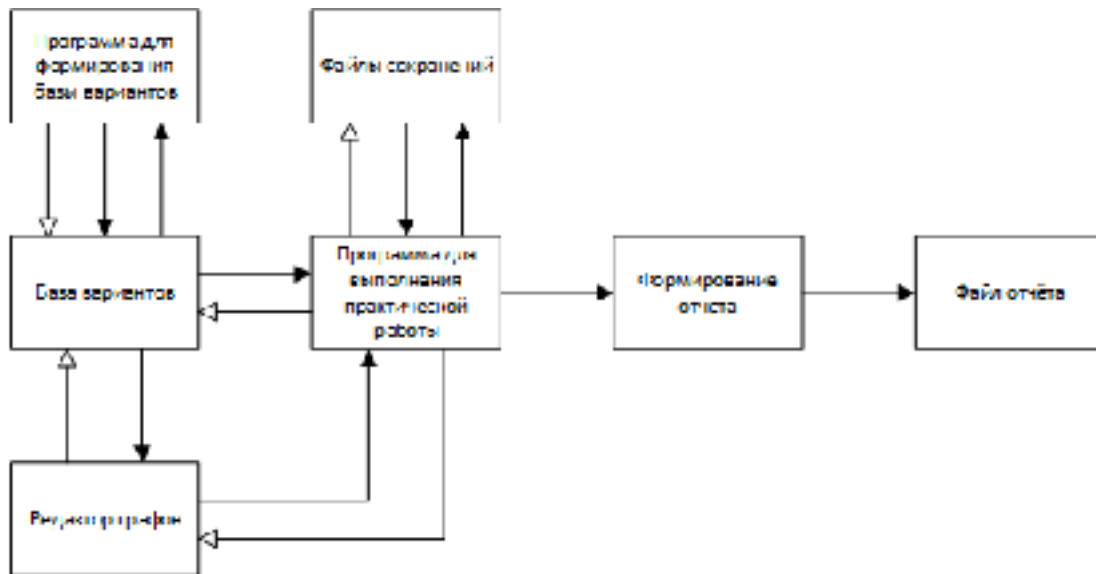


Рисунок 1 – Структурная схема программного комплекса

3.1 Структура программы для формирования базы вариантов

Программа для формирования базы вариантов состоит из трёх классов:

- класс Program;
- класс (форма) StartWindow;
- класс (форма) MainWindow.

Для отражения общей структуры программы была разработана структурная схема классов, изображённая на рисунке 2, состоящая из классов и их методов.

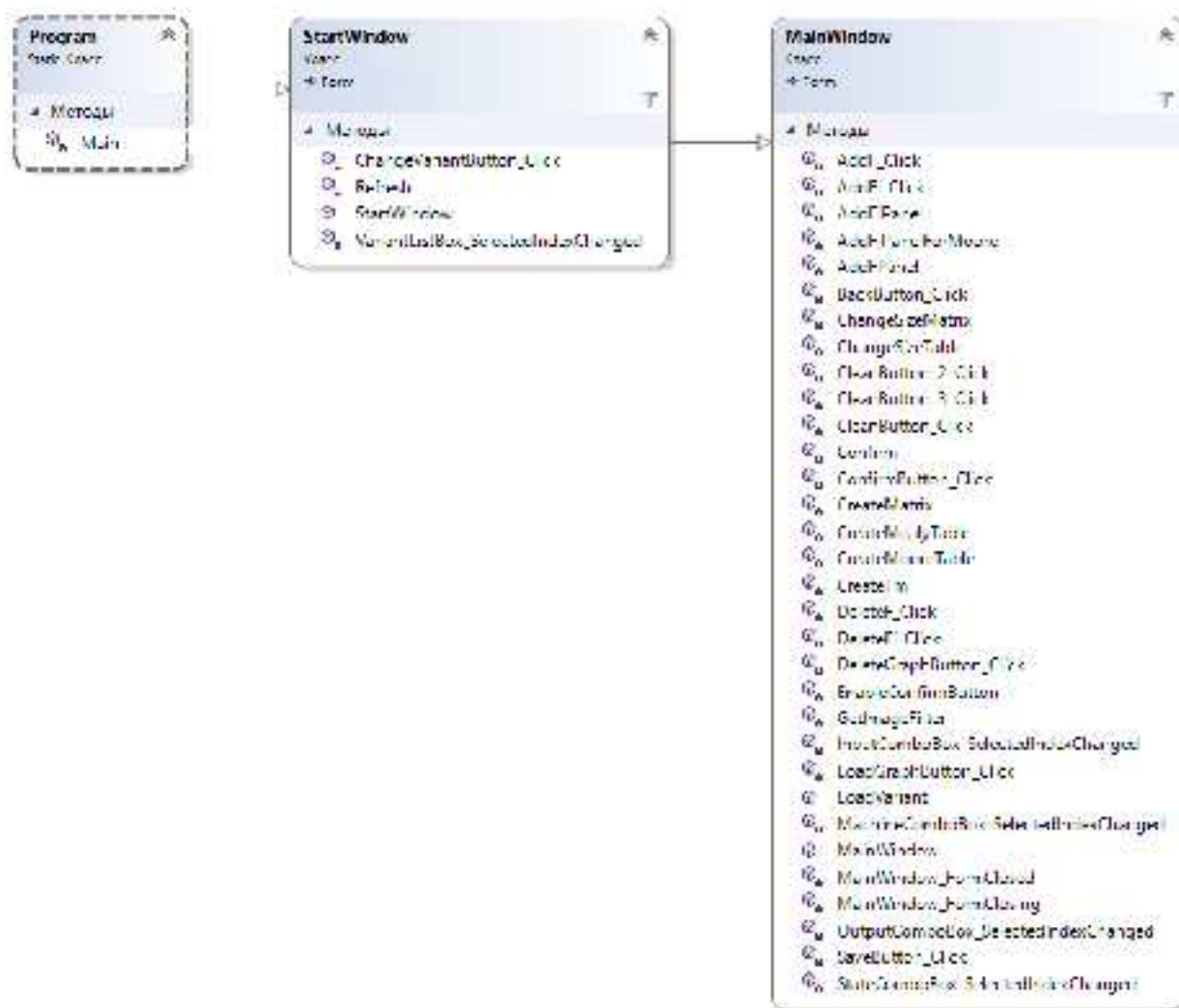


Рисунок 2 – Структурная схема классов программы для формирования базы вариантов

Класс Program является отправной точкой программы и содержит единственный метод Main, открывающий форму StartWindow.

Класс (форма) StartWindow является формой для выбора варианта для редактирования и просмотра краткой информации о выбранном варианте. Основные методы:

- VariantListBox_SelectedIndexChanged – загружает и дешифрует информационный файл из базы вариантов для выбранного варианта и выводит краткую информацию о нём;

– ChangeVariantButton_Click – открывает форму MainWindow и загружает данные о выбранном варианте из базы вариантов для дальнейшего редактирования;

Класс (форма) MainWindow является формой для редактирования варианта и сохранения изменений в базе вариантов. Основные методы:

– LoadVariant – загружает данные из базы вариантов, дешифрует их и выводит на рабочую область;

– CreateMatrix – создаёт шаблон матричного представления автомата, основываясь на исходных данных;

– CreateMealyTable – создаёт шаблон табличного представления автомата Мили, основываясь на исходных данных;

– CreateMooreTable – создаёт шаблон табличного представления автомата Мура, основываясь на исходных данных;

– CreateTm – создаёт шаблон теоретико-множественного представления автомата, основываясь на исходных данных;

– LoadGraphButton_Click – загружает изображение графового представления на рабочую область;

– DeleteGraphButton_Click – удаляет изображение графового представления из рабочей области;

– SaveButton_Click_Click – шифрует данные о варианте и сохраняет их в базу вариантов;

Блок-схема программы для формирования базы вариантов представлена на рисунке 3.

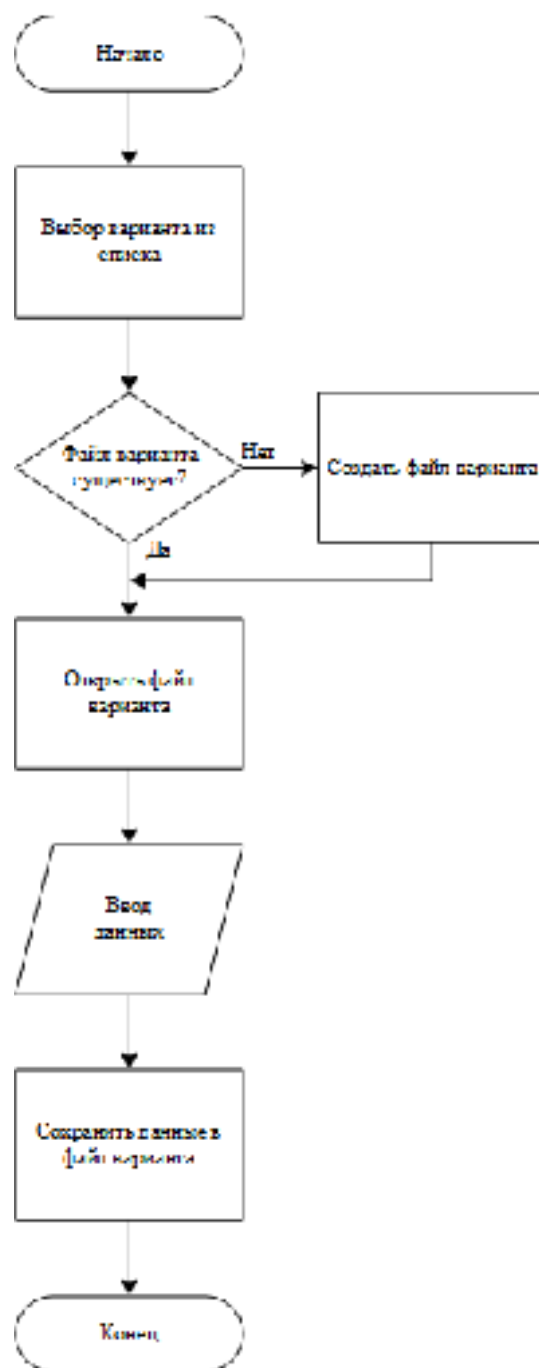


Рисунок 3 – Блок-схема программы для формирования базы вариантов

3.2 Структура программы для выполнения практической работы

Программа для выполнения практической работы состоит из трёх классов:

- класс Program;
- класс (форма) MainForm;
- класс (форма) ReportForm.

Для отражения общей структуры программы была разработана структурная схема классов, изображённая на рисунке 4, состоящая из классов и их методов.

Класс Program является отправной точкой программы и содержит единственный метод Main, открывающий форму MainForm.

Класс (форма) MainForm является формой для выполнения практической работы и проверки результатов. Основные методы:

- VariantComboBox_SelectedIndexChanged – загружает и дешифрует задание из базы вариантов для выбранного варианта и выводит задание на экран;

- CheckMatrixButton_Click – сверяет матричное представление автомата, введённое пользователем, с тем, что хранится в базе вариантов. Выводит сообщение об ошибке при несовпадении;

- CheckTableButton_Click – сверяет табличное представление автомата, введённое пользователем, с тем, что хранится в базе вариантов. Выводит сообщение об ошибке при несовпадении;

- CheckTMButton_Click – сверяет теоретико-множественное представление автомата, введённое пользователем, с тем, что хранится в базе вариантов. Выводит сообщение об ошибке при несовпадении;

- EnableDocxButton – включает кнопку создания отчёта, если вся введённая пользователем информация прошла проверку;

- DocxButtonClick – открывает форму ReportForm для создания отчёта.

Класс (форма) ReportForm является формой для создания отчёта. Позволяет пользователю ввести исходные данные для автоматического формирования отчёта. Содержит метод ConfirmButton_Click, запускающий процесс создания отчёта.

Структурная схема программы для выполнения практической представлена на рисунке 5.

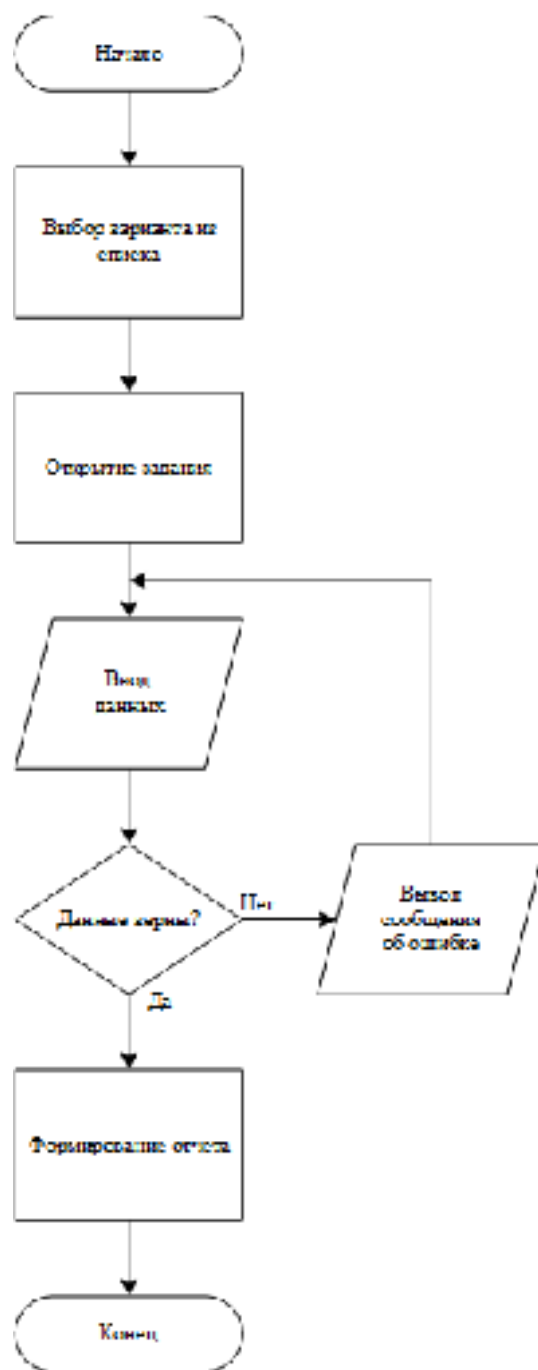


Рисунок 5 – Блок-схема программы для выполнения практической работы

4 Формирование отчёта

Для формирования отчета по выполненной практической работе используется заранее подготовленный шаблон. Оформление шаблона отчёта выполнено с учетом стандарта организации «Общие требования к построению, изложению и оформлению документов учебной деятельности» (СТО 4.2-07-2014) [5].

Для заполнения шаблона используются закладки с уникальным текстом. Данный подход позволяет преподавателю редактировать шаблон по собственному желанию, сохраняя его функционал.

Шаблон титульной страницы с закладками представлен на рисунке 6.

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра вычислительной техники

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ №%LabNum%
«Язык описания конечных автоматов»
Вариант %VarNum%

Преподаватель		<u>%TeacherName%</u> Инициалы, фамилия
	подпись, дата	
Студент <u>%Group%</u> <u>%BookN%</u>		<u>%StudentName%</u> Инициалы, фамилия
номер группы, номер книги	подпись, дата	

Красноярск %Year%

Рисунок 6 – Шаблон титульной страницы с закладками

Шаблон содержит разделы:

- титульный лист;
- цель и задание;
- решение.

Титульный лист содержит закладки:

- %LabNum%, для замены на номер практической работы;
- %VarNum%, для замены на номер варианта;
- %TeacherName%, для замены на ФИО преподавателя;
- %Group%, для замены на номер группы;
- %BookN%, для замены на номер зачётной книжки;
- %StudentName%, для замены на ФИО студента;
- %Year%, для замены текущий год.

Цель и задание содержат закладки:

- %Machine%, для замены на тип автомата;
- %Task%, для замены на задание;
- %See%, для замены на изображение представления.

Решение содержит закладки:

- %DoneText1%, для замены на название представления;
- %DonePic1%, для замены на изображение представления;
- %DoneText2%, для замены на название представления;
- %DonePic2%, для замены на изображение представления;
- %DoneText3%, для замены на название представления;
- %DonePic3%, для замены на изображение представления.

На рисунке 7 представлена блок-схема алгоритма формирования отчёта.

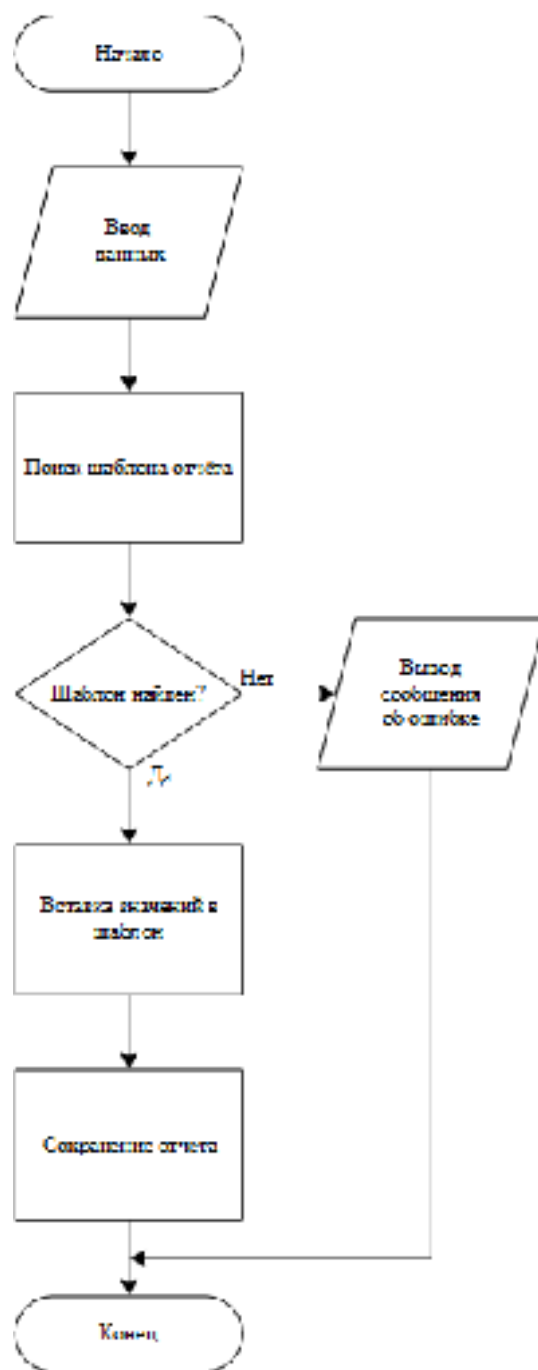


Рисунок 7 – Блок-схема алгоритма формирования отчета

5 Руководство пользователя

5.1 Программа для формирования базы вариантов

Программа для формирования базы вариантов позволяет пользователю просматривать и редактировать варианты для практической работы.

Пользовательский интерфейс программы разделен на две формы:

- форма «База вариантов»;
- форма «Редактор базы вариантов».

На рисунке 8 представлена форма «База вариантов».

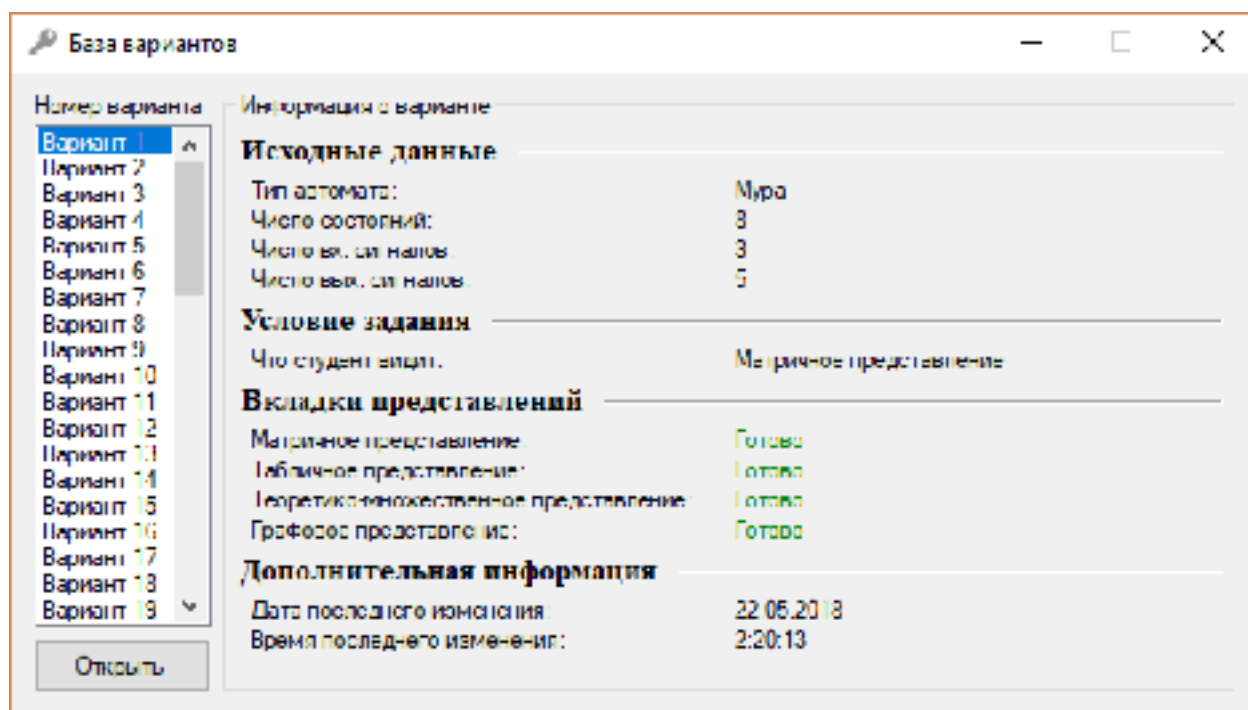


Рисунок 8 – Форма «База вариантов»

На форме «База вариантов» пользователю доступен список «Номер варианта», содержащий 60 вариантов. Выделив вариант из списка левой кнопкой мыши, пользователь может увидеть краткую информацию о выделенном варианте в области «Информация о варианте», если выделенный вариант уже имеется в

базе вариантов. Если выделенного варианта в базе вариантов нет, то область «Информация о варианте» будет пуста.

Нажатие на кнопку «Открыть» на форме «База вариантов» открывает форму «Редактор базы вариантов» и загружает данные о выделенном варианте в эту форму для дальнейшего редактирования. Если выделенного варианта в базе вариантов нет, то открывается пустая форма «Редактор базы вариантов», позволяющая создать выделенный вариант.

На рисунке 9 представлена форма «Редактор базы вариантов» и элементы управления.

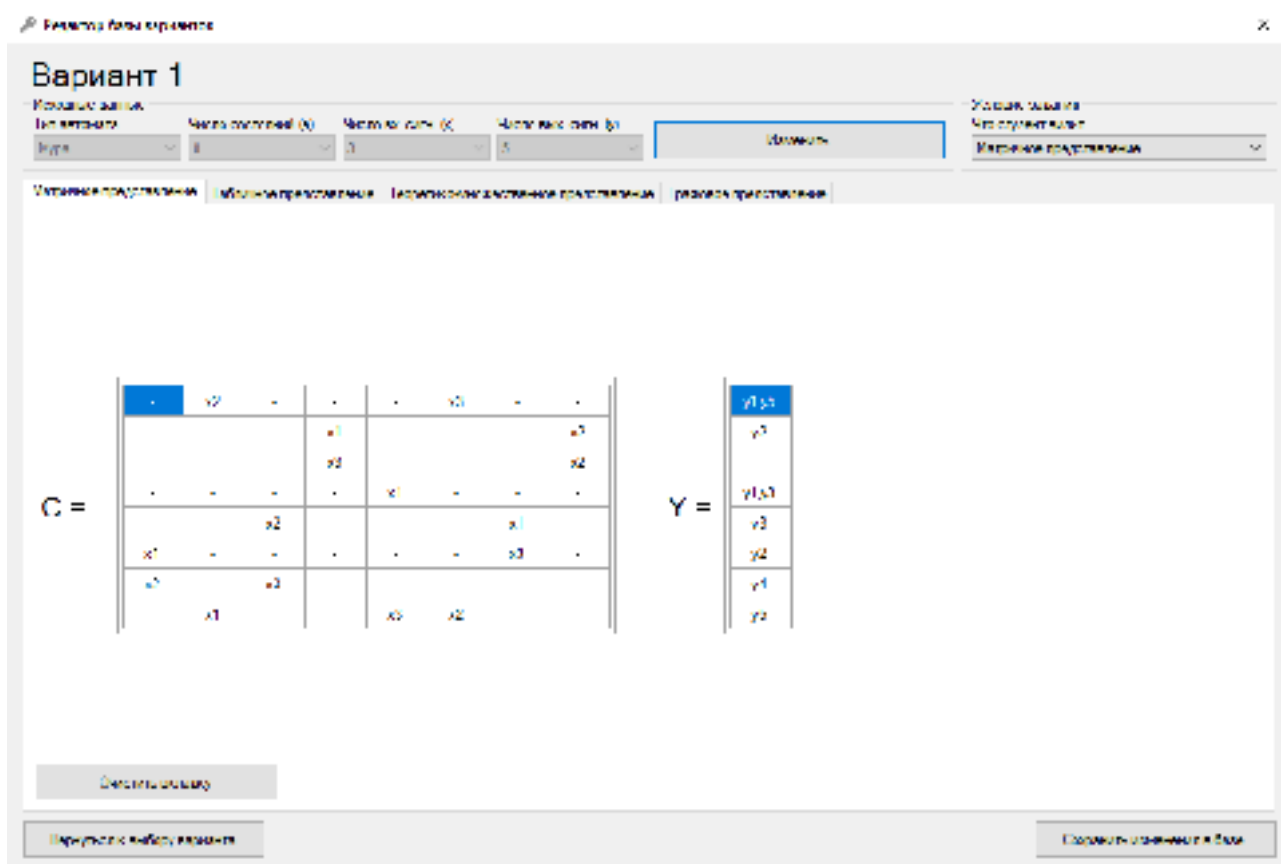


Рисунок 9 – Форма «Редактор базы вариантов» и элементы управления

На форме «Редактор базы вариантов» пользователю доступна область «Исходные данные», позволяющая выбрать необходимые исходные данные и

сформировать шаблоны всех представлений автомата, основываясь на этих данных.

На рисунке 10 представлена область «Исходные данные».

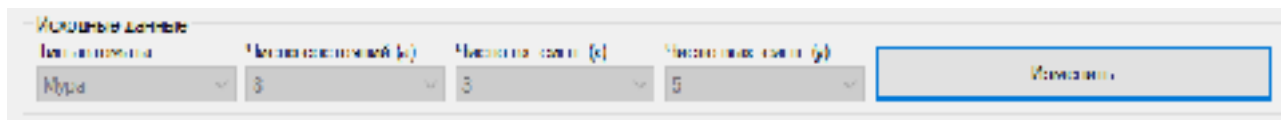


Рисунок 10 – Область «Исходные данные»

Область «Исходные данные» содержит четыре выпадающих списка и одну кнопку.

Выпадающий список «Тип автомата» позволяет пользователю выбрать тип автомата.

Выпадающий список «Число состояний (a)» позволяет пользователю задать число состояний автомата.

Выпадающий список «Число вх. сигн. (x)» позволяет пользователю задать число входных сигналов автомата.

Выпадающий список «Число вых. сигн. (y)» позволяет пользователю задать число выходных сигналов автомата.

Кнопка «Принять» активируется, когда пользователь заполнит все выпадающие списки, указанные выше. Нажатие на кнопку «Принять» создаёт в рабочей области вкладки «Матричное представление», «Табличное представление», «Теоретико-множественное представление» и «Графовое представление» и формирует шаблоны каждого из представлений автомата соответственно. После нажатия на кнопку «Принять» все выпадающие списки из области «Исходные данные» блокируются для редактирования, а кнопка «Принять» изменяется на кнопку «Изменить».

Нажатие на кнопку «Изменить» удаляет все вкладки из рабочей области. После нажатия на кнопку «Изменить» все выпадающие списки из области

«Исходные данные» становятся доступными для редактирования, а кнопка «Изменить» изменяется на кнопку «Принять».

Рабочая область с активной вкладкой «Матричное представление» представлена на рисунке 11.

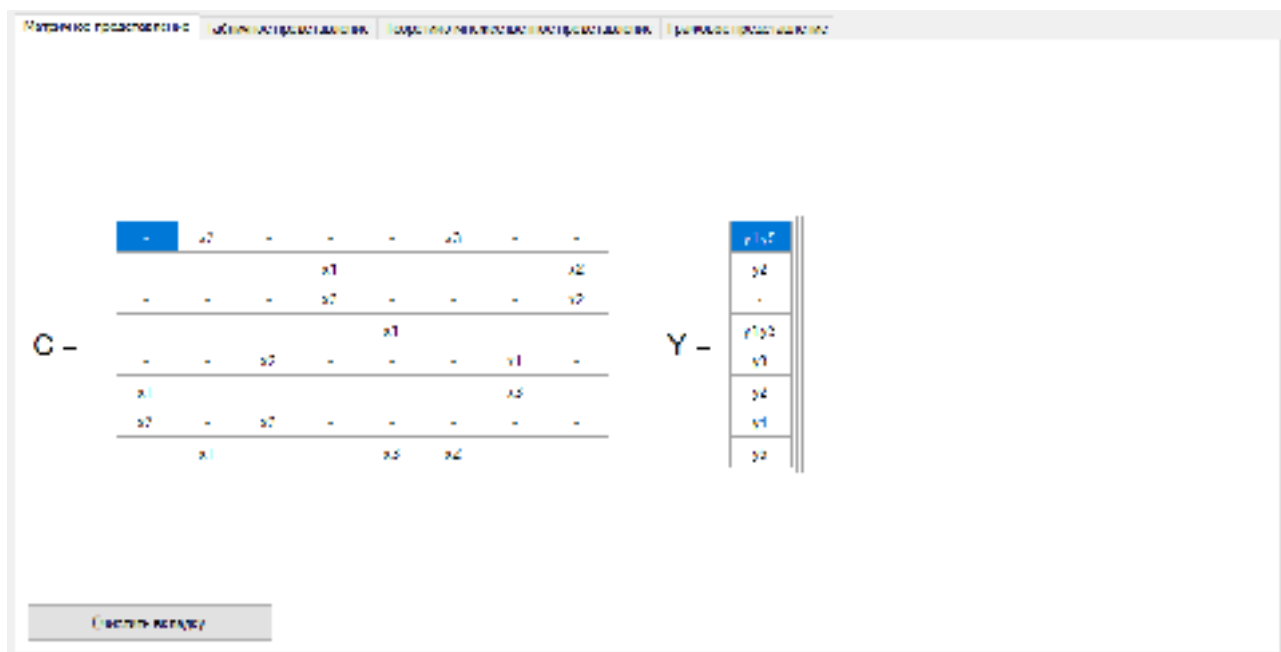


Рисунок 11 – Рабочая область с активной вкладкой «Матричное представление»

Вкладка «Матричное представление» содержит одну или две матрицы, в зависимости от типа автомата и одну кнопку.

Пользователь может выделить любую ячейку матрицы левой кнопкой мыши и ввести с клавиатуры необходимое значение.

Нажатие на кнопку «Очистить вкладку» полностью очищает все имеющиеся на вкладке матрицы.

Рабочая область с активной вкладкой «Табличное представление» представлена на рисунке 12.

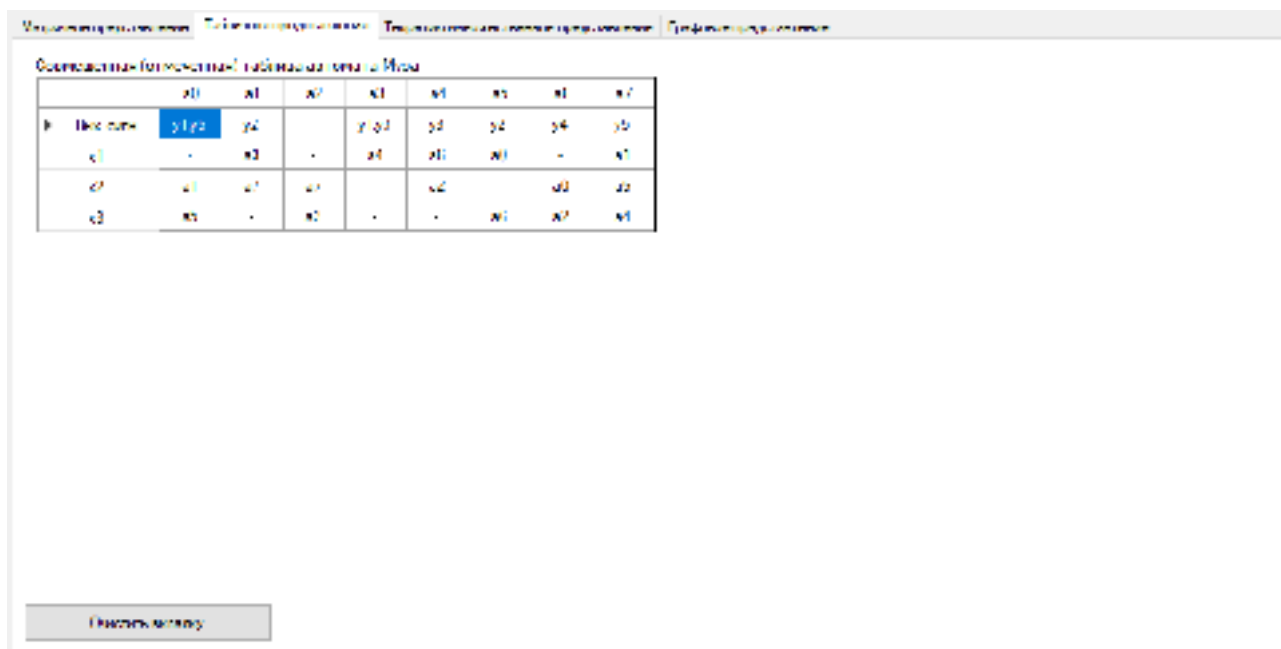


Рисунок 12 – Рабочая область с активной вкладкой «Табличное представление»

Вкладка «Табличное представление» содержит совмещённую таблицу автомата Мура или совмещённую (отмеченную) таблицу автомата Мили, в зависимости от типа автомата и одну кнопку.

Пользователь может выделить любую ячейку таблицы левой кнопкой мыши и ввести с клавиатуры необходимое значение.

Нажатие на кнопку «Очистить вкладку» полностью очищает таблицу.

Рабочая область с активной вкладкой «Теоретико-множественное представление» представлена на рисунке 13.

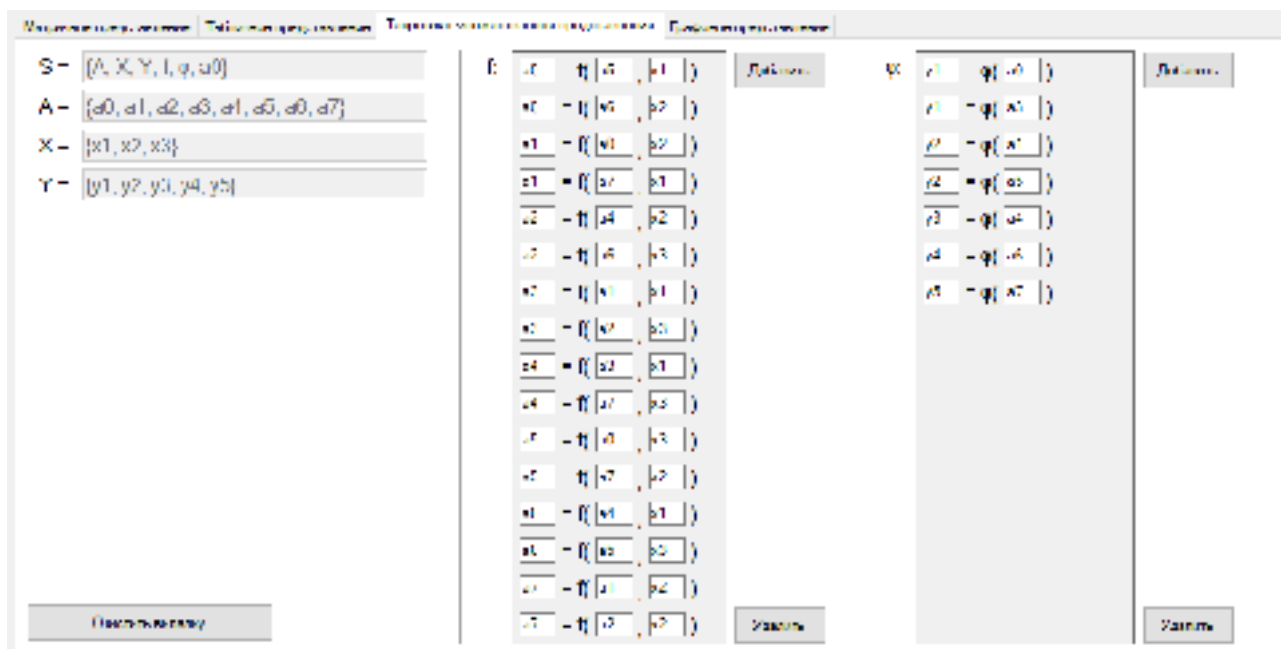


Рисунок 13 – Рабочая область с активной вкладкой «Теоретико-множественное представление»

Вкладка «Теоретико-множественное представление» содержит множество конечного автомата, множество состояний автомата, множество входных сигналов автомата, множество выходных сигналов автомата, область для создания функций переходов автомата, область для создания функций выходов автомата и пять кнопок.

Множество конечного автомата, множество состояний автомата, множество входных сигналов автомата, множество выходных сигналов автомата заблокированы для редактирования пользователем и заполняются автоматически, в зависимости от исходных данных.

Область для создания функций переходов автомата «f:» содержит две кнопки.

Нажатие на кнопку «Добавить» добавляет функцию перехода автомата в конец области для создания функций переходов автомата «f:».

Нажатие на кнопку «Удалить» удаляет функцию перехода автомата с конца области для создания функций переходов автомата «f:».

Область для создания функций выходов автомата «φ:» содержит две кнопки.

Нажатие на кнопку «Добавить» добавляет функцию выхода автомата в конец области для создания функций выходов автомата «ф:».

Нажатие на кнопку «Удалить» удаляет функцию выхода автомата с конца области для создания функций выходов автомата «ф:».

Пользователь может выделить любую функцию перехода или выхода левой кнопкой мыши и ввести с клавиатуры необходимое значение.

Нажатие на кнопку «Очистить вкладку» удаляет все функции переходов и выходов автомата.

Рабочая область с активной вкладкой «Графовое представление» представлена на рисунке 14.

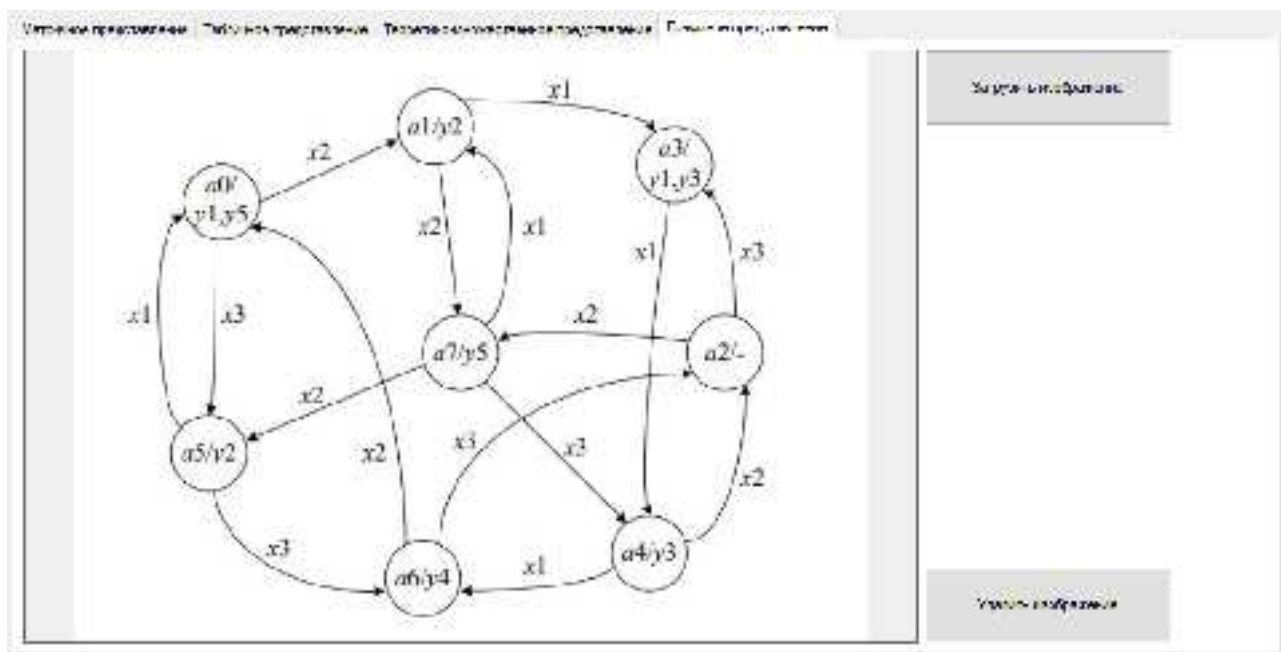


Рисунок 14 – Рабочая область с активной вкладкой «Графовое представление»

Вкладка «Графовое представление» содержит область просмотра изображения и две кнопки.

Нажатие на кнопку «Загрузить изображение» открывает проводник для выбора необходимого изображения. После выбора изображения оно отображается в области просмотра изображения.

Нажатие на кнопку «Удалить изображение» очищает область просмотра изображения.

Также на форме «Редактор базы вариантов» пользователю доступна область «Условие задания», содержащая один выпадающий список «Что студент видит». С помощью этого списка пользователь может назначить, какое именно представление автомата будет видеть студент, выполняющий практическую работу.

На рисунке 15 представлена область «Условие задания».

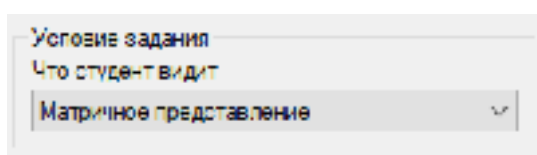


Рисунок 15 – Область «условие задания»

Для взаимодействия с базой вариантов пользователю доступны две кнопки.

Нажатие на кнопку «Вернуться к выбору варианта» закрывает форму «Редактор базы вариантов» без сохранения изменений в базе вариантов.

Нажатие на кнопку «Сохранить изменения в базе» сохраняет текущее состояние варианта в базе вариантов.

Кнопки для взаимодействия с базой вариантов представлены на рисунке 16.

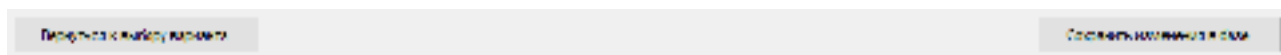


Рисунок 16 – Кнопки для взаимодействия с базой вариантов

Также в программе для формирования базы вариантов предусмотрены информационные сообщения для различных ситуаций.

На рисунке 17 представлены все информационные сообщения.

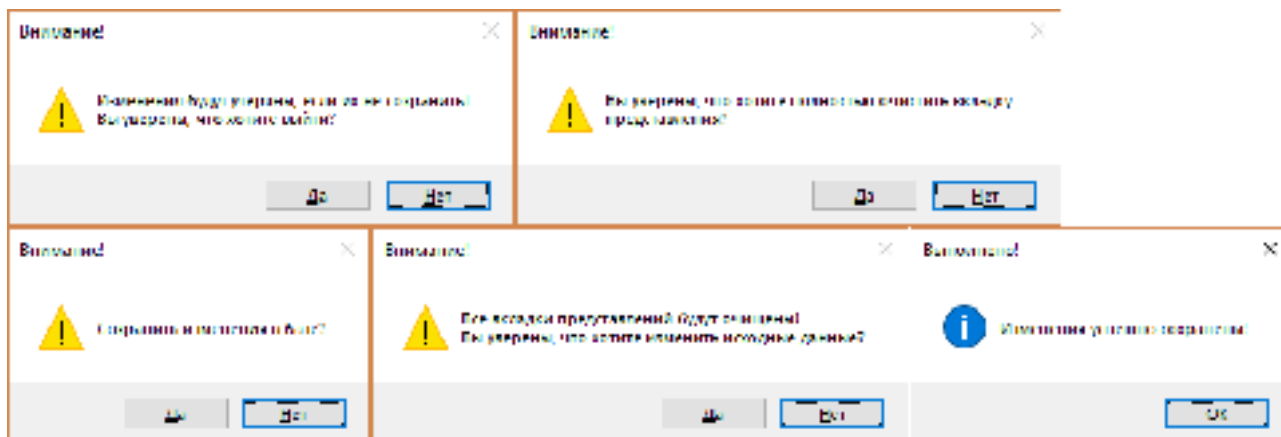


Рисунок 17 – Информационные сообщения

5.2 Программа для выполнения практической работы

Программа для выполнения практической работы позволяет пользователю выбрать вариант, загрузить задание для выбранного варианта из базы вариантов, предоставляет полный набор функций для выполнения полученного задания и позволяет автоматически сформировать отчет о проделанной работе.

Пользовательский интерфейс программы разделен на две формы:

- форма «Практическая работа»;
- форма «Создание отчёта».

На рисунке 18 представлена форма «Практическая работа» и элементы управления.

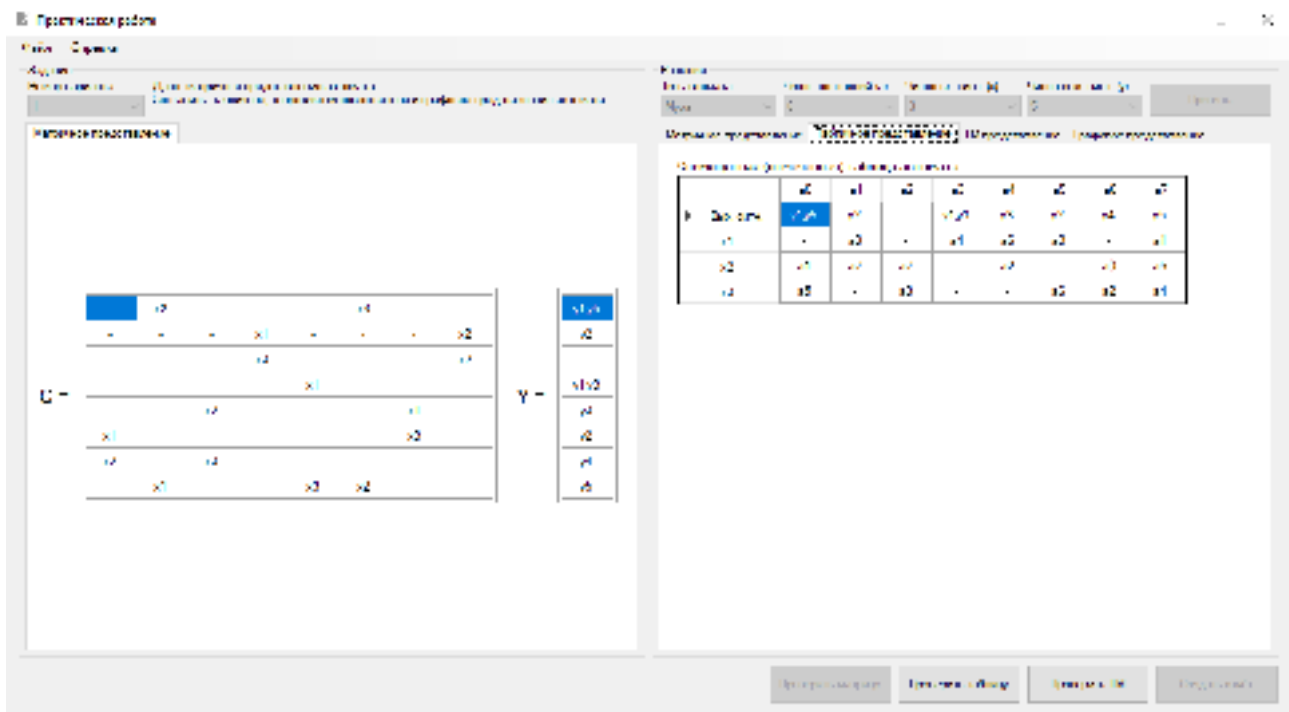


Рисунок 18 – Форма «Практическая работа»

На форме «Практическая работа» пользователю доступна область «Задание», содержащая один выпадающий список «Номер варианта», текст задания, генерируемый автоматически и одну вкладку представления автомата, которая соответствует заданию.

При изменении номера варианта в выпадающем списке «Номер варианта», текст задания и вкладка представления автомата меняются, в соответствии с выбранным вариантом.

Любое представление автомата в данной области заблокировано для редактирования и доступно только для просмотра.

На рисунке 19 представлена область «Задание».

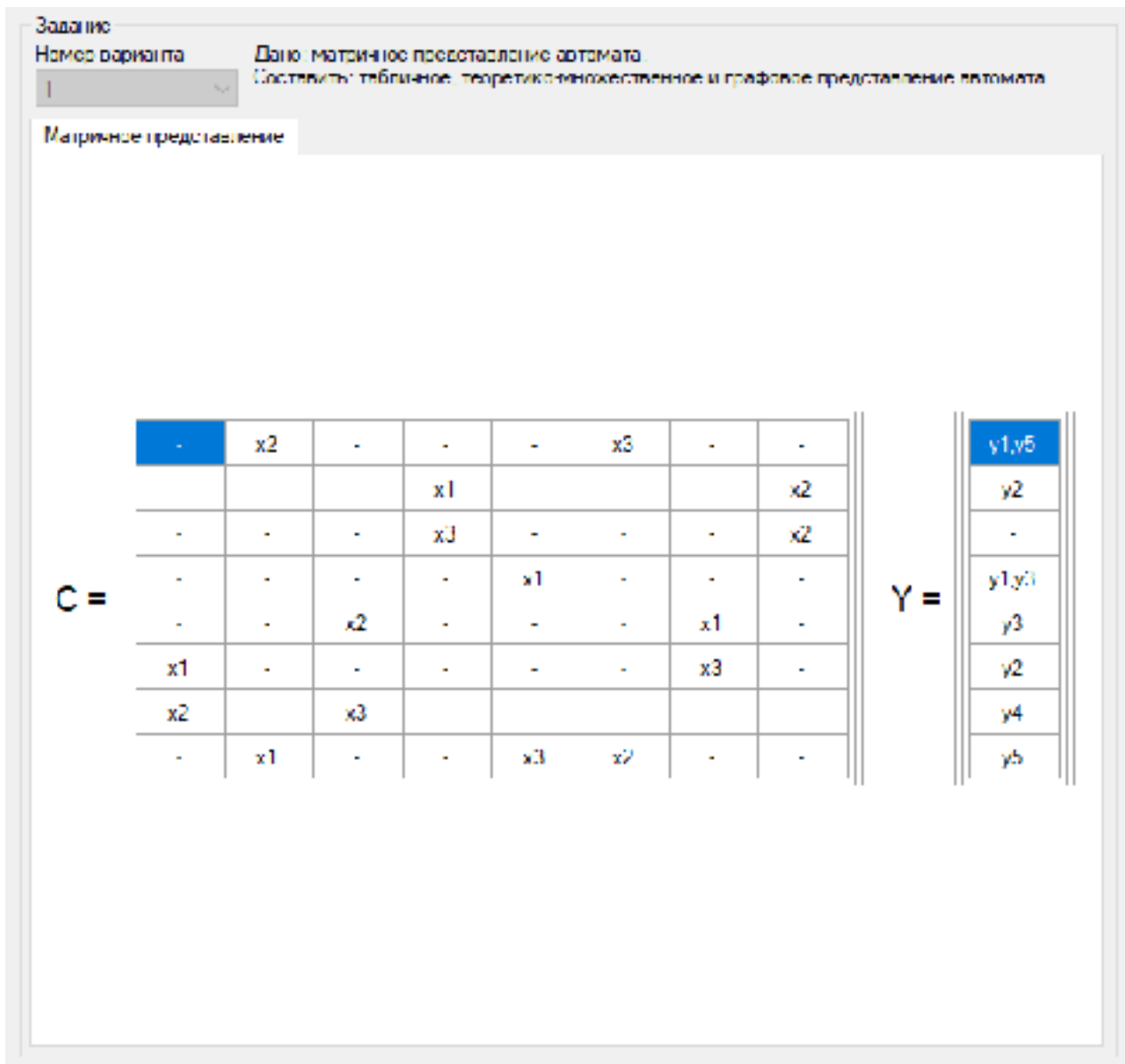


Рисунок 19 – Область «Задание»

На форме «Практическая работа» пользователю доступна область «Решение», содержащая четыре выпадающих списка, одну кнопку и четыре вкладки представления автомата.

Выпадающие списки полностью дублируют функционал аналогичных списков из программы для формирования базы вариантов.

Кнопка «Принять» активируется, когда пользователь заполнит все выпадающие списки, указанные выше. Нажатие на кнопку «Принять» создаёт в

рабочей области вкладки «Матричное представление», «Табличное представление», «Теоретико-множественное представление» и «Графовое представление» и формирует шаблоны каждого из представлений автомата соответственно. После нажатия на кнопку «Принять» все выпадающие списки из области «Решение» блокируется для редактирования, а кнопка «Принять» становится неактивной.

На рисунке 20 представлена область «Решение».

Решение

Тип автомата: Муса

Число состояний (n): 3

Число вх. сигн. (k): 3

Число вых. сигн. (y): 5

Принять

Матричное представление | Табличное представление | ТМ представление | Графовое представление

$$C = \begin{pmatrix} - & x_2 & - & - & - & x_3 & - & - \\ - & - & - & x_1 & - & - & - & x_2 \\ & & & x_3 & & & & x_2 \\ - & - & - & - & x_1 & - & - & - \\ - & - & x_2 & - & - & - & x_1 & - \\ x_1 & - & - & - & - & - & x_3 & - \\ x_2 & & x_3 & & & & & \\ & x_1 & & & x_3 & x_2 & & \end{pmatrix} \quad Y = \begin{pmatrix} y_1 y_5 \\ y_2 \\ y_1 y_3 \\ y_3 \\ y_2 \\ y_4 \\ y_5 \end{pmatrix}$$

Рисунок 20 – Область «Решение»

Все вкладки представлений автомата в области «Решение» полностью дублируют функционал аналогичных вкладок из программы для формирования базы вариантов, кроме вкладки «Графовое представление».

Если вкладка дана по условию задания, то она заполняется автоматически и блокируется для редактирования (полностью дублирует аналогичную вкладку в области «Задание»).

Если по условию задания студенту необходимо составить графовое представление автомата, то на вкладке «Графовое представление» появится кнопка «Открыть редактор графов», при нажатии которой открывается стороннее приложение для создания графового представления автоматов в отдельном окне. После завершения работы, это приложение должно сохранить полученный граф в виде изображения, которое автоматически загрузится во вкладку «Графовое представление», а кнопка «Открыть редактор графов» будет удалена.

На рисунке 21 представлена рабочая область с активной вкладкой «Графовое представление».

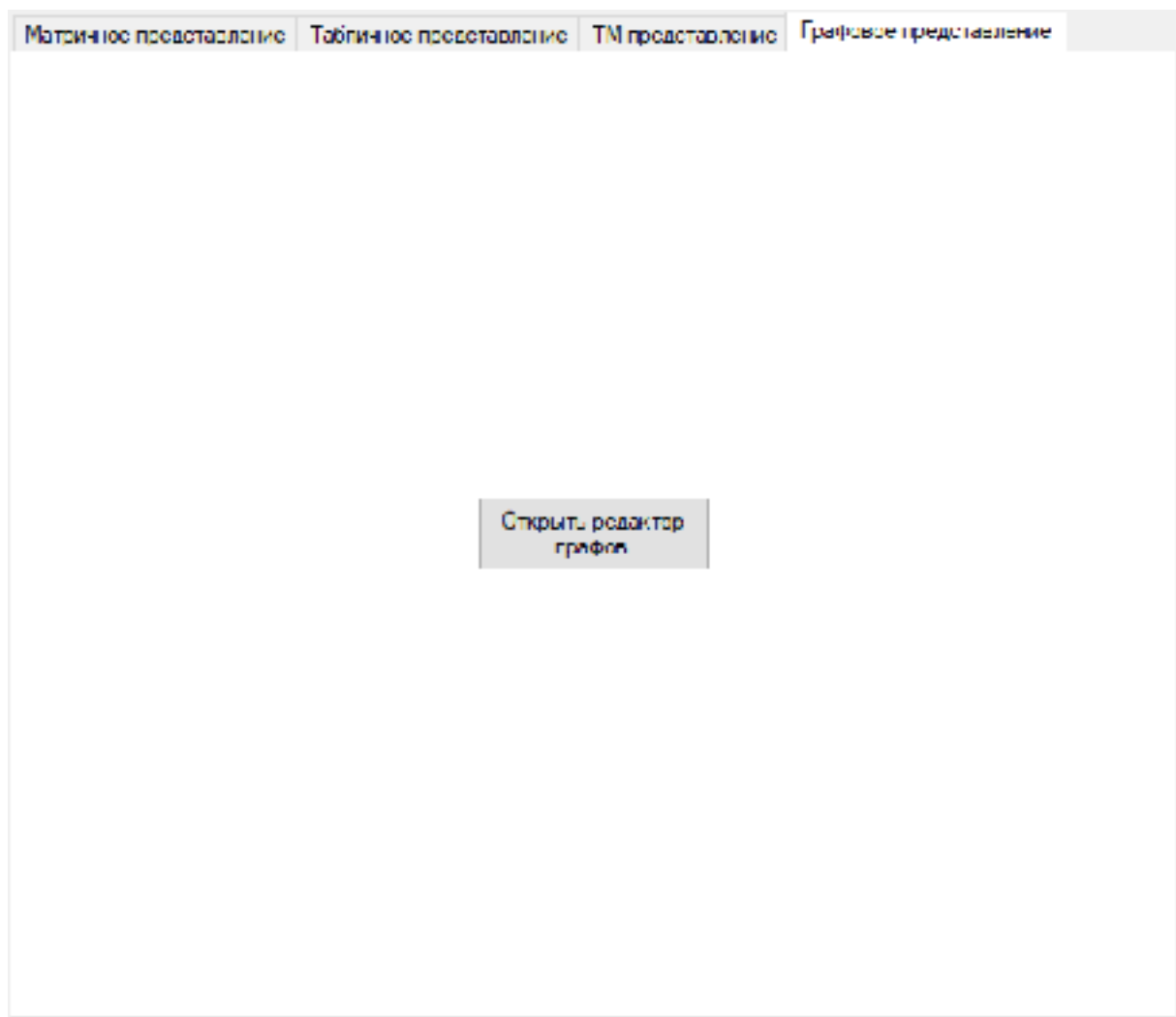


Рисунок 21 – Рабочая область с активной вкладкой «Графовое представление»

Если редактор графов не найден, то вместо изображения графа во вкладку «Графовое представление» загружается изображение, сообщающее об ошибке. Изображение, сообщающее об ошибке открытия редактора графов представлено на рисунке 22.

РЕДАКТОР ГРАФОВ НЕ НАЙДЕН

Рисунок 22 – Изображение, сообщающее об ошибке открытия редактора графов

На форме «Практическая работа» пользователю доступно три кнопки для проверки правильности выполнения задания. Кнопки для проверки правильности выполнения задания представлены на рисунке 23.

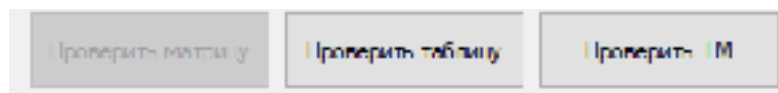


Рисунок 23 – Кнопки для проверки правильности выполнения задания

Нажатие на кнопку «Проверить матрицу» проверяет, соответствует ли матричное представление, введённое пользователем тому, которое хранится в базе вариантов для текущего варианта. Если проверка пройдена успешно, то данная кнопка становится неактивной.

Нажатие на кнопку «Проверить таблицу» проверяет, соответствует ли табличное представление, введённое пользователем тому, которое хранится в базе вариантов для текущего варианта. Если проверка пройдена успешно, то данная кнопка становится неактивной.

Нажатие на кнопку «Проверить ТМ» проверяет, соответствует ли теоретико-множественное представление, введённое пользователем тому, которое хранится в базе вариантов для текущего варианта. Если проверка пройдена успешно, то данная кнопка становится неактивной.

Если одно из представлений дано по условию задания, то кнопка для проверки этого представления становится неактивной.

Если все необходимые проверки были пройдены и изображение графа (либо изображение, сообщающее об ошибке) было загружено, то активируется кнопка «Создать отчёт».

Кнопка «Создать отчёт» представлена на рисунке 24.

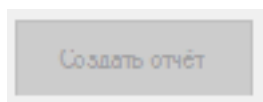


Рисунок 24 – Кнопка «Создать отчёт»

Нажатие на кнопку «Создать отчёт» открывает форму «Создание отчёта», представленную на рисунке 25.

A screenshot of a web form titled «Создание отчёта». The form has a title bar with a close button (X) and a maximize button (L). It contains five input fields: «Номер практической работы:», «И.О. Фамилия преподавателя:», «И.О. Фамилия студента:», «Группа:», and «Номер ленточной книжки:». Each field is a simple text box. At the bottom right of the form is a button labeled «Гринтъ».

Рисунок 25 – Форма «Создание отчёта»

Форма «Создание отчёта» содержит пять полей для ввода данных, необходимых для автоматического формирования отчёта по практической работе и одну кнопку.

Если все вышеуказанные поля заполнены, то при нажатии кнопки «Принять» открывается проводник для указания места сохранения отчёта в формате «.docx», после чего форма «Создание отчёта» закрывается.

Также на форме «Практическая работа» присутствует меню, состоящее из двух подменю, представленное на рисунке 26.

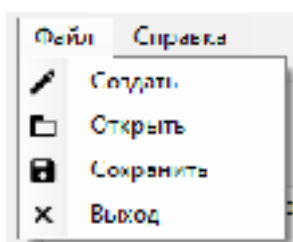


Рисунок 26 – Меню

Подменю «Файл» состоит из четырёх пунктов:

- «Создать» – полностью очищает рабочую область
- «Открыть» – открывает ранее сохранённый файл и загружает его содержимое в рабочую область
- «Сохранить» – сохраняет в файл текущее состояние рабочей области
- «Выход» – закрывает программу

Подменю «Справка» не содержит пунктов и открывает файл «help.htm», содержащийся в каталоге с программным комплексом. Это позволяет пользователю редактировать справку по своему усмотрению.

Также в программе для выполнения практической работы предусмотрено множество информационных сообщений для различных ситуаций.

На рисунке 27 представлены некоторые информационные сообщения.

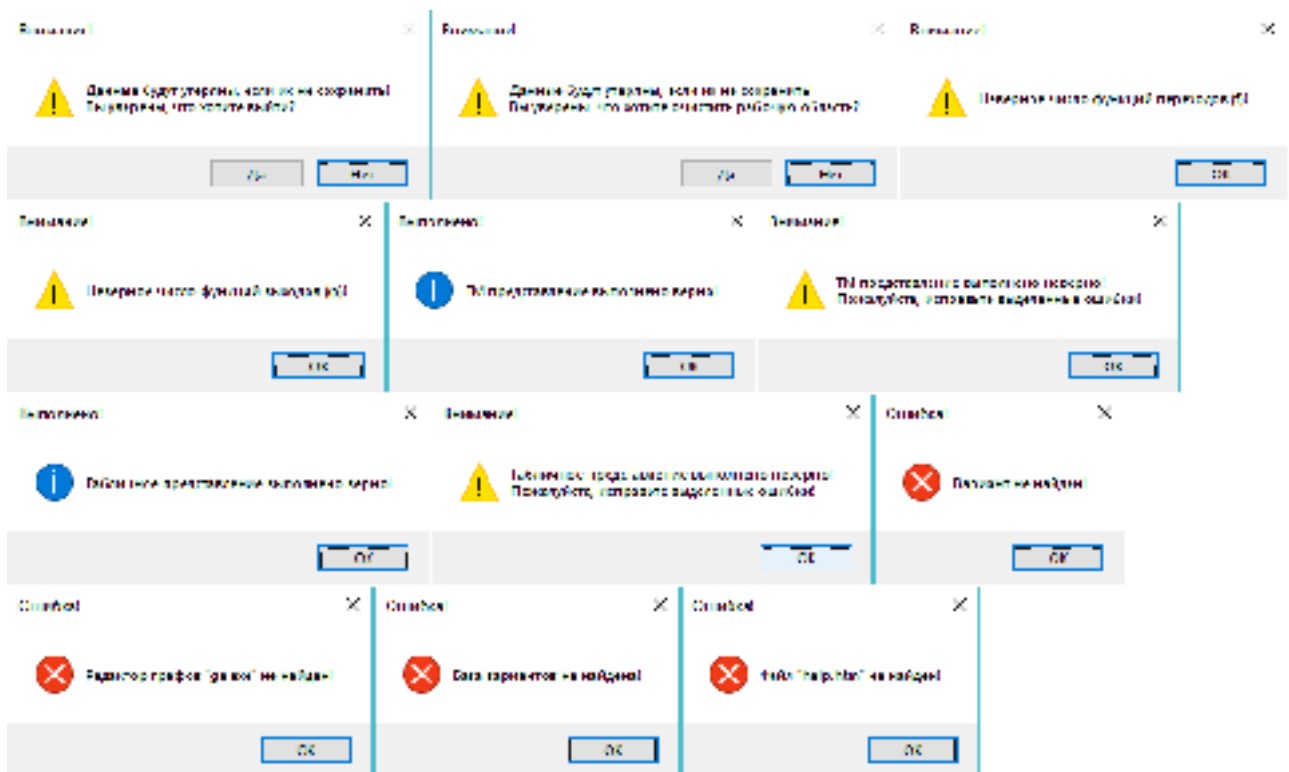


Рисунок 27 – Информационные сообщения

ЗАКЛЮЧЕНИЕ

В выпускной квалификационной работе были поставлены и выполнены следующие задачи:

- Произведен теоретический анализ предметной области;

- Сформулированы требования к проектируемому программному комплексу. Программный комплекс должен обеспечивать ввод, удаление и сохранение данных в базе вариантов для преподавателя. Студентам предоставляется возможность выполнить практическую работу и автоматически сформировать отчёт. Суть практической работы заключается в составлении различных представлений конечных автоматов на основе имеющихся представлений. Кроме этого, программа должна обеспечивать возможность сохранения и загрузки результатов работы;

- Выбраны средства для разработки программного комплекса. В качестве языка программирования был выбран C#, в качестве среды программирования – Microsoft Visual Studio Community 2017. Программный комплекс предназначен для работы в ОС семейства Windows, начиная с Windows 7 по условиям задания;

- Разработана структурная схема программного комплекса;

- Реализован программный продукт, что и являлось основной целью данной работы.

СПИСОК СОКРАЩЕНИЙ

ПТЦА – прикладная теория цифровых автоматов;

ПО – программное обеспечение;

ОС – операционная система;

IDE – Integrated Development Environment (интегрированная среда разработки).

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Постников, А. И. Теория автоматов и машинная арифметика : учеб. пособие для студентов вузов / А. И. Постников, Е. А. Вейсов – Красноярск : ИПЦ КГТУ, 2006. - 376 с
2. Представления автомата [Электронный ресурс] – Режим доступа: <https://studfiles.net/preview/5971014/page:32/>.
3. Полное руководство C# [Электронный ресурс] – Режим доступа: <https://metanit.com/sharp/tutorial/>.
4. Пространство имён System.Security.Cryptography [Электронный ресурс] – Режим доступа: [https://msdn.microsoft.com/ru-ru/library/system.security.cryptography\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/system.security.cryptography(v=vs.110).aspx).
5. СТО 4.2-07-2014 Стандарт организации «Общие требования к построению, изложению и оформлению документов учебной деятельности. – Красноярск : ИПК СФУ, 2014. – 60 с.

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Космических и информационных технологий

институт

Вычислительная техника

кафедра

УТВЕРЖДАЮ

Заведующий кафедрой

О.В. Непомнящий

инициалы, фамилия

«__» _____ 2018 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.01 Информатика и вычислительная техника

код и наименование направления

Программный комплекс для автоматизации проверки выполнения
практической работы «Языки описания конечных автоматов» по дисциплине
«Прикладная теория цифровых автоматов»

тема

Руководитель

подпись, дата

доцент, канд.техн.наук

должность, ученая степень

А.И. Постников

инициалы, фамилия

Выпускник

подпись, дата

М. А. Совков

инициалы, фамилия

Нормоконтролер

подпись, дата

доцент, канд.техн.наук

должность, ученая степень

В.И. Иванов

инициалы, фамилия

Красноярск 2018