

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

Космических и информационных технологий

институт

Вычислительная техника

кафедра

УТВЕРЖДАЮ

Заведующий кафедрой

О.В. Непомнящий

подпись

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 «Аксиомы и свойства алгебры логики» 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

А.Н. Анисимова

инициалы, фамилия

Нормоконтролер

подпись,
дата

доцент, канд.техн.наук

должность, ученая степень

В.И. Иванов

инициалы, фамилия

Красноярск 2018

СОДЕРЖАНИЕ

Введение.....	3
1 Задание на выпускную квалификационную работу.....	4
2 Анализ технического задания на выпускную квалификационную работу	5
2.1 Анализ предметной области.....	5
2.2 Анализ задания на выпускную квалификационную работу.....	9
2.3 Выбор программных средств.....	10
2.3.1 Выбор языка программирования.....	10
2.3.2 Выбор среды программирования.....	11
2.3.3 Выбор операционной системы.....	11
3 Структура программного комплекса.....	12
3.1 Программа для подготовки базы данных.....	14
3.2 Программа для проверки результатов практической работы.....	27
3.2.1 Вкладка выбора варианта.....	32
3.2.2 Вкладка «Часть 1».....	34
3.2.3 Вкладка «Часть 2».....	37
4 Формирование отчета.....	39
5 Руководство пользователя.....	44
5.1 Программа для подготовки базы данных.....	44
5.2 Программа для проверки результатов практической работы.....	51
Заключение.....	62
Список сокращений.....	63
Список использованных источников.....	64
ПРИЛОЖЕНИЕ А.....	65

ВВЕДЕНИЕ

На сегодняшний день уровень развития высших учебных заведений возрастает вместе с количеством обучающихся в них студентах, это, конечно же, влияет на преподавателей в плане учебной нагрузке. В итоге процесс обучения усложняется, традиционные методы обучения уже не так эффективны, следовательно, это влияет на качество обучения.

Именно автоматизация отдельных функций преподавателя может способствовать уменьшению его нагрузки. Например, предоставить преподавателю возможность автоматической проверки практической работы студента, в результате не будет сомнений, что работа выполнена без ошибок и нагрузка на преподавателя будет снижена.

Темой выпускной квалификационной работы является «Программный комплекс для автоматизации проверки выполнения практического задания «Аксиомы и свойства алгебры логики» по дисциплине Прикладная теория цифровых автоматов».

Актуальность данной работы состоит в:

- Улучшении качества обучения студентов
- Упрощении проверки работ студентов

Таким образом, целью данной ВКР является разработка программного комплекса для автоматизации проверки выполнения практического задания по дисциплине «Прикладная теория цифровых автоматов» (ПТЦА).

1 Задание на выпускную квалификационную работу

Разработать программный комплекс, позволяющий автоматизировать процесс проверки результатов выполнения практической работы «Аксиомы и свойства алгебры логики» по дисциплине «Прикладная теория цифровых автоматов».

Программный комплекс должен состоять из двух программ:

1. Программа для подготовки базы вариантов заданий.
2. Программа для проверки правильности результатов, выполненных студентами практических работ.

Исходные данные:

- Аксиомы, свойства и законы алгебры логики;
- Число переменных, заданных функций алгебры логики – до 6;

Программа для подготовки базы вариантов заданий должна обеспечивать:

- Занесение в базу вариантов информации;
- Удаление информации;
- Коррекция занесённой информации;
- Шифрование информации при записи в базу;
- Сохранение базы вариантов;
- Число вариантов – до 60.

Программа для проверки правильности результатов, выполненных студентами практических работ должна обеспечивать проверку выполнения задания, состоящего из двух частей:

- Выбор варианта по его номеру;

Часть 1:

- Вывод на экран дисплея функции алгебры логики, которую нужно упростить, пользуясь аксиомами, свойствами и законами алгебры логики;
- Вывод на экран дисплея ФАЛ, которую нужно упростить, пользуясь аксиомами, свойствами и законами алгебры логики;

- Ввод полученного результата;
- Сравнение введенного вычисленного результата с правильным результатом, хранящимся в базе вариантов;
- Вывод сообщения о правильности/неправильности введенной информации.

Часть 2:

- Вывод на экран дисплея ФАЛ, которую нужно преобразовать, пользуясь правилом де Моргана;
- Ввод полученного результата;
- Сравнение введенного вычисленного результата с правильным результатом, полученным путём подстановки всех возможных комбинаций переменных в исходную и вычисленную ФАЛ;
- Вывод сообщения о правильности/неправильности введенной информации.

Отчет:

Должно производиться автоматизированное формирование файла отчёта по практической работе. Отчёт должен быть составлен с обязательным учётом требований стандарта СФУ "sfu-sto-4.2-07-2014.doc".

2 Анализ технического задания на выпускную квалификационную работу

2.1. Анализ предметной области

Высказывание является ключевым понятием алгебры логики. Высказывание – это предложение, которое может быть истинным или ложным [3]. Любое высказывание обозначается символом x , считается, что $x = 1$, если высказывание является истинным, а $x = 0$, если высказывание является ложным. Логическая (булева) переменная – это величина x , которая

принимает только два значения (1 или 0). Высказывание абсолютно истинно, если соответствующая ей логическая переменная принимает значение $x = 1$ в любом случае. Высказывание абсолютно ложно, если соответствующая ей логическая переменная принимает значение $x = 0$ в любом случае. Логическая функция (функция алгебры логики, ФАЛ) – это функция $f(x_1, x_2, \dots, x_n)$, которая принимает значение 0 или 1 на наборе логических переменных $x_1, x_2, x_3, \dots, x_n$. На рисунке 1 представлена таблица, содержащая логические функции от одной переменной [3].

x	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$
0	1	0	0	1
1	1	0	1	0

Рисунок 1 – Таблица логических функций от одной переменной

Функция $f_1(x)$ является истинной; функция $f_2(x)$ – ложная. Функция $f_3(x)$ – тождественная функция, функция $f_4(x)$ является функцией логического отрицания.

На рисунке 2 изображена таблица, содержащая 5 часто употребляемых логических функций от двух переменных.

x_1	x_2	$f_1(x_1, x_2)$	$f_2(x_1, x_2)$	$f_3(x_1, x_2)$	$f_4(x_1, x_2)$	$f_5(x_1, x_2)$
0	0	0	0	1	1	0
0	1	1	0	1	0	1
1	0	1	0	1	0	1
1	1	1	1	0	0	0

Рисунок 2 – Таблица, содержащая 5 часто употребляемых логических функций от двух переменных

Функция $f_1(x_1, x_2)$ называется функцией дизъюнкции (функция логического сложения), она истинна тогда, когда истинны или x_1 , или x_2 , или обе переменные. Алгебраическое обозначение функции:

Функция $f_2(x_1, x_2)$ – функция конъюнкции (функция логического умножения), она истинна только тогда, когда истинны x_1 и x_2 . Алгебраическое обозначение функции

На рисунке 3 представлены аксиомы алгебры логики

$$\begin{array}{lll}
 1. \ x = \overline{\overline{x}} & \text{закон исключения двойного отрицания.} & \\
 2. \ x + 0 = x & 3. \ x \cdot 0 = 0 & 4. \ x \oplus 0 = x \\
 \quad x + 1 = \overline{x} & \quad x \cdot 1 = x & \quad x \oplus 1 = \overline{x} \\
 \quad x + x = x & \quad x \cdot x = x & \quad x \oplus x = 0 \\
 \quad x + \overline{x} = 1 & \quad x \cdot \overline{x} = 0 & \quad x \oplus \overline{x} = 1
 \end{array}$$

Рисунок 3 – Аксиомы алгебры логики

Дизъюнкция и конъюнкция обладают свойствами, которые аналогичны свойствам обычных арифметических операций сложения и умножения.

Свойство ассоциативности (сочетательный закон):

$$x_1 + (x_2 + x_3) = (x_1 + x_2) + x_3;$$

$$x_1 \cdot (x_2 \cdot x_3) = (x_1 \cdot x_2) \cdot x_3;$$

Свойство коммутативности (переместительный закон):

$$x_1 + x_2 = x_2 + x_1;$$

$$x_1 \cdot x_2 = x_2 \cdot x_1;$$

Свойство дистрибутивности (распределительный закон):

$$x_1 \cdot (x_2 + x_3) = x_1 \cdot x_2 + x_1 \cdot x_3;$$

$$x_1 + x_2 \cdot x_3 = (x_1 + x_2) \cdot (x_1 + x_3);$$

На рисунке 4 представлены правила "склеивания":

$$x_1 \cdot x_2 + x_1 \cdot \overline{x_2} = x_1; \quad (x_1 + x_2) \cdot (x_1 + \overline{x_2}) = x_1.$$

Рисунок 4 – Правила «склеивания»

Существуют законы де Моргана, с их помощью можно выразить конъюнкцию через дизъюнкцию и отрицание или дизъюнкцию через конъюнкцию и отрицание [1]. На рисунке 5 представлены данные законы.

$$\overline{x1 \cdot x2} = \overline{x1} + \overline{x2},$$

$$\overline{x1 + x2} = \overline{x1} \cdot \overline{x2}.$$

Рисунок 5 – Законы де Моргана

На рисунке 6 представлены следствия из законов де Моргана:

$$x1 \cdot x2 = \overline{\overline{x1} + \overline{x2}},$$

$$x1 + x2 = \overline{\overline{x1} \cdot \overline{x2}}.$$

Рисунок 6 – Следствия из законов де Моргана

Законы де Моргана справедливы для любого числа переменных. На рисунке 7 представлены законы поглощения.

$$x1 + (x1 \cdot x2) = x1 ;$$

$$x1 \cdot (x1 + x2) = x1 .$$

Рисунок 7 – Законы поглощения

На рисунке 8 представлены законы обобщенного поглощения.

$$x1 \cdot x2 + \overline{x1} \cdot x3 + x2 \cdot x3 = x1 \cdot x2 + \overline{x1} \cdot x3 ;$$

$$(x1 + x2) \cdot (\overline{x1} + x3) \cdot (x2 + x3) = (x1 + x2) \cdot (\overline{x1} + x3) .$$

Рисунок 8 – Законы обобщенного поглощения

Кроме того, существует необходимый порядок выполнения логических операций:

- 1) Инверсия отдельных переменных.
- 2) Операции в скобках и операции со знаком отрицания.
- 3) Конъюнкция.
- 4) Дизъюнкция.

2.2 Анализ задания на выпускную квалификационную работу

По заданию на выпускную квалификационную работу (ВКР) должен быть разработан программный комплекс для автоматизации проверки практической работы «Аксиомы и свойства алгебры логики». Целью данной практической работы является следующее: научиться пользоваться аксиомами законами и свойствами алгебры логики для преобразования логических функций [4].

Практическая работа состоит из двух частей. Задание для 1-ой части следующее: пользуясь аксиомами и свойствами алгебры логики, упростить выражение.

На рисунке 9 представлен один из вариантов для первой части практической работы.

$$F = \overline{x_1 \cdot x_2 \cdot x_4} + \overline{x_1 \cdot x_2} + \overline{x_2 \cdot x_3 \cdot x_4} + \overline{x_1 \cdot x_2 \cdot x_3} + \overline{x_1 \cdot x_2} + (\overline{x_2} + \overline{x_1 \cdot x_2 \cdot x_3}) \cdot (\overline{x_2 \cdot x_3} + \overline{x_2 \cdot x_3 \cdot x_4} + \overline{x_2 \cdot x_3} + \overline{x_2 \cdot x_3 \cdot x_4}) + \overline{x_1 \cdot x_2 \cdot x_3} + \overline{x_1 \cdot x_3 \cdot x_4} + \overline{x_1 \cdot x_3}.$$

Рисунок 9 – Вариант №1 для первой части практической работы

Задание для 2-ой части следующее: в заданной ФАЛ с помощью правила де Моргана заменить знак конъюнкции на знак дизъюнкции, а знак дизъюнкции – на знак конъюнкции.

На рисунке 10 представлен один из вариантов для второй части практической работы.

$$F = x^4 + \overline{x1} \cdot x2 + x1 \cdot \overline{x2} \cdot x3 + \overline{x1} \cdot \overline{x2} \cdot \overline{x3} \cdot x4 .$$

Рисунок 10 – Вариант №1 для второй части практической работы

Автоматизация проверки практической работы состоит в следующем:

Для 1-ой части: преподаватель заполняет с помощью первой программы базу данных ответами для вариантов задания. Студент во второй программе заполняет соответствующее поле окончательным ответом на задание. Если ответ студента и ответ из базы данных совпадают, то задание считается выполненным верно. В качестве примера на рисунке 11 изображен один из возможных ответов на данное задание.

$$\text{Ответ: } F = x5 .$$

Рисунок 11 – Ответ для варианта №1 первой части практической работы

Для 2-ой части: в этом случае база данных уже не заполняется преподавателем, ответ на вариант задания вычисляется автоматически в программе, предназначенной для студента. Студент также вводит свой ответ в соответствующее поле. В качестве примера на рисунке 12 изображен один из возможных ответов на данное задание.

$$\text{Ответ: } F = (\overline{x1} + x2) \cdot (\overline{x1} + x3) \cdot (\overline{x2} + x4 + x5) .$$

Рисунок 12 – Ответ для варианта №1 второй части практической работы

2.3 Выбор программных средств

2.3.1 Выбор языка программирования

Исходя из технического задания, решено, что данный программный комплекс будет реализован на языке C#. Используются различные технологии для создания графического интерфейса с помощью .NET, например, Windows Forms, WPF и другие [5]. Однако основываясь на личных предпочтениях и имеющемся опыте, была выбрана платформа Windows Forms.

2.3.2 Выбор среды программирования

Возможные среды для программирования на C#: MS Visual Studio, SharpDevelop, MonoDevelop. Основываясь на личных предпочтениях и возможностях среды, была выбрана MS Visual Studio. Visual Studio используется как стартовая платформа для программирования, для коммерческих и любительских разработок [2].

2.3.3 Выбор операционной системы

Выбор операционной системы не был произведен, так как исходя из требований задания на ВКР, разрабатываемый комплекс должен использоваться в компьютерных классах Института космических и информационных технологий Сибирского Федерального Университета, а в

них установлены Windows 7, Windows 10. Поэтому данный программный комплекс будет разработан на операционной системе Windows.

3 Структура программного комплекса

Программный комплекс по заданию состоит из двух программ. Одна программа является базой данных, в которую преподаватель заносит задания на практическую работу, ответы на первую часть практической работы, а другая программа используется студентом для проверки правильности полученных ответов в ходе решения практической работы.

На рисунке 13 представлена структурная схема для программного комплекса «Автоматизация проверки выполнения практического задания «Аксиомы и свойства алгебры логики» по дисциплине ПТЦА».



Рисунок 13 – Структурная схема программного комплекса

Структурная схема разрабатываемого программного комплекса состоит из следующих блоков:

- Программы для подготовки базы данных (Программа для преподавателя);
- Базы данных;
- Программы для проверки результатов практической работы (Программа для студента);
- Отчета.

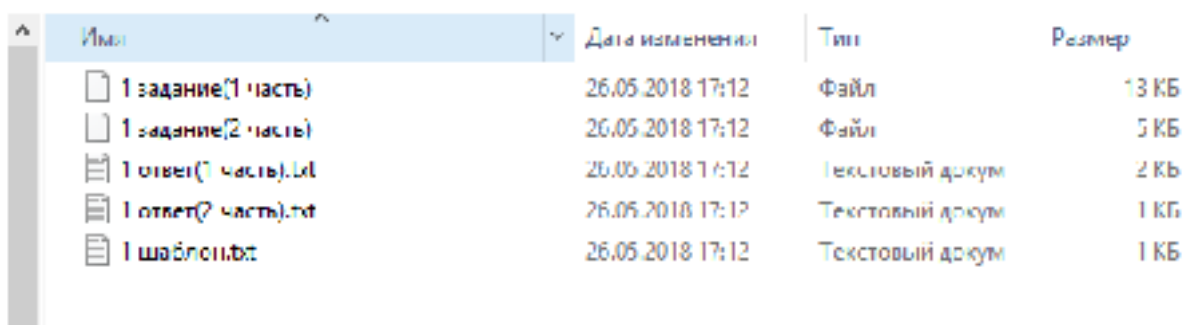
Далее эти блоки будут рассмотрены подробнее.

1. Программы для подготовки базы данных (Программа для преподавателя):

Данный блок содержит программу, которая используется только преподавателем в целях заполнения базы данных соответствующими ответами и заданиями. Программа служит для заполнения базы ответами только для первого задания работы. Также в базу заносятся задания для всей практической работы. У преподавателя будет возможность вносить, удалять, изменять информацию. Имеющиеся ответы будут храниться в зашифрованном виде.

2. База данных:

База данных содержит в себе зашифрованные ответы для 1-ой части практической работы, задания для обеих частей практической работы. База данных представляет из себя папку, в которой структурированно находятся: файлы с ответами для первой части работы, файлы с заданием для первой части работы, файлы с заданием для второй части работы, файлы с шаблонами для второй части работы. Файлы с шаблонами для второй части работы используются программой для проверки результатов практической работы следующим образом: введенный преподавателем шаблон отображается в программе для студента, который используется для ввода ответа для второй части работы. На рисунке 14 изображен внешний вид папки, содержащей базу данных для данного программного комплекса.



Имя	Дата изменения	Тип	Размер
1 задание(1 часть)	26.05.2018 17:12	Файл	13 КБ
1 задание(2 часть)	26.05.2018 17:12	Файл	5 КБ
1 ответ(1 часть).txt	26.05.2018 17:12	Текстовый докум	2 КБ
1 ответ(2 часть).txt	26.05.2018 17:12	Текстовый докум	1 КБ
1 шаблон.txt	26.05.2018 17:12	Текстовый докум	1 КБ

Рисунок 14 – База данных

На рисунке 15 представлено содержание шаблона для одного из вариантов задания для второй части работы.

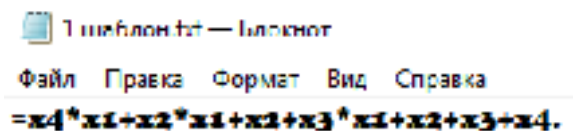


Рисунок 15 – Шаблон для второй части практической работы

3. Программы для проверки результатов практической работы (Программа для студента)

Данный блок содержит программу, которая используется студентами для проверки правильности введенных ответов на практическую работу. Программа предполагает две части задания: первая часть — минимизация ФАЛ, вторая часть — использование законов де Моргана. Данная программа использует файлы из базы данных для загрузки задания на две части практической работы, а также для загрузки шаблона выполнения второй части работы.

4. Отчет

Данный блок является отчетом, сформированным автоматически после успешного выполнения всей практической работы. Отчет должен состоять из титульного листа, задания и ответов студента.

3.1 Программа для подготовки базы данных

Исходя из требований задания на ВКР данная программа должна обеспечивать:

- занесение в базу данных ответов на первую часть практической работы;
- занесение в базу данных вариантов для практической работы;
- удаление информации;
- коррекция занесённой информации;
- шифрование информации при записи в базу;

- сохранение базы вариантов;
- число вариантов – до 60.

Данная программа состоит из трех действующих форм: Program, Form1, Form2. Program – основная форма. Form1 представляет собой окно, где имеются вкладки с двумя частями практической работы. Form2 – главное окно программы. На рисунке 16 представлены методы формы Program.

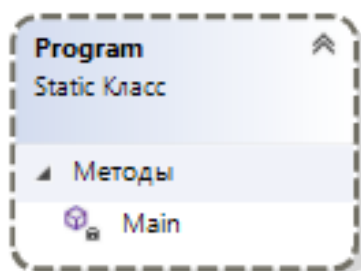


Рисунок 16 – Методы формы Program

Данная форма предназначена только для того, чтобы указать форму, которая будет в первую очередь. Метод Main – указывает на то, что при запуске программы будет открываться всегда сперва Form2.

На рисунке 17 представлены методы формы Form1.

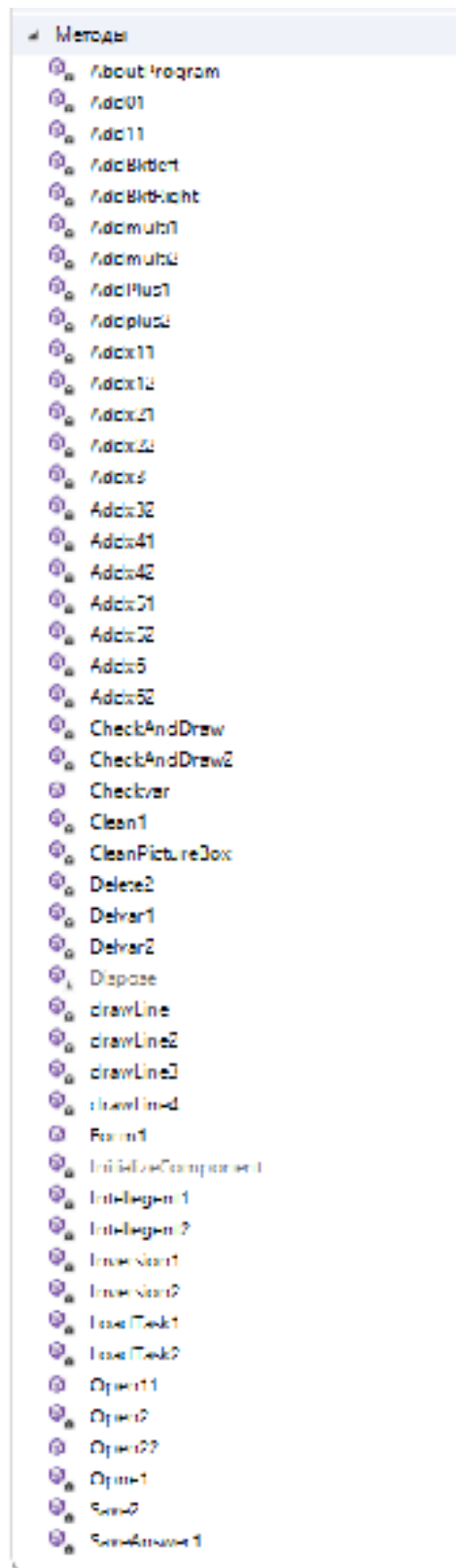


Рисунок 17 - Методы Form1

Метод AboutProgram – выводит ознакомительную информацию о программе;

Метод Add01 – выводит символ «0» в поле для ввода ответа первой части работы;

Метод Add11 – выводит символ «1» в поле для ввода ответа первой части работы;

Метод AddBrkleft – выводит символ «(» в поле для ввода задания второй части работы;

Метод AddBktRight - выводит символ «)» в поле для ввода задания второй части работы;

Метод Addmulti1 – выводит символ «*» в поле для ввода ответа первой части работы;

Метод Addmulti2 – выводит символ «*» в поле для ввода задания второй части работы;

Метод AddPlus1 – выводит символ «+» в поле для ввода ответа первой части работы;

Метод Addplus2 – выводит символ «+» в поле для ввода задания второй части работы;

Метод Addx11 – выводит строку «x1» в поле для ввода ответа первой части работы;

Метод Addx12 – выводит строку «x1» в поле для ввода задания второй части работы;

Метод Addx21 – выводит строку «x2» в поле для ввода задания второй части работы;

Метод Addx22 – выводит строку «x2» в поле для ввода задания второй части работы;

Метод Addx3 – выводит строку «x3» в поле для ввода ответа первой части работы;

Метод Addx32 – выводит строку «x3» в поле для ввода задания второй части работы;

Метод Addx41 – выводит строку «x4» в поле для ввода ответа первой части работы;

Метод Addx42 – выводит строку «x4» в поле для ввода задания второй части работы;

Метод Addx51 – выводит строку «x5» в поле для ввода ответа первой части работы;

Метод Addx52 – выводит строку «x5» в поле для ввода задания второй части работы;

Метод Addx6 – выводит строку «x6» в поле для ввода ответа первой части работы;

Метод Addx62 – выводит строку «x6» в поле для ввода задания второй части работы;

Метод CheckAndDraw – прорисовывает отрицание в первой части работы;

Метод CheckAndDraw2 – прорисовывает отрицание во второй части работы;

Метод Checkvar – проверяет введенный вариант в главном окне на корректность;

Метод Clean1 – очищает поле ввода для первой части работы;

Метод Delete2 – очищает поле ввода для второй части работы;

Метод Delvar1 – удаляет вариант первой части работы и его данные;

Метод Delvar2 – удаляет вариант второй части работы и его данные;

Метод drawLine – «прорисовывает» инверсию в ответе для первой части работы;

Метод drawLine2 – «прорисовывает» инверсию для второй части работы;

Метод Intellegent1 – проверяет корректность введенного ответа для первой части работы;

Метод Intellegent2 – проверяет корректность введенного ответа для второй части работы;

Метод Inversion1 – передает данные в метод drawLine для расстановки инверсии в первой части работы;

Метод `Inversion2` – передает данные в метод `drawLine2` для расстановки инверсии во второй части работы;

Метод `LoadTask1` – загружает изображение с заданием в базу данных для первой части работы;

Метод `LoadTask2` – загружает изображение с заданием в базу данных для второй части работы;

Метод `Open2` – открывает вариант второй части работы из базы данных;

Метод `Open1` – открывает вариант первой части работы из базы данных;

Метод `Save2` – сохраняет задание для второй части работы в базу данных;

Метод `SaveAnswer1` – сохраняет ответ и задание для первой части работы в базу данных.

На рисунке 18 представлены методы `Form2`.

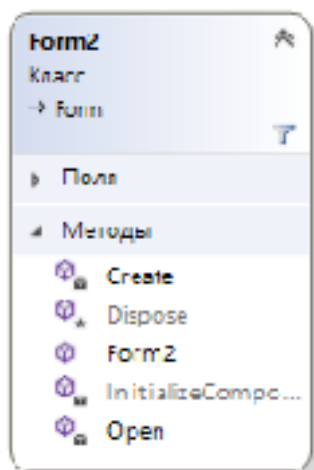


Рисунок 18 – Методы `Form2`

Метод `Create` – создает новый вариант в базе данных;

Метод `Open` – открывает имеющийся вариант в базе данных;

Метод `Form2` – инициализирует форму.

На рисунке 19 представлена обобщенная блок-схема работы программы для подготовки базы данных.

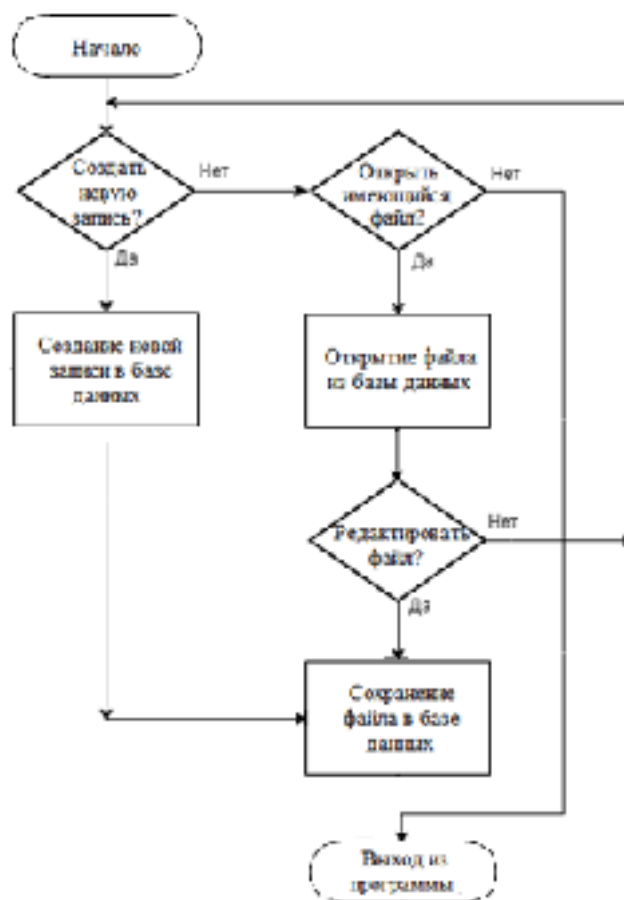


Рисунок 19 – Обобщенная блок-схема работы программы для подготовки базы данных

Программа работает следующим образом: перед преподавателем открывается главное окно, где он должен выбрать необходимое действие (создать ответ, открыть уже имеющийся). На рисунке 20 представлено главное окно данной программы.

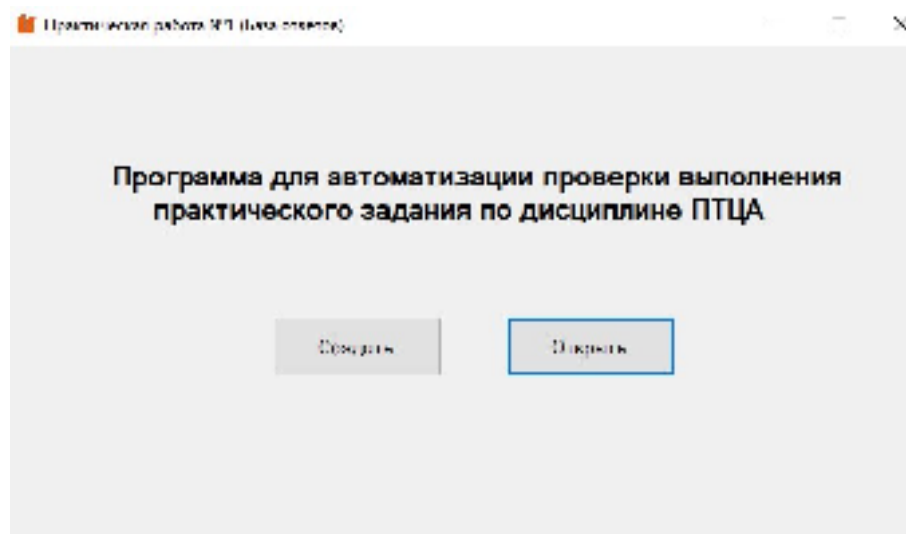


Рисунок 20 – Главное окно программы для подготовки базы данных

Далее открывается окно ввода ответа для выбранного варианта. На рисунке 21 изображено окно для работы с первой частью практической работы.

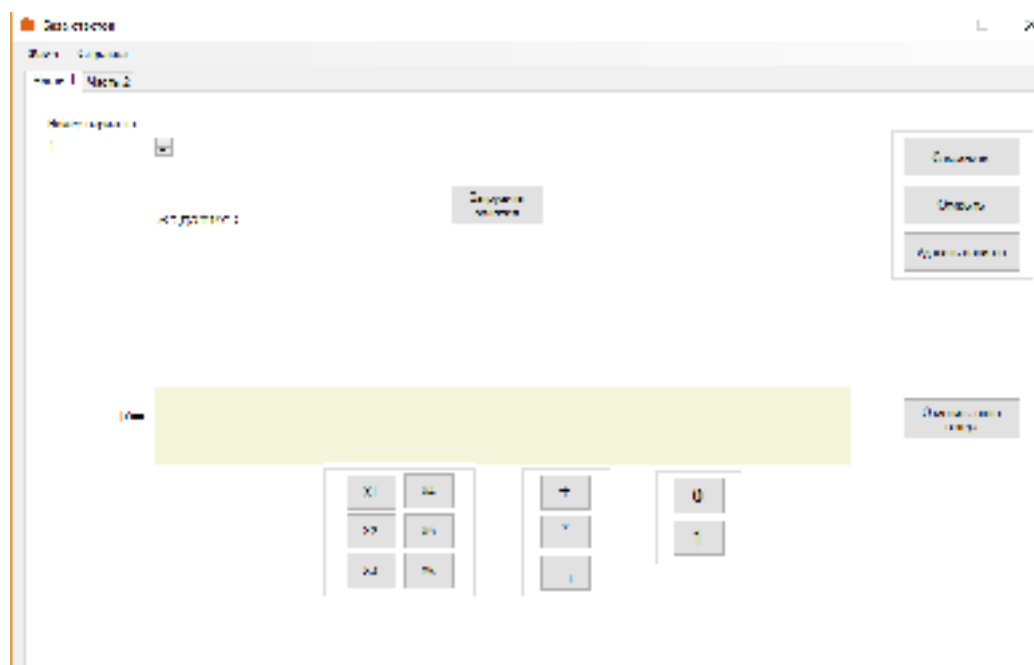


Рисунок 21 – Вкладка «Задание 1»

На рисунке 22 изображено окно для работы со второй частью практической работы.

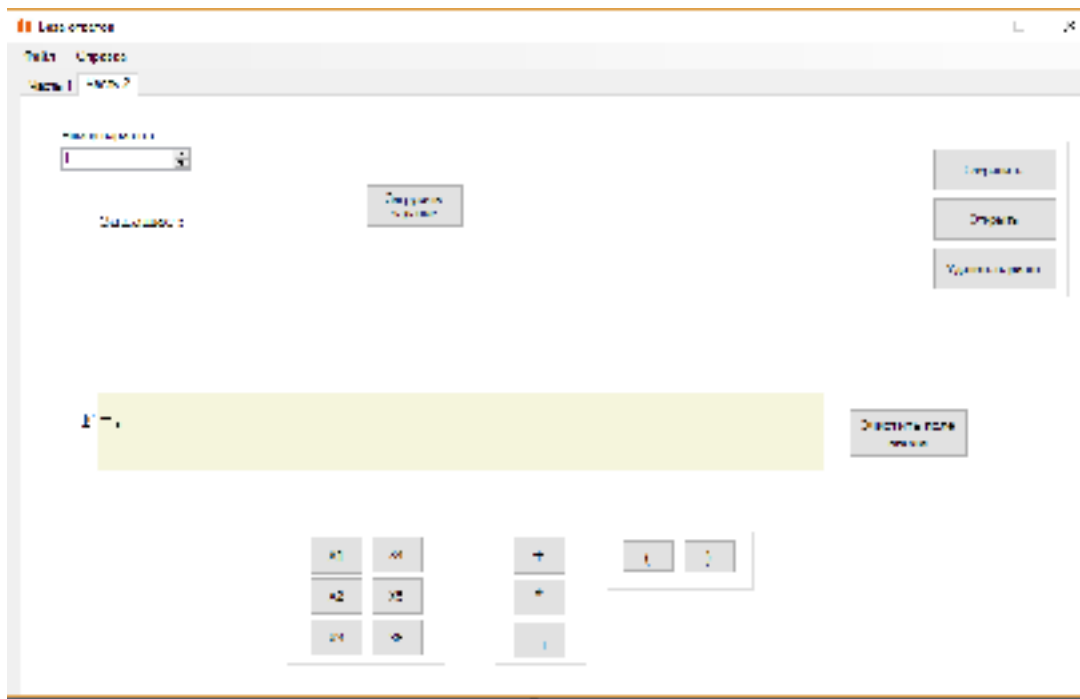


Рисунок 22 – Вкладка «Задание 1»

На рисунке 23 изображена загрузка задания и запись ответа для первой части работы.

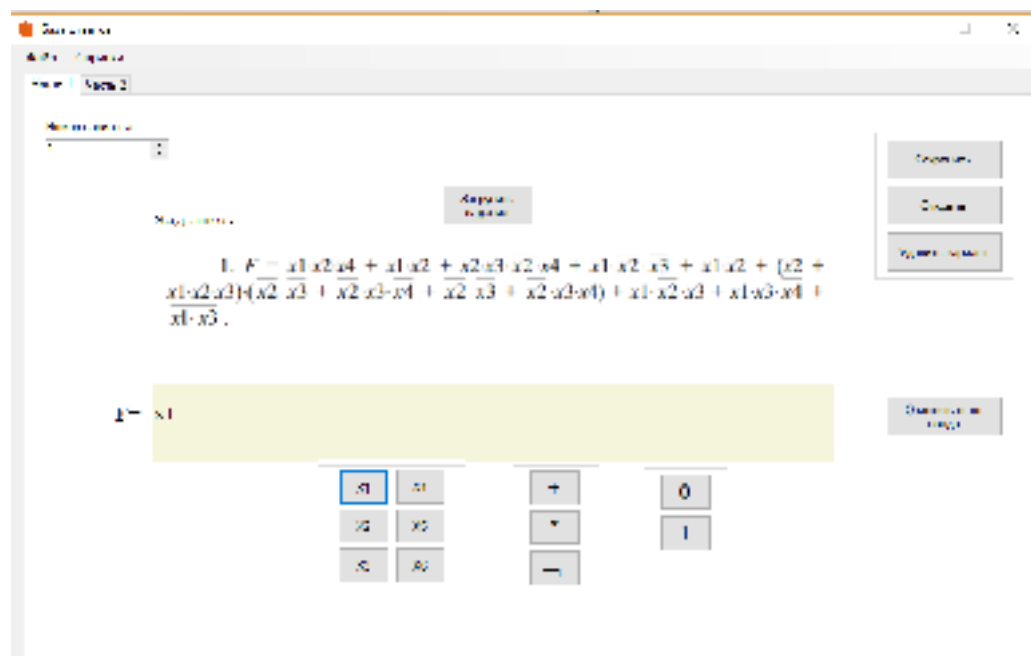


Рисунок 23 – Загрузка задания и запись ответа

На рисунке 24 изображена загрузка задания и запись шаблона для второй части работы.

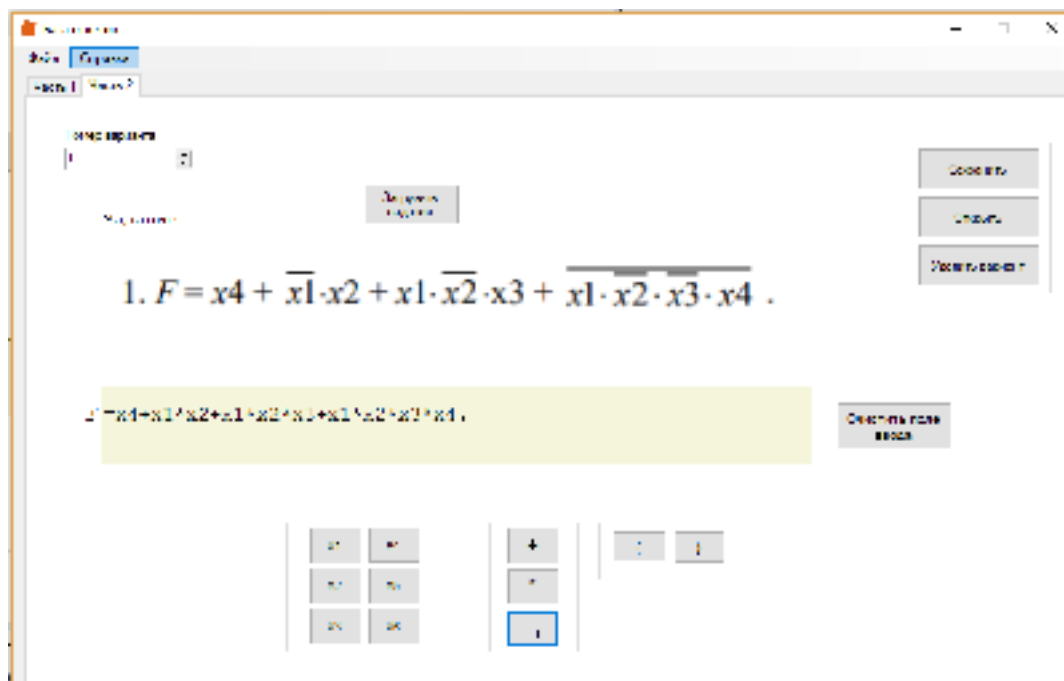


Рисунок 24 – Загрузка задания и запись ответа

При сохранении ответа появляется окно с подтверждением сохранения в базе данных (рисунок 25).

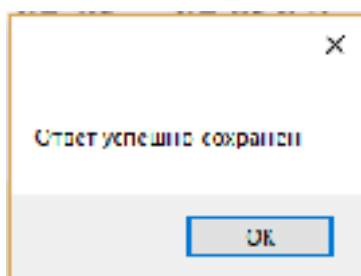


Рисунок 25 – Сохранение ответа в базе данных

При удалении ответа из базы также появляется окно с оповещением (рисунок 26).

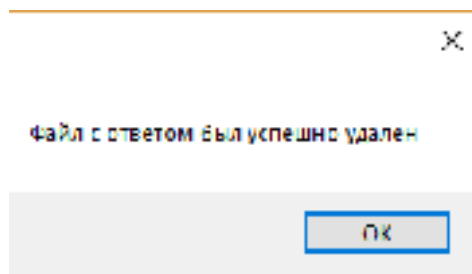


Рисунок 26 – Удаление ответа из базы данных

По заданию данные в базе ответов должны быть зашифрованы. На рисунке 27 изображен вид одного из файлов с вариантом в базе ответов.

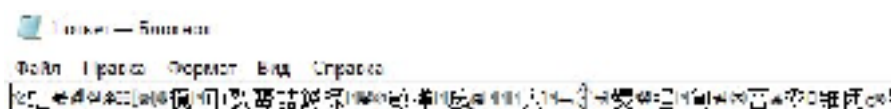


Рисунок 27 – Вид файла с ответом в базе данных

Шифрование данных происходит в момент сохранения ответа в базу данных, то есть при нажатии на кнопку «Сохранить» вызывается метод `SaveAnswer1()`, который как раз и выполняет шифрование. Дешифрование выполняется в момент открытия файла из базы данных, то есть при нажатии на кнопку «Открыть» вызывается метод `Open1()`, который как раз и выполняет дешифрование. Шифрование и дешифрование происходит с использованием класса `DesCryptoServiceProvider` (пространство `System.Security.Cryptography`). Данный вид шифрования является алгоритмом для симметричного шифрования, разработанным фирмой IBM и утверждённым правительством США в 1977 году как официальный стандарт. В основе шифрования лежит использование двух классов: `SymmetricAlgorithm.IV` (возвращает или задает вектор инициализации (IV) для симметричного алгоритма), `SymmetricAlgorithm.Key` (возвращает или задает секретный ключ для симметричного алгоритма). Но так как в данной работе шифрование не является главным в плане реализации, то достаточно просто сказать об используемом методе шифрования.

При создании данного программного комплекса был разработан метод, который позволяет ставить инверсию над выделенным элементом. Например, метод `Inversion1()` используется для расстановки инверсии в первой части практической работы. Прежде, чем перейти к описанию данного метода, стоит объяснить, поле для ввода ответа в программе состоит из следующих элементов: `richtextbox` и `picturebox`. В `richtextbox` содержится шаблон логического выражения (именно туда вставляются такие элементы, как « x_1 », « $*$ » и так далее). В `picturebox` уже отображаются имеющиеся инверсии над элементами. То есть по итогу `richtextbox` – это массив (строка) символьных элементов, например, значение `richtextbox` = “ $x_1+x_2*x_3$ ”. А в `picturebox` содержится двумерный массив `underline[8,100](level, i)`, который содержит в себе следующие значения: `level` – это сколько уровней отрицаний может быть в данной форме, `i` – это наличие отрицания над элементом (если `I` = 0, то отрицания над этим элементом нет, если = 1, то отрицание есть). Таким образом, каждый раз, когда нажимается кнопка отрицания в форме для выделенного элемента, происходит процесс записи соответствующих нулей и единиц в имеющийся массив `underline`.

На рисунке 28 представлена обобщенная блок-схема алгоритма расстановки инверсии.

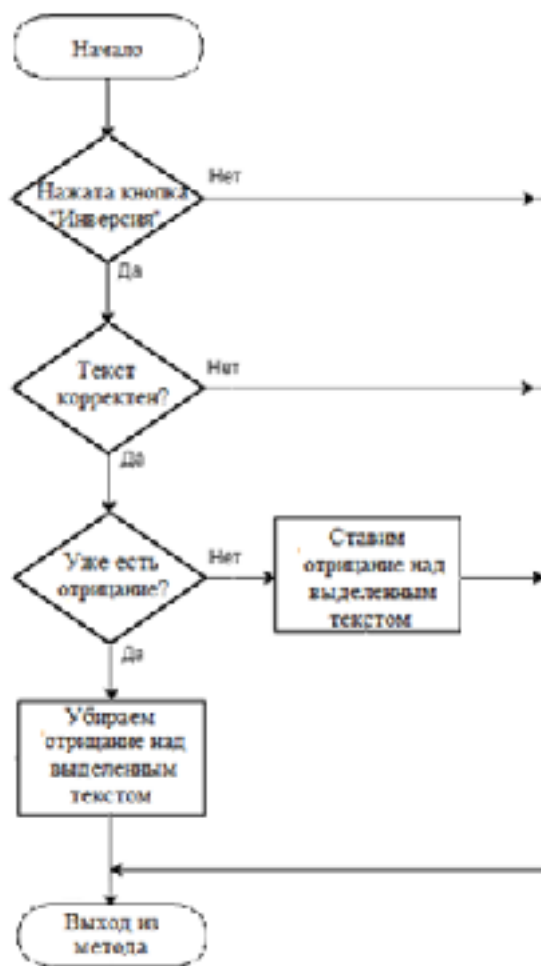


Рисунок 28 – Обобщенная блок-схема расстановки инверсии

Теперь более подробно о методе `Inversion1()`. Когда вызывается данный метод (после нажатия кнопки инверсии), с необходимыми параметрами из этого же метода вызывается другой метод – `DrawLine()`. В `DrawLine` определяется путем изучения массива `underline`, есть ли над выделенным текстом уже отрицание, если имеется, то отрицание над выделенным текстом должно исчезнуть (исходя из свойств алгебры логики, двойное отрицание должно исключаться). Удаление отрицания совершается путем вызова метода `CheckAndDraw()`. Этот метод в данном случае использует объект `White Pen` встроенного класса `Pen`, который фактически просто нарисует белую линию, что с нашей стороны будет выглядеть как удаление отрицания. Если над выделенным элементом нет отрицания, то по такому же принципу рисуется черная линия объектом `Black Pen` класса `Pen`.

3.2 Программа для проверки результатов практической работы

Исходя из требований задания на ВКР программа должна предоставить студенту возможность проверки правильности выполнения практической работы. Данная программа содержит следующие формы: Form1, WaitForm, FormaOtcheta. Form1 – основная форма для работы студента, WaitForm – форма, которая по сути является оповещающим окном о процессе проверки ответов студента, FormaOtcheta – форма, в которой студент вводит необходимые данные для формирования отчета.

На рисунке 29 представлены методы формы Form1.

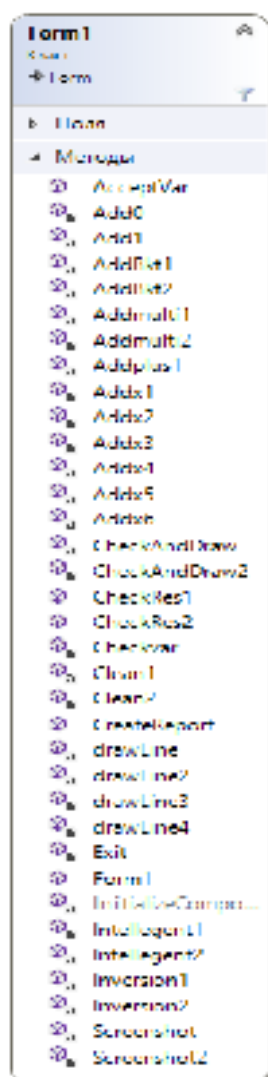


Рисунок 29 – Методы Form1

Метод `AcceptVar` – загружает задание и шаблон для ответа;

Метод `Add0` – вставляет в поле ввода ответа для первой части символ «0»;

Метод `Add1` – вставляет в поле ввода ответа для первой части символ «1»;

Метод `AddBkt1` – вставляет в поле ввода ответа для второй части символ «(»;

Метод `AddBkt2` – вставляет в поле ввода ответа для второй части символ «)»;

Метод `Addmulti1` – вставляет в поле ввода ответа для первой части символ «*»;

Метод `Addmulti2` – вставляет в поле ввода ответа для второй части символ «*»;

Метод `Addplus1` – вставляет в поле ввода ответа для первой части символ «+»;

Метод `Addx1` – вставляет в поле ввода ответа для первой части строку «x1»;

Метод `Addx2` – вставляет в поле ввода ответа для первой части строку «x2»;

Метод `Addx3` – вставляет в поле ввода ответа для первой части строку «x3»;

Метод `Addx4` – вставляет в поле ввода ответа для первой части строку «x4»;

Метод `Addx5` – вставляет в поле ввода ответа для первой части строку «x5»;

Метод `Addx6` – вставляет в поле ввода ответа для первой части строку «x6»;

Метод `CheckAndDraw` – прорисовывает отрицание над выделенным текстом для первой части работы;

Метод CheckAndDraw2 – прорисовывает отрицание над выделенным текстом для второй части работы;

Метод CheckRes1 – проверяет ответ для первой части работы;

Метод CheckRes2 – проверяет ответ для второй части работы;

Метод Checkvar – проверяет ввод варианта на корректность;

Метод Clean1 – очищает поле для ввода ответа первой части работы;

Метод Sclean2 – очищает поле для ввода ответа второй части работы;

Метод CreateReport – создает отчет, если CheckRes1 и CheckRes2 вернули значение «true»;

Метод drawline – проверяет наличие инверсии над выделенными элементами в первой части работы;

Метод drawLine2 – проверяет наличие инверсии над выделенными элементами во второй части работы;

Метод Exit – завершает выполнение программы;

Метод Intellegent1 – проверяет на корректность расстановки инверсии при вводе ответа для первой части работы;

Метод Intellegent2 – проверяет на корректность расстановки инверсии при вводе ответа для второй части работы;

Метод Inversion1 – передает необходимые параметры для прорисовки в drawLine;

Метод Inversion2 – передает необходимые параметры для прорисовки в drawLine2;

Метод Screenshot – сохраняет ответ на первую часть работы в виде изображения;

Метод Screenshot2 – сохраняет ответ на вторую часть работы в виде изображения;

После того, как студент введет свой ответ для второй части практической работы в поле ввода, в связи с тем, что процесс проверки ответа занимает несколько секунд, появляется окошко с просьбой подождать, пока выполнится процесс (рисунок 30).

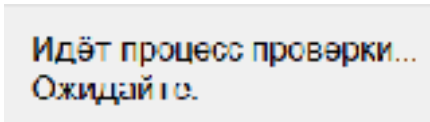


Рисунок 30 – Окно WaitForm

На рисунке 31 представлены методы формы WaitForm.

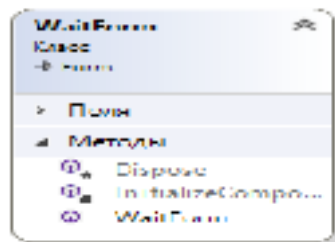


Рисунок 31 – Метода формы WaitForm

Метод WaitForm – инициализирует форму, выводит сообщение «Идет процесс проверки, ожидайте».

После того, как практическая работа будет полностью выполнена верна (и часть 1, и часть 2), появится форма с необходимыми данными для формирования отчета (рисунок 32).

A screenshot of a Windows application window titled "Создание отчета". The window contains five text input fields with labels: "ФИО студента:", "Номер группы:", "Номер зак. книжки.", "ФИО преподавателя:", and "Номер практической работы:". Below the fields is a button labeled "Принять".

Рисунок 32 – Форма для создания отчета

На рисунке 33 представлены методы FormaOtcheta.

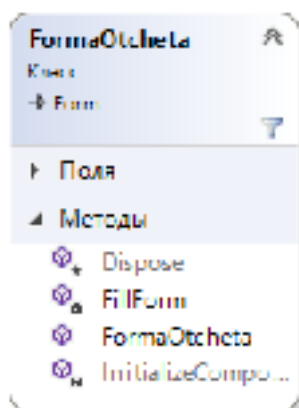


Рисунок 33 – Методы формы FormaOtcheta

Метод FillForm – заполняет необходимые закладки в docx-документе;

Метод FirmaOtcheta – инициализирует форму, проверяет корректность введенных данных;

На рисунке 34 изображена обобщенная блок-схема работы данной программы.



Рисунок 34 – Обобщенная блок-схема программы для проверки результатов выполнения практической работы

Перед студентом изначально имеется главное окно, где он должен выбрать свой номер варианта. После этого появляются две дополнительные вкладки, в которых содержатся две части задания для выполнения. Затем студент должен ввести свои ответы в соответствующие поля, если они верны, то автоматически формируется отчет.

3.2.1 Вкладка «Выбор варианта»

На рисунке 35 изображена блок-схема функционирования вкладки «Выбор варианта».

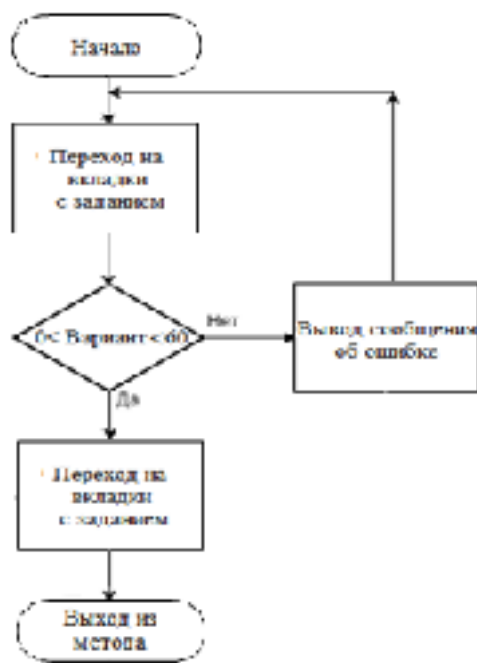


Рисунок 35 – Блок-схема функционирования вкладки «Выбор варианта»

На рисунке 36 изображено главное окно программы для проверки результатов практической работы.

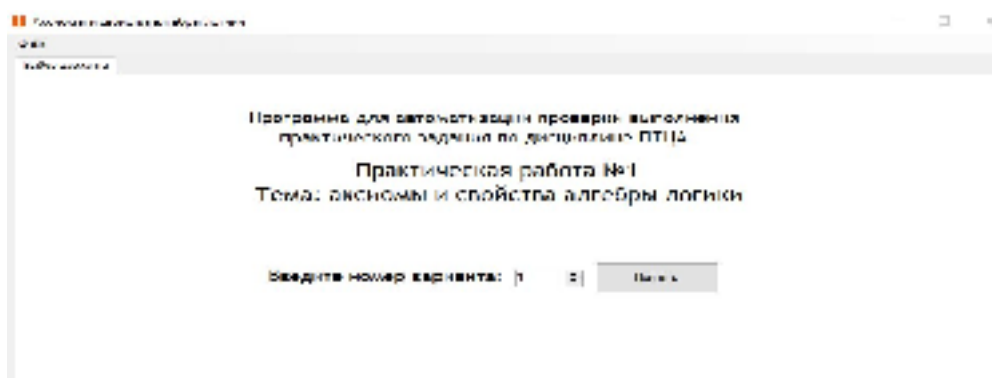


Рисунок 36 – Главное окно программы для проверки результатов практической работы

Для начала студент должен ввести свой номер варианта в соответствующее поле. По заданию на ВКР число вариантов ограничено (всего 60 вариантов). Если студент введет число больше 60 или меньше 1, то программа выдаст следующее окно с ошибкой, представленное на рисунке 37.

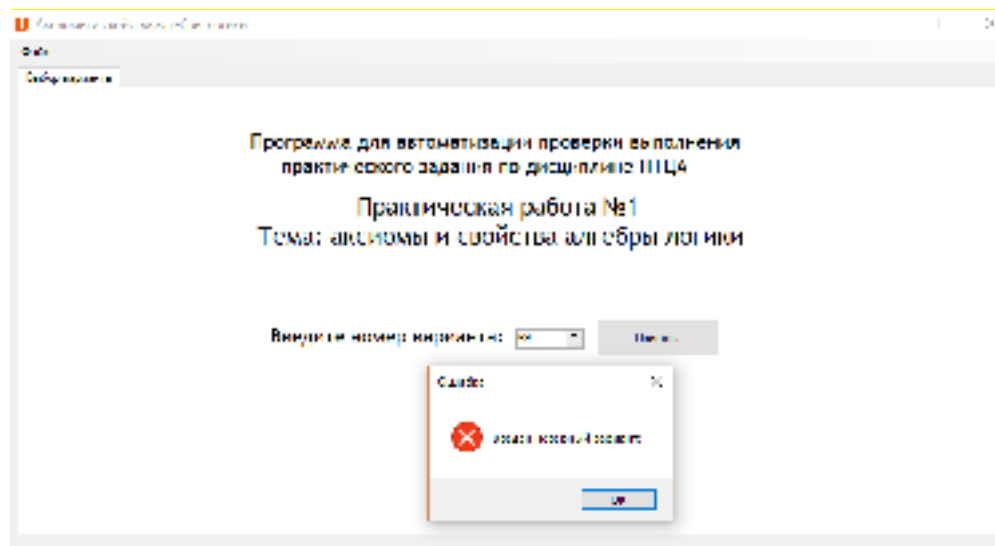


Рисунок 37 – Ошибка «Введен неверный вариант»

Помимо этого, для уменьшения возможных ошибок при вводе был разрешен ввод только цифр (т.е. буквы и любые другие символы в окно для ввода варианта студент не сможет напечатать).

Если номер варианта корректный, то автоматически перед студентом появляются две вкладки, содержащие задания на практическую работу. Приступить к выполнению заданию можно с любой его части. Данные вкладки будут рассмотрены далее.

3.2.2 Вкладка «Часть 1»

На рисунке 38 представлена обобщенная блок-схема функционирования вкладки «Часть 1».

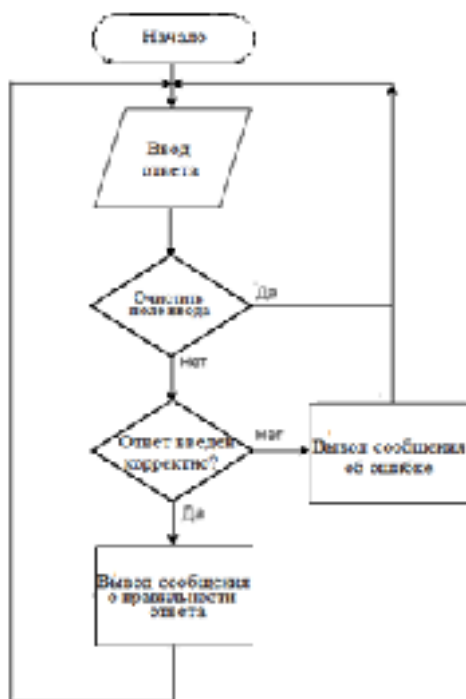


Рисунок 38 – Обобщенная блок-схема вкладки «Часть 1»

На рисунке 39 изображена вкладка, содержащая первую часть задания практической работы.

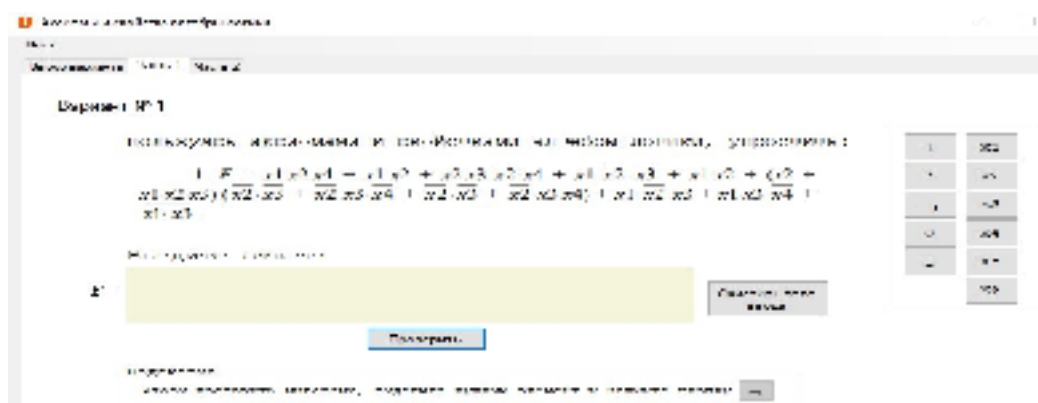


Рисунок 39 – Вкладка «Часть 1»

После того, как студент упростит имеющееся выражение, он должен ввести полученный ответ в предназначенное для этого поле. Студент может расставлять знак инверсии, арифметические знаки, переменные, 0 и 1. Если студент ошибся, то он всегда может нажать кнопку «Очистить поле ввода» и попробовать заново. Также имеется подсказка о том, как расставить инверсию.

Чтобы проверить правильность введенного ответа, необходимо нажать кнопку «Проверить». Если ответ оказался неверным, появляется сообщение об этом. На рисунке 40 изображен данный случай.

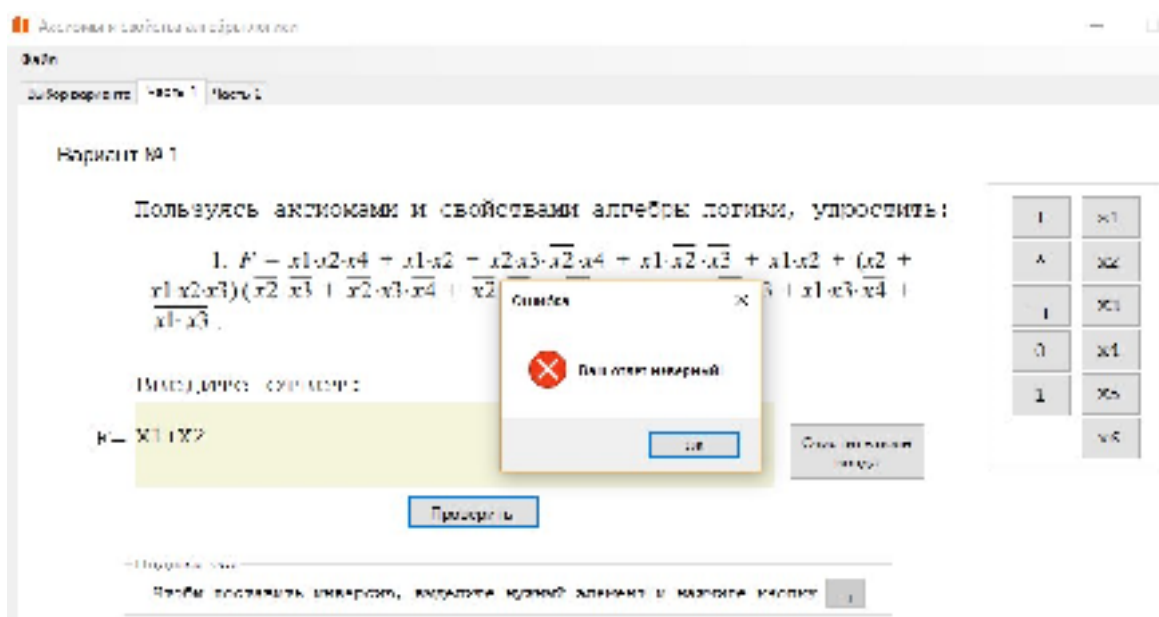


Рисунок 40 – Сообщение о неправильном ответе

Если же ответ оказался верным, то появляется сообщение о подтверждении. На рисунке 41 изображен данный случай.

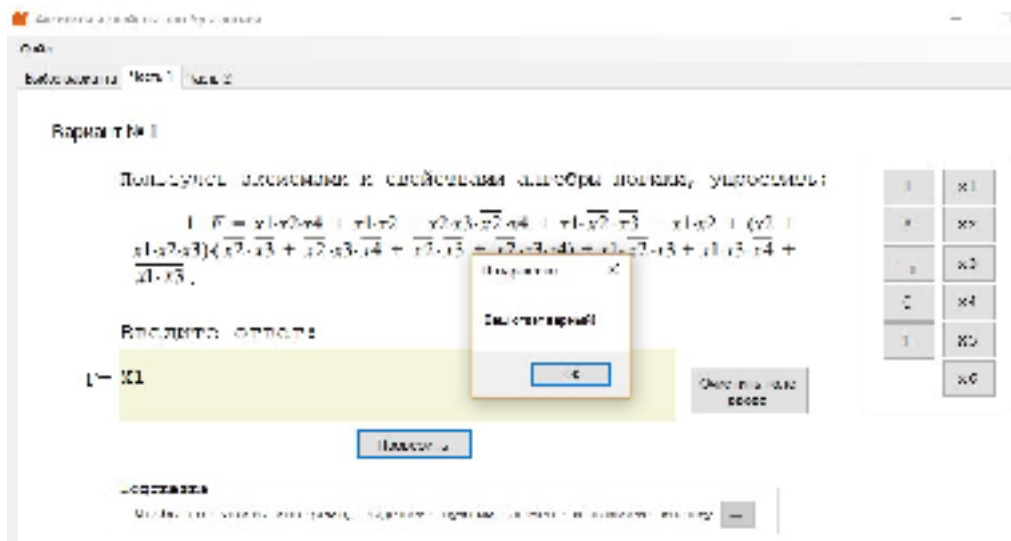


Рисунок 41 – Сообщение о правильном ответе

Полученный ответ студента записывается в отдельный файл. После этого из базы данных берется файл, содержащий ответ на выполняемый вариант. Затем эти файлы сравниваются, если они совпадают, то задание считается выполненным верно.

3.2.3 Вкладка «Часть 2»

На рисунке 42 изображена вкладка, содержащая вторую часть задания.

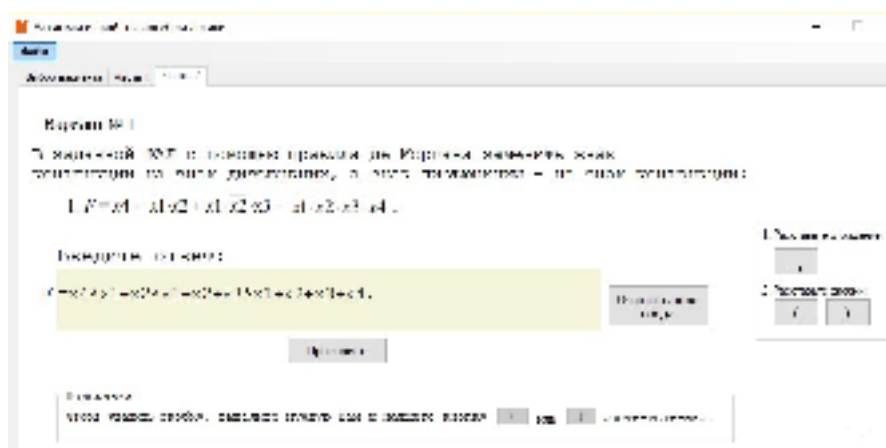


Рисунок 42 – Вкладка «Часть 2»

На рисунке 43 представлена обобщенная блок-схема функционирования вкладки «Часть 2».

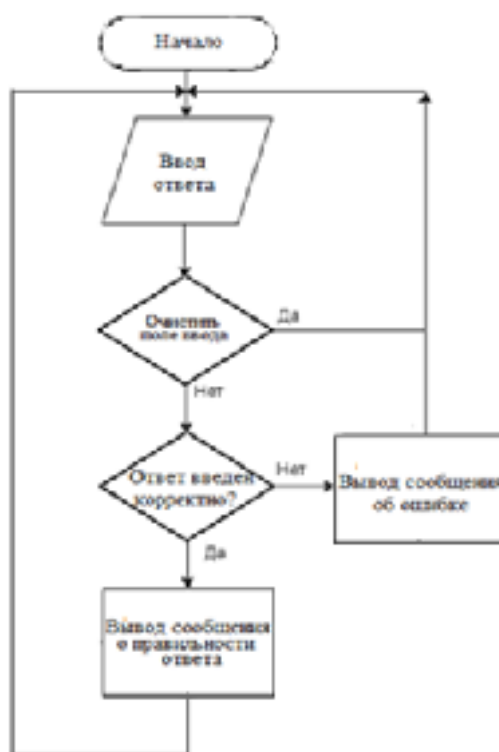


Рисунок 43 – Обобщенная блок-схема функционирования вкладки «Часть 2»

В этой части задания уже имеется готовый шаблон ответа для удобства студента. Ему остается расставить только скобки и знаки инверсии, где это необходимо. Если студент хочет начать заполнение шаблона заново, то для этого имеется кнопка «Очистить поле ввода». Также имеется подсказка о том, как удалить ненужную скобку. Для проверки ответа существует кнопка «Проверить». Сообщение о правильности вариантов появляется так же, как и в первой части задания: если ответ неверный, то появится окошко с сообщением об ошибке, если ответ верный, то появится окошко с сообщением о выполнении части работы.

В данной программе проверка правильности ответа студента выполняется так: происходит подстановка всех возможных комбинаций переменных в исходную ФАЛ и ФАЛ, которую ввел студент. Если в

результате сравнения этих двух подстановок не было отличий, тогда вариант решен верно.

4 Формирование отчета

После того, как студент решил обе части практической работы верно, что подтверждается самой программой в виде сообщений, имеется возможность автоматического создания отчета.

Данная возможность будет полезной не только для студента, но и для преподавателя.

Для автоматического формирования отчета по выполненной практической работе требуется готовый шаблон. Согласно стандарту «СТО 4.2-07-2014» [4], данные, необходимые для отчета следующие: Ф.И.О (фамилия, имя, отчество) студента, номер учебной группы, номер зачетной книжки, номер практической работы, Ф.И.О. преподавателя, год. Шаблон отчета выполнен в соответствии с этим стандартом.

На рисунке 44 изображена блок-схема создания отчета.



Рисунок 44 – Блок-схема создания отчета

Шаблон имеет расширение docx, и в него встроены закладки, которые будут заполняться вышеперечисленными данными. Данные вводятся студентом (кроме года, он берется из системы).

На рисунке 45 представлен шаблон для отчета по практической работе (титульный лист).

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

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ №
«Аксиомы и свойства алгебры логики»
Вариант

Преподаватель					
		подпись, дата		И	инициалы, фамилия
Студент				И	
	инициалы, фамилия	подпись, дата			инициалы, фамилия

Крыльцов

Рисунок 45 – Шаблон отчета по практической работе (титульный лист)

В данном шаблоне имеются следующие закладки (сверху вниз, слева направо):

- NumberLab (номер практической работы);
- NumberVar (номер варианта);
- NameTeacher (Ф.И.О. преподавателя);
- GroupName (номер группы);
- BookName (номер зачетной книжки);
- NameStudent (Ф.И.О. студента);
- Year (год);

Так как в данном случае в отчете нецелесообразно приводить решение к полученным ответам, в отчет будут вставлены только задания на практическую работу и конечные ответы.

На рисунке 46 представлен шаблон для отчета по практической работе (основная часть).

Цель работы: научиться пользоваться аксиомами исчисления и свойствами алгебры логики для преобразования логических функций.

Задание №1

Пользуясь аксиомами и свойствами алгебры логики, упростить выражение:

I

Ответ:

I

Задание №2

В данной ФАТГ поместить принцип де Моргана для вычисления значений комбинации на входных переменных, а также дизъюнкции – на входах конъюнкции:

I

Ответ:

I

Рисунок 46 – Шаблон отчета для практической работы (основная часть)

Данный шаблон содержит следующие закладки:

- Task1 (задание для первой части практической работы);
- Answer1 (ответ студента на первую часть практической работы);
- Task2 (задание для второй части практической работы);
- Answer2 (ответ студента на вторую часть практической работы);

Данные для этих закладок автоматически берутся со вкладок «Часть 2», «Часть 2».

После успешного выполнения двух частей задания студент имеет возможность создать отчет по данной практической работе (рисунок 47).

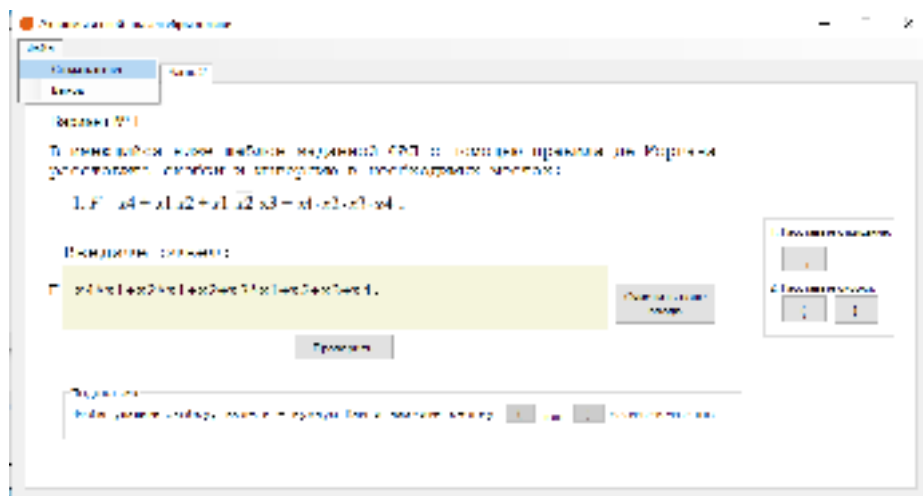


Рисунок 47 – Кнопка «Создать отчет»

Если хоть одна часть задания не будет выполнена, то появится окошко, сообщающее о невозможности создания отчета, пока не будут выполнены все части задания (рисунок 48).

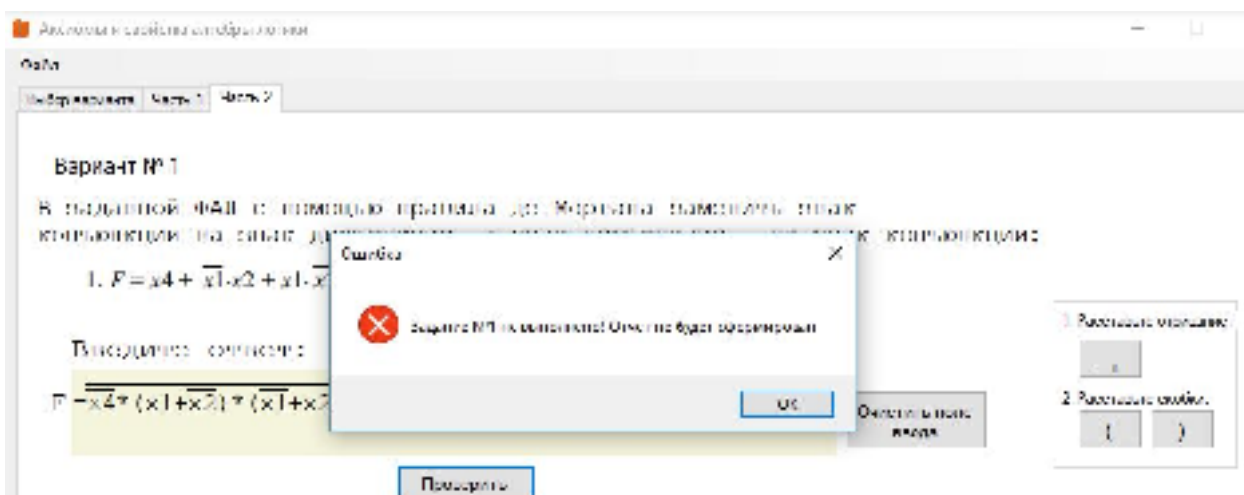


Рисунок 48 – Ошибка формирования отчета

После нажатия на вкладку «Создать отчет» появится окно с необходимыми данными для создания отчета (рисунок 49).

Рисунок 49 – Форма для создания отчета

Пример отчета содержится в ПРИЛОЖЕНИИ А.

5 Руководство пользователя

5.1 Программа для подготовки базы данных

Данная программа предназначена для создания базы ответов для первой части практической работы «Аксиомы и свойства алгебры логики», а также для загрузки заданий для каждого варианта практической работы.

При запуске программы перед пользователем открывается главное окно, которое содержит следующие компоненты: кнопка «Создать» (позволяет создать новые данные для выбранного варианта), кнопка «Открыть» (открывает уже имеющийся файл из базы данных в зависимости от выбранного варианта). **Последовательность действий добавления ответа и задания для варианта первой части задания:**

- 1) Нажмите кнопку «Создать» (рисунок 50).

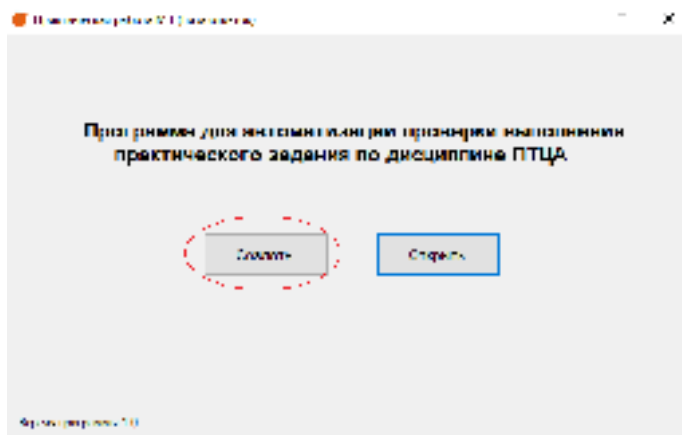


Рисунок 50 – Главное окно программы для подготовки базы данных

2) Перед вами откроется вкладка «Часть 1», в которой для каждого варианта создается свое задание и ответ на это задание (рисунок 51).

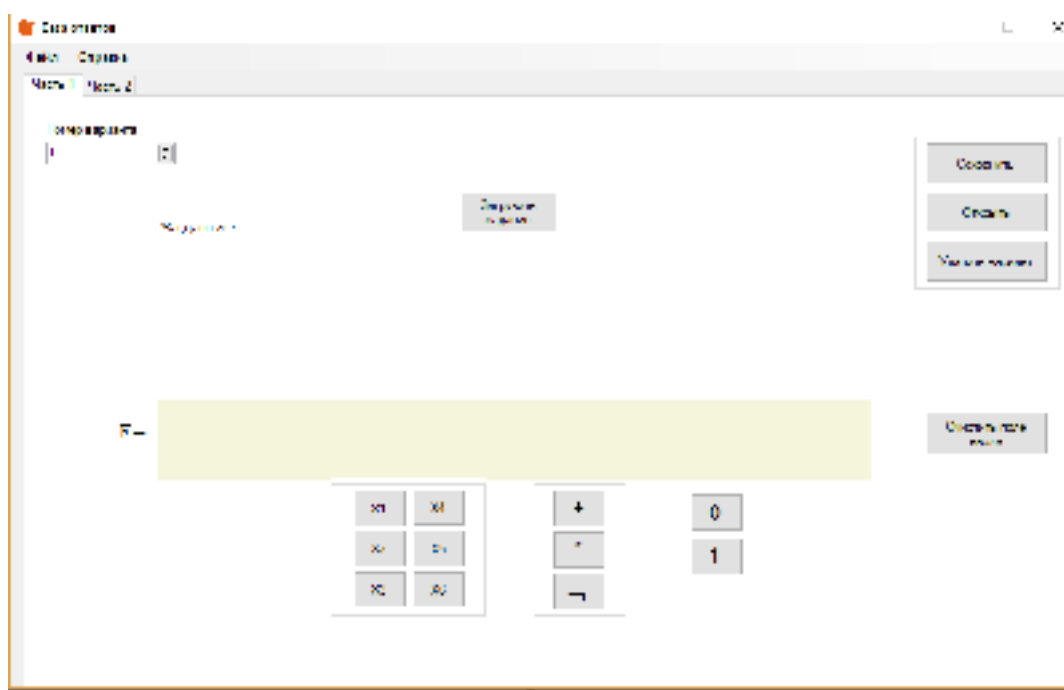


Рисунок 51 – Вкладка «Часть 1»

Далее необходимо выбрать необходимый вариант в графе «Номер варианта» (1), загрузить для него задание с помощью кнопки «Загрузить задание» (2), ввести ответ в поле для ввода (3). В случае, если при вводе ответа произошла опечатка, либо необходимо ввести ответ заново, требуется

нажать на кнопку «Очистить поле ввода» (4). На рисунке 52 изображены данные элементы.

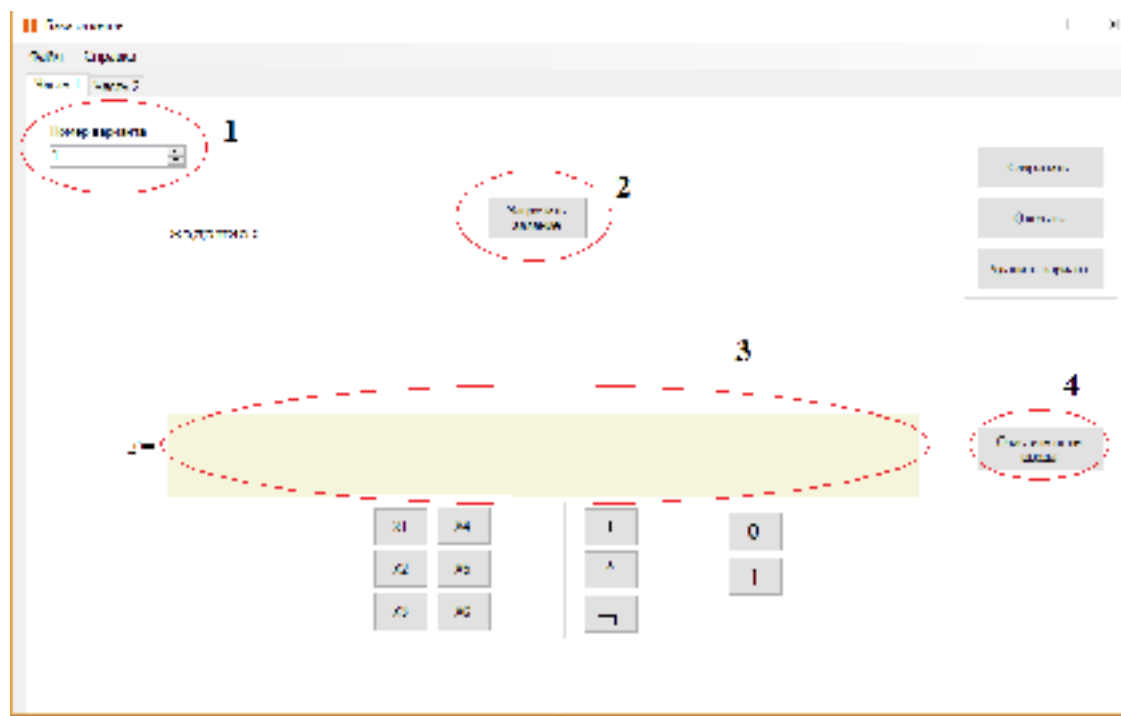



Рисунок 52 – Вкладка «Часть 1»

Для того, чтобы поставить инверсию над выделенными элементами, необходимо нажать на кнопку . Чтобы убрать отрицание над выделенными элементами следует также нажать на эту кнопку.

Для того, чтобы сохранить введенные данные, необходимо нажать на кнопку «Сохранить» (1). После этого появится подтверждающее сообщение (рисунок 54). Если в процессе работе нужно открыть другой вариант, тогда необходимо выбрать вариант в графе «Номер варианта» (2) и нажать на кнопку «Открыть» (3). Если требуется удалить данные для текущего варианта, тогда необходимо нажать на кнопку «Удалить вариант» (4). На рисунке 53 представлены данные элементы.

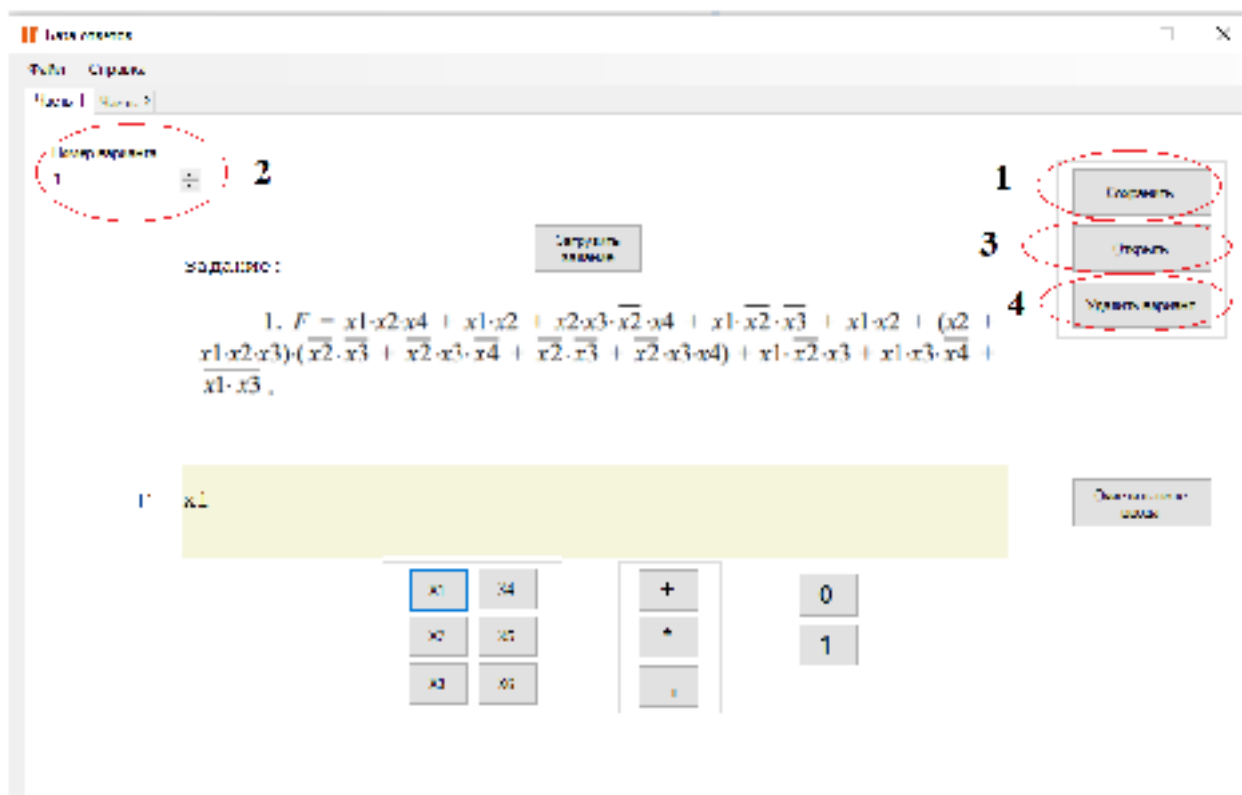


Рисунок 53 – Вкладка «Часть 1»

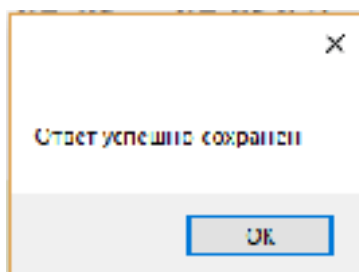


Рисунок 54 – Сообщение о сохранении

Последовательность действий загрузки задания для второй части практической работы:

- 1) Нажмите кнопку «Создать» (рисунок 55).

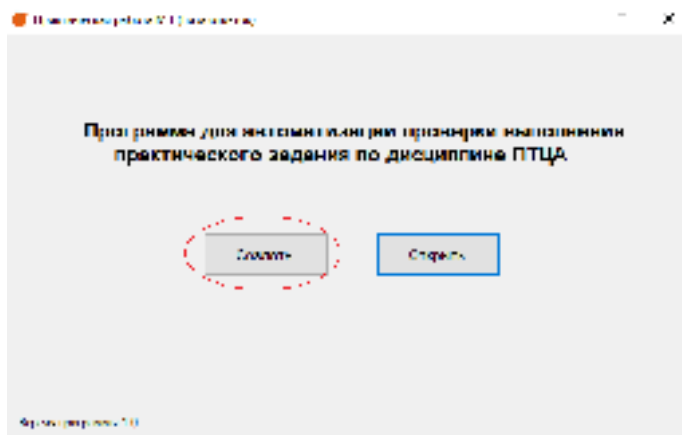


Рисунок 55 – Главное окно программы для подготовки базы данных

2) Перед вами откроется вкладка «Часть 2», в которой для каждого варианта создается свое задание (рисунок 56).

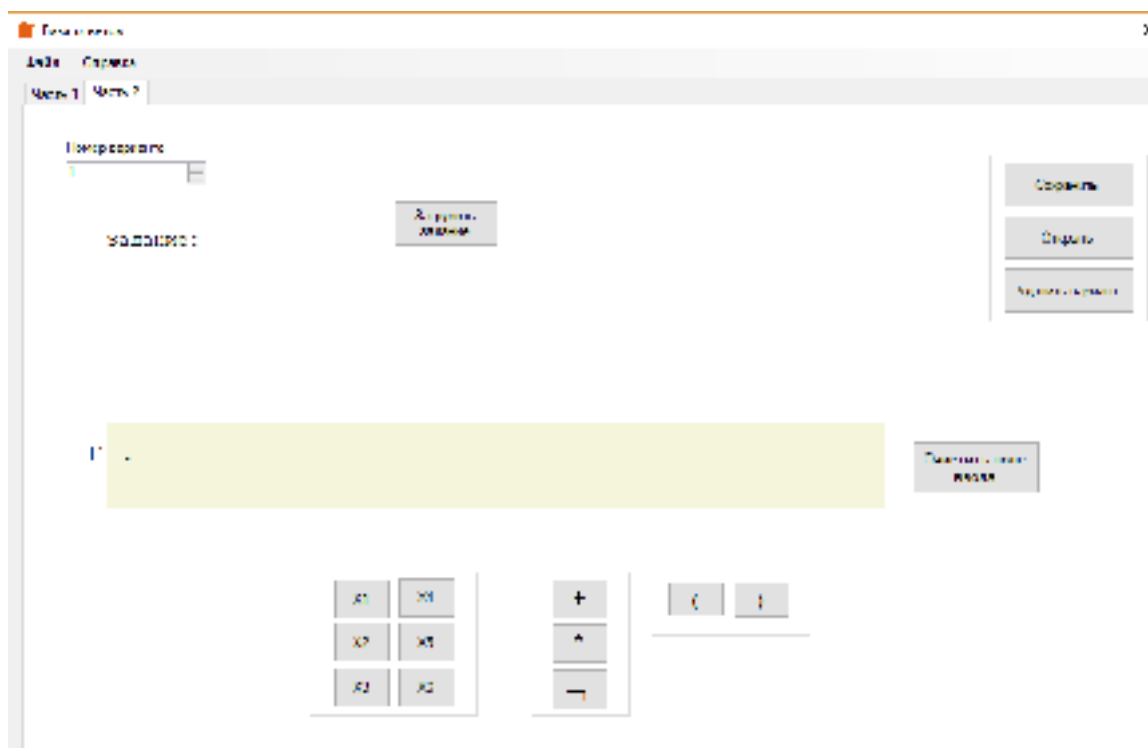


Рисунок 56 – Вкладка «Часть 1»

Далее необходимо выбрать вариант в графе «Номер варианта» (1), загрузить для него задание с помощью кнопки «Загрузить задание» (2), ввести это же задание вручную в поле для ввода (3). В случае, если при вводе

произошла опечатка, либо необходимо ввести задание заново, требуется нажать на кнопку «Очистить поле ввода» (4). На рисунке 57 изображены данные элементы.

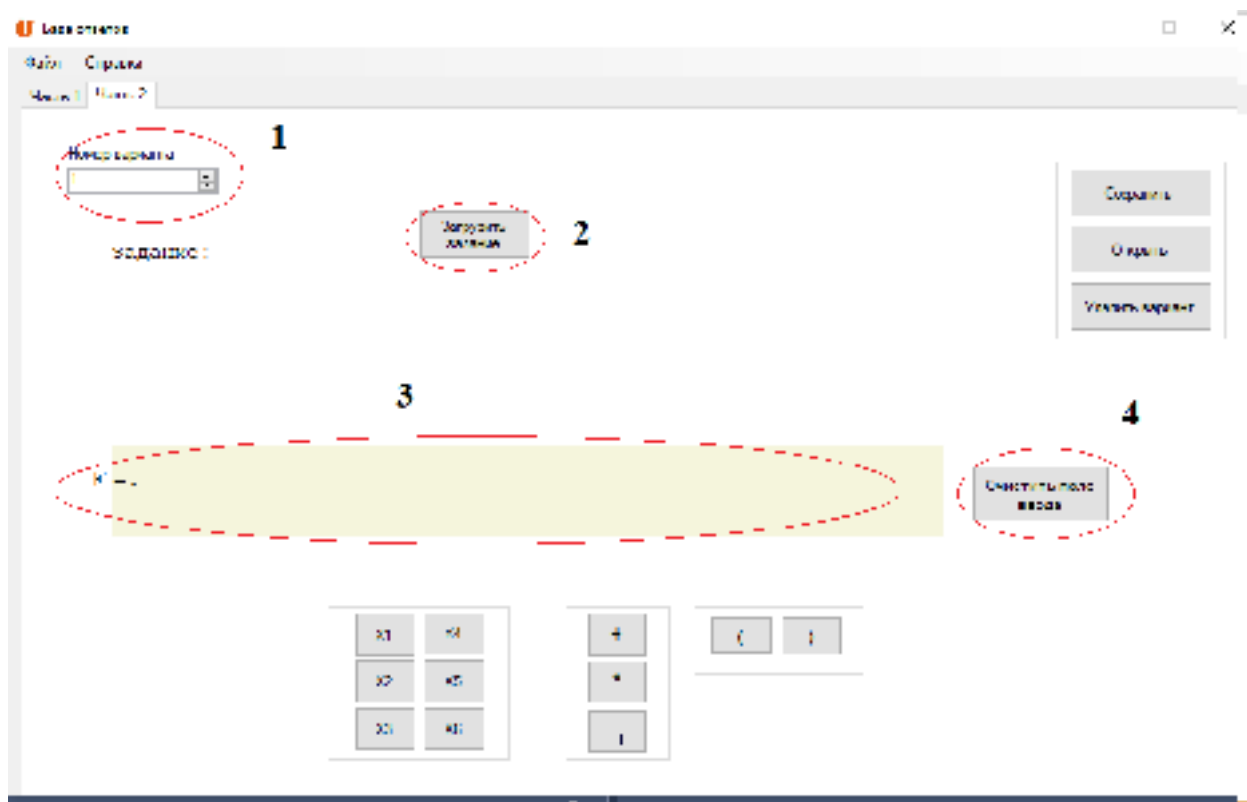




Рисунок 57 – Вкладка «Часть 1»

Для того, чтобы поставить инверсию над выделенными элементами, необходимо нажать на кнопку отрицания. Чтобы убрать отрицание над выделенными элементами следует также нажать на эту кнопку.

Для того, чтобы убрать имеющуюся скобку, необходимо нажать на кнопку  или  соответственно.

Для того, чтобы сохранить введенные данные, необходимо нажать на кнопку «Сохранить» (1). После этого появится подтверждающее сообщение (рисунок 59). Если в процессе работе нужно открыть другой вариант, тогда необходимо выбрать вариант в графе «Номер варианта» (2) и нажать на кнопку «Открыть» (3). Однако, если студенту требуется удалить данные для

текущего варианта, тогда необходимо нажать на кнопку «Удалить вариант» (4). На рисунке 58 представлены данные элементы.

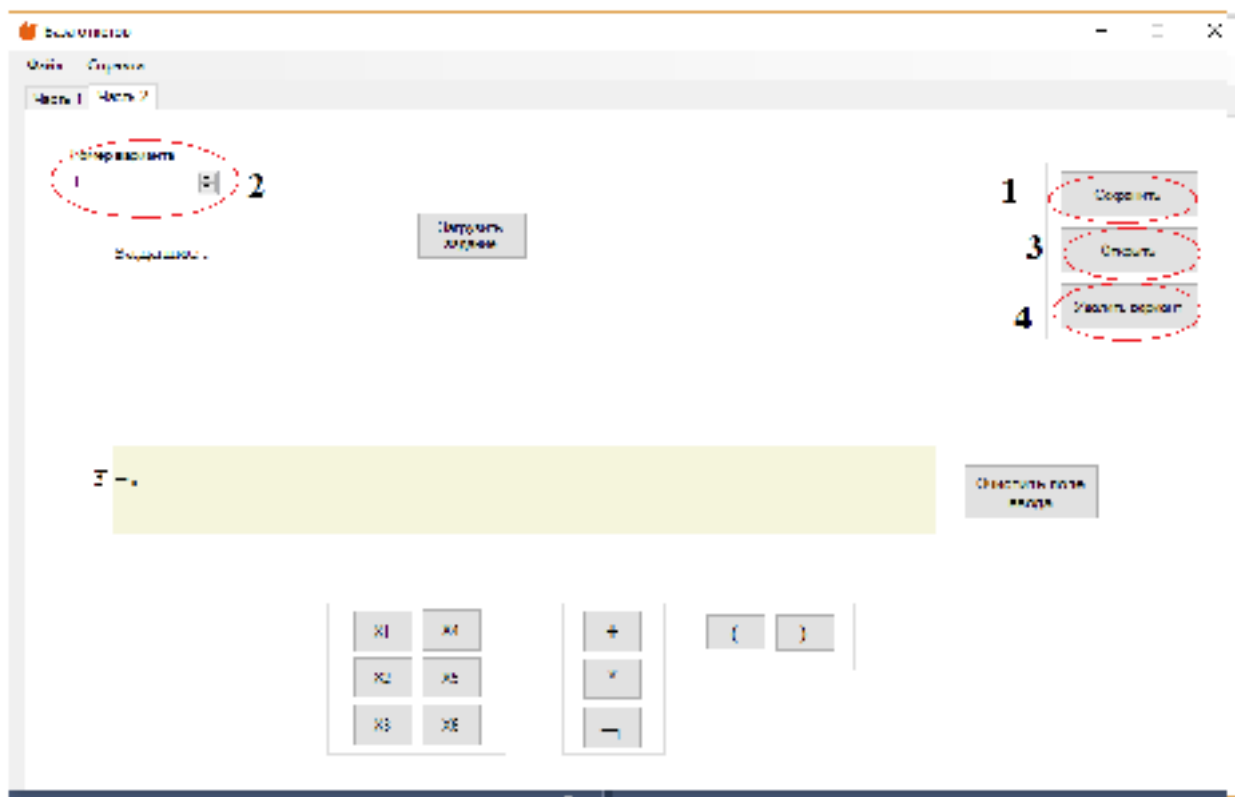


Рисунок 58 – Вкладка «Часть 1»

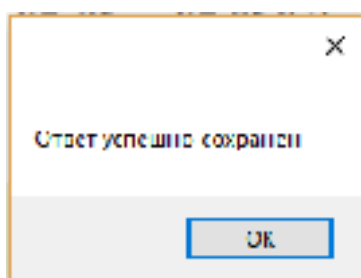


Рисунок 59 – Сообщение о сохранении

Последовательность действий открытия уже имеющихся файлов для варианта:

- 1) Нажмите на кнопку «Открыть» (рисунок 60).

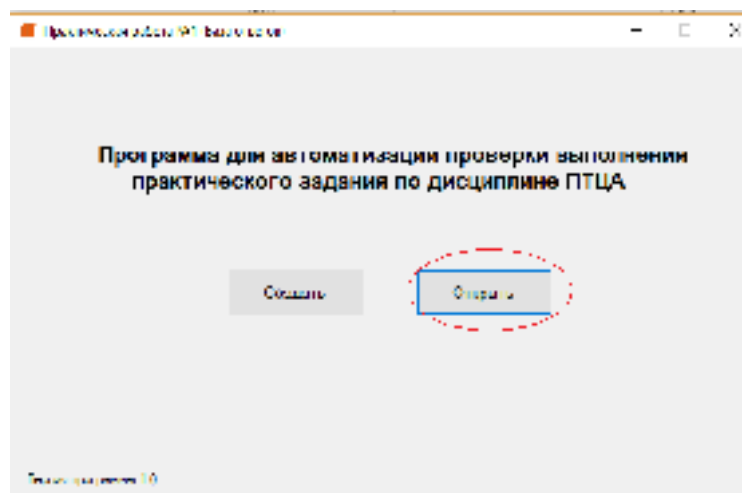


Рисунок 60 – Главное окно программы для проверки ответов

2) Откроется папка с базой данных, где нужно выбрать вариант и часть работы (рисунок 61).

1 ответ(1 часть).txt	25.05.2018 17:12	Текстовый докум
1 ответ(7 часть).txt	25.05.2018 17:12	Текстовый докум
1 шебюнк.txt	25.05.2018 17:12	Текстовый докум

Рисунок 61 – База данных

После этого откроется выбранный файл.

5.2 Программа для проверки результатов практической работы

Студент использует данную программу для того, чтобы узнать, верно ли он решил задание на практическую работу.

1) После запуска программы появляется главное окно, где необходимо выбрать вариант для выполнения практической работы (ввести номер варианта с помощью клавиатуры, либо с помощью соответствующих стрелочек в поле ввода варианта) (рисунок 62).

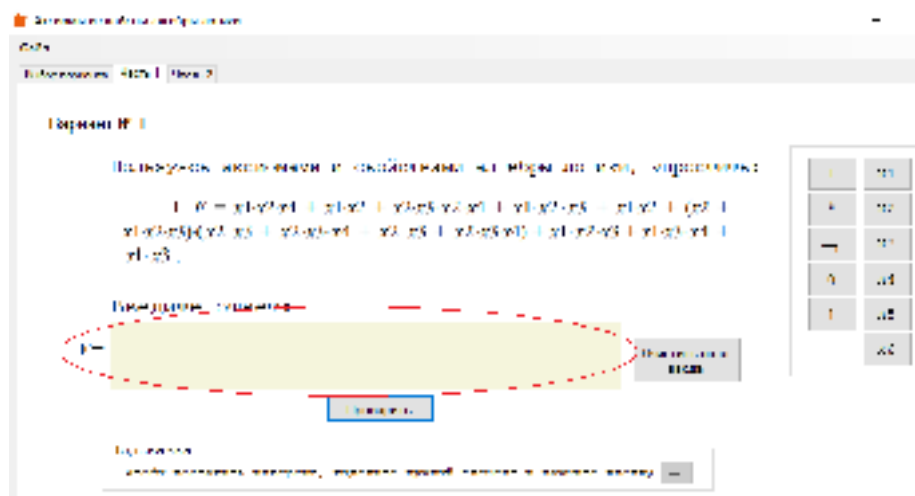


Рисунок 64 – Ввод ответа для первой части практической работы

4) Элементы, которые используются для ввода ответа в поле для ввода приведены на рисунке 65.

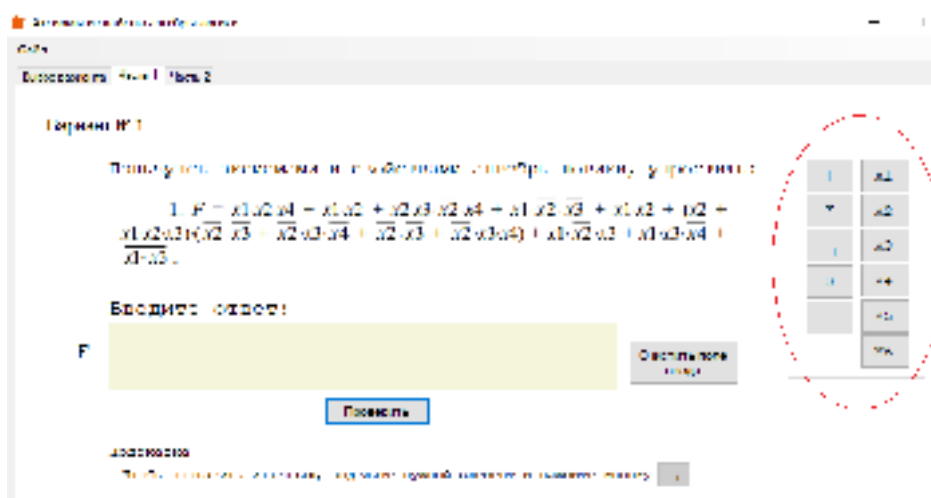


Рисунок 65 – Элементы для ввода ответа

5) Для того, чтобы поставить инверсию, необходимо выделить нужный элемент и нажать на кнопку инверсии (рисунок 66). На рисунке 67 представлен результат расстановки инверсии.

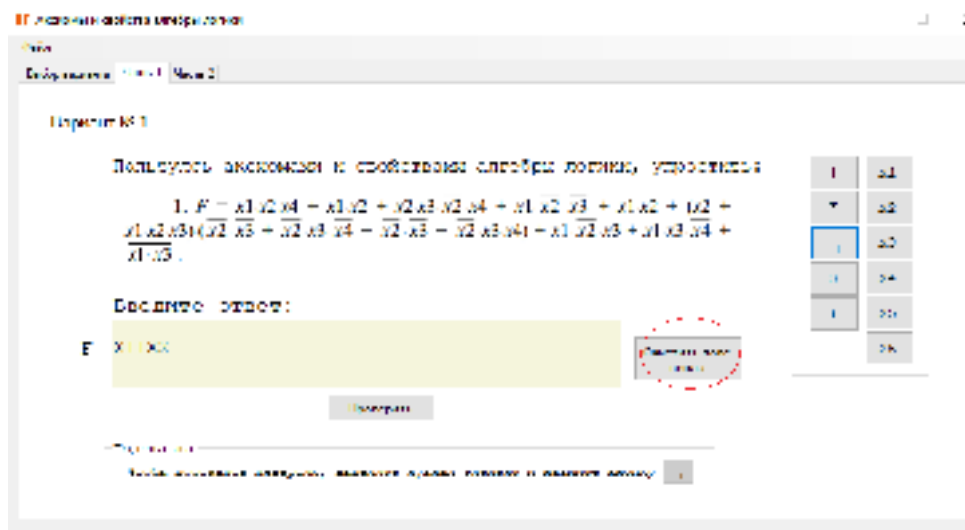


Рисунок 68 – Кнопка «Очистить поле ввода»

7) После ввода ответа для проверки необходимо нажать на кнопку «Проверить» (рисунок 69).

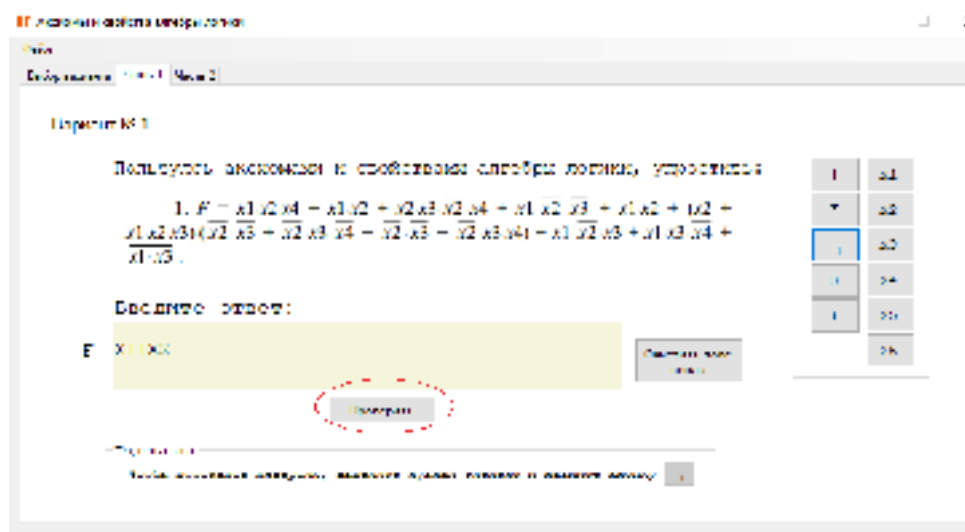


Рисунок 69 – Кнопка «Проверить»

Если ответ верный, тогда появится окошко с подтверждением (рисунок 70).

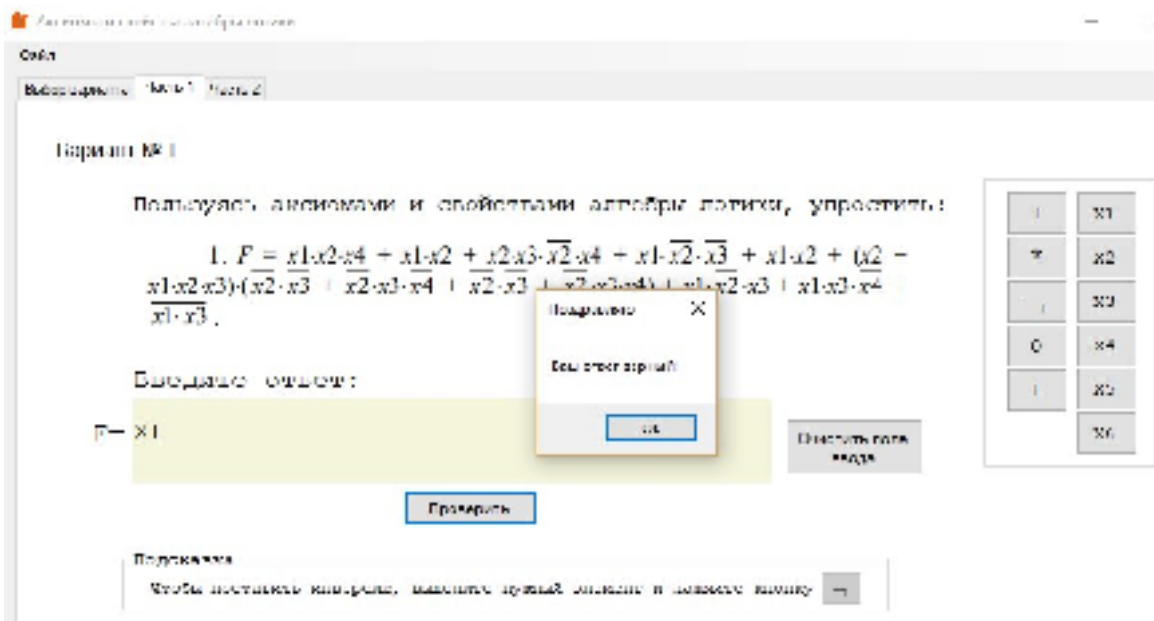


Рисунок 70 – Сообщение о правильном ответе

Если ответ неверный, тогда появится окошко с подтверждением (рисунок 71).

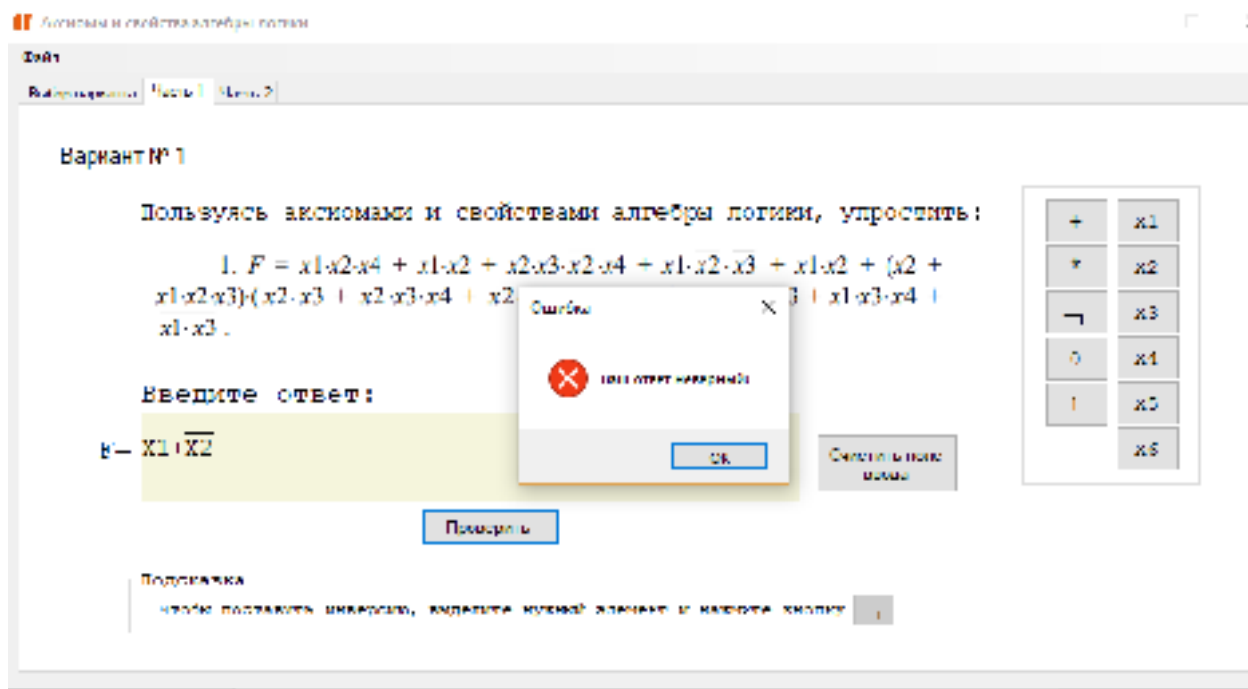


Рисунок 71 – Сообщение о неправильном ответе

8) После выполнения первой части задания необходимо перейти ко второй вкладке «Часть 2» (рисунок 72).

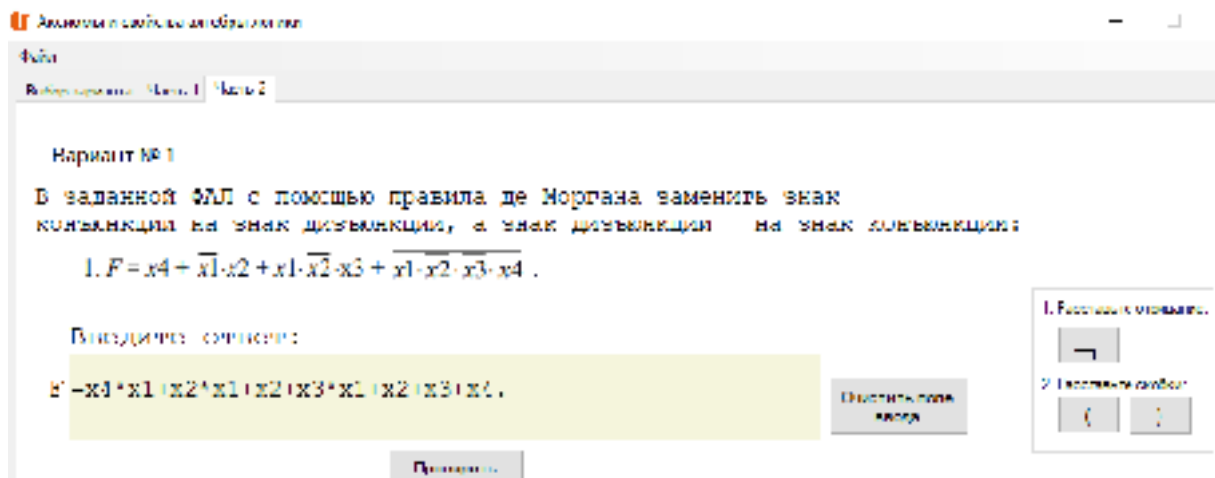


Рисунок 72 – Вкладка «Часть 2»

В этой части необходимо применить правила де Моргана к имеющемуся выражению. В поле ввода для ответа уже имеется шаблон, где необходимо расставить только инверсию и скобки в нужных местах (рисунок 73).

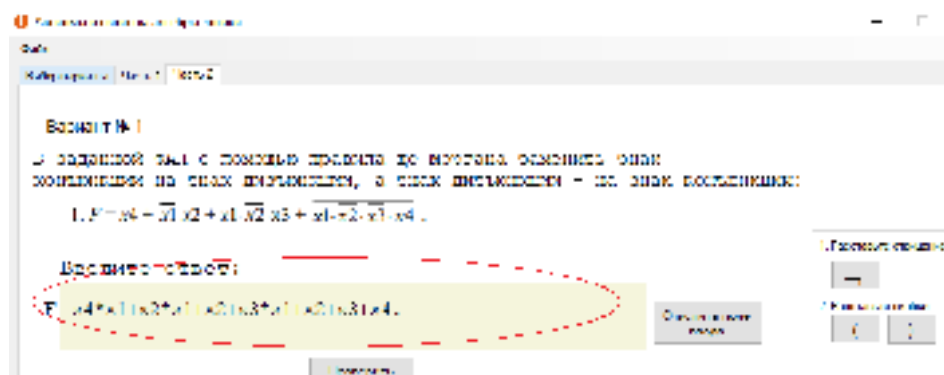




Рисунок 73 – Поле для ввода ответа

Инверсия ставится над элементами так же, как и в предыдущей вкладке «Часть 1» (см. пункт 5). Чтобы поставить скобку, необходимо нажать на кнопку  или  в зависимости от того, какая требуется (рисунок 74). На рисунке 75 представлен результат расстановки скобок.

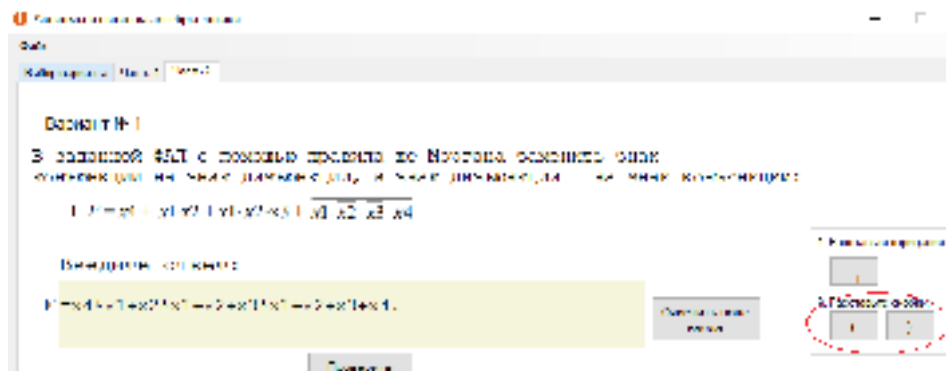


Рисунок 74 – Расстановка скобок в выражении

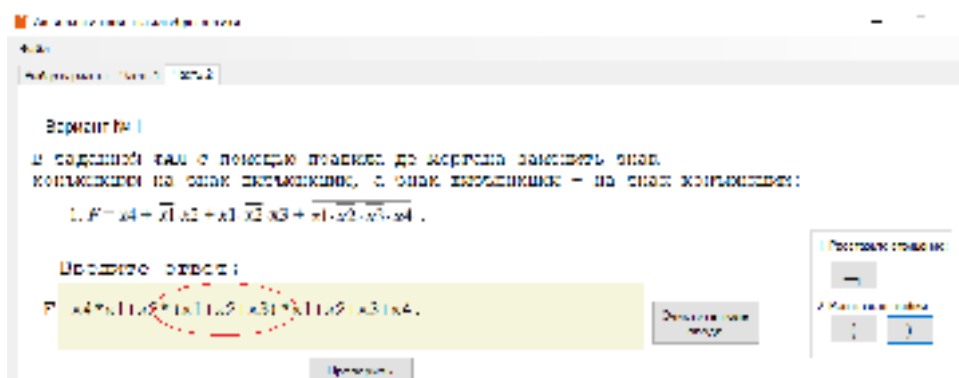




Рисунок 75 – Результат расстановки скобок

Чтобы удалить имеющуюся скобку, необходимо выделить эту скобку и нажать на кнопку  или  в зависимости от того, какую требуется удалить (рисунок 76).

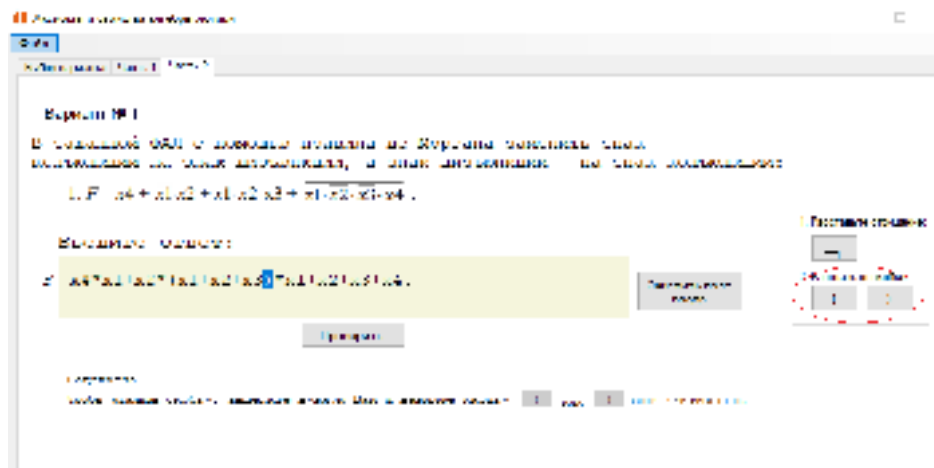


Рисунок 76 – Удаление скобок

Для проверки ответа необходимо нажать на кнопку «Проверить» (рисунок 77).

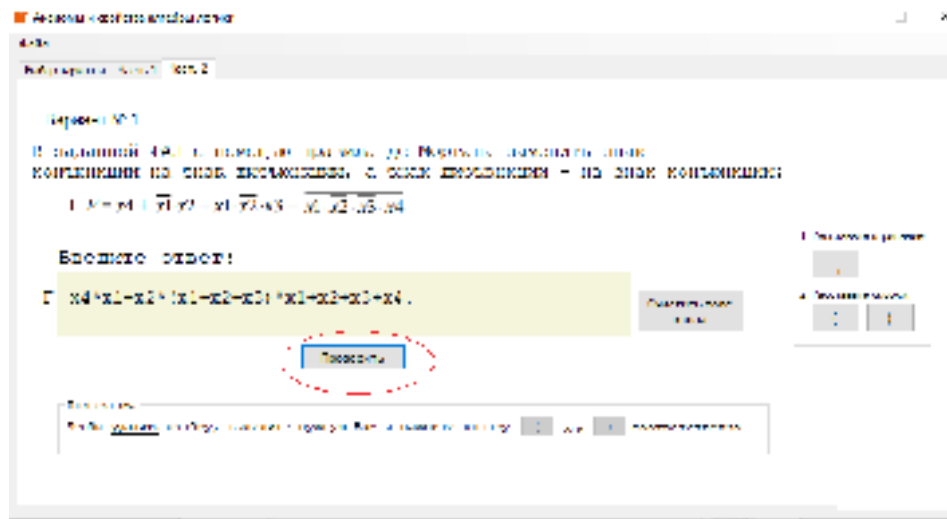


Рисунок 77 – Кнопка «Проверить»

Если ответ оказался верным, то появится подтверждающее окно (рисунок 78).

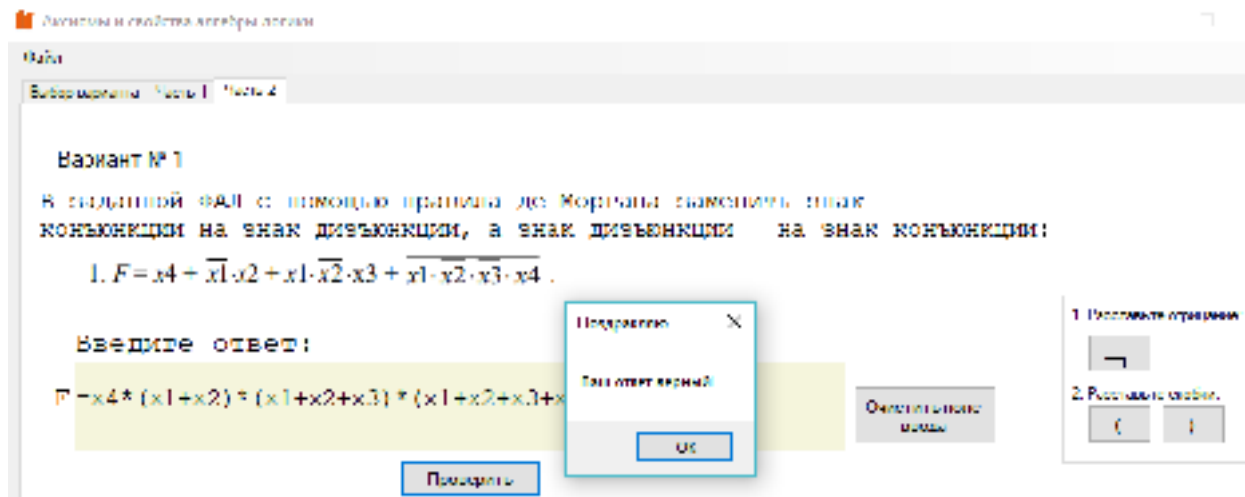


Рисунок 78 – Сообщение о правильном ответе

Если ответ оказался неверным, то появится подтверждающее окно (рисунок 79).

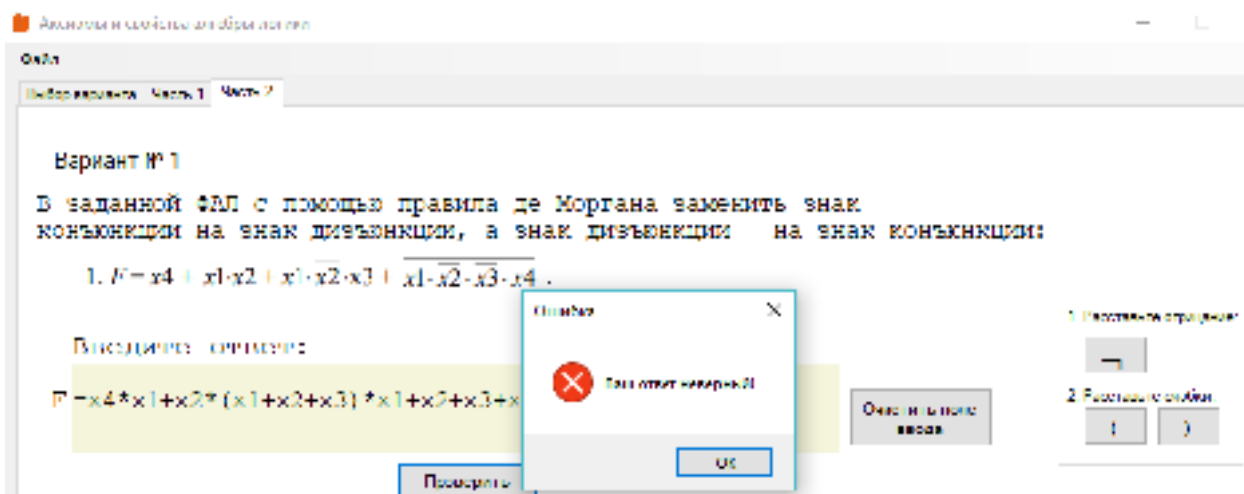


Рисунок 79 – Сообщение о неправильном ответе

При успешном выполнении двух частей практической работы существует возможность автоматического создания отчета. Для того, чтобы создать отчет о практической работе:

- Убедитесь в том, что обе части задания выполнены верно, иначе отчет не будет сформирован.
- Нажмите на «Файл» -> «Создать отчет» (рисунок 80).

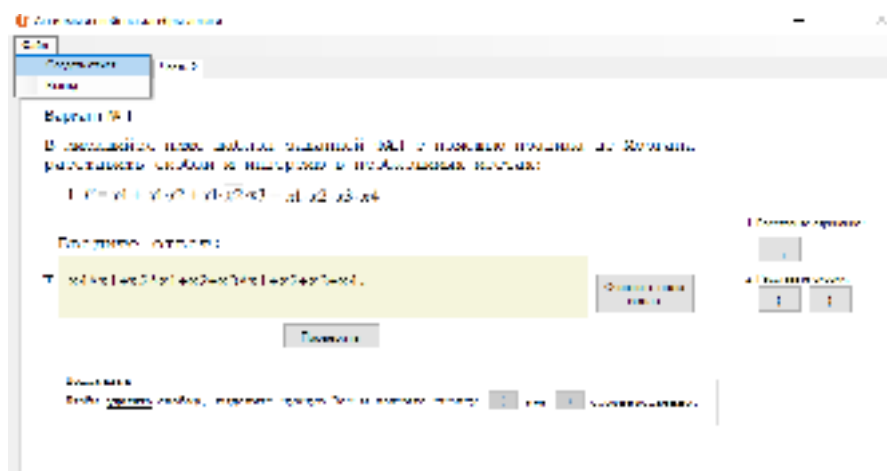


Рисунок 80 – Создание отчета

Появится окно, где необходимо ввести следующую информацию о себе: Ф.И.О, номер учебной группы, номер зачетной книжки (рисунок 81). После ввода данной информации необходимо нажать на кнопку «Принять». Далее будет автоматически сформирован отчет.

Создание отчета

ФИС студента:

Номер группы:


Номер зак. книжки:

ФИС преподавателя:

Номер практической работы:

Принять

Рисунок 81 – Форма для создания отчета

Чтобы выйти из программы, следует нажать на кнопку «Выход», либо нажать на  в правом верхнем углу.

ЗАКЛЮЧЕНИЕ

Таким образом, в ходе выполнения ВКР был разработан программный комплекс для автоматизации проверки выполнения практической работы «Аксиомы и свойства алгебры логики» по дисциплине «Прикладная теория цифровых автоматов». Данный программный комплекс включает в себя две программы: одна программа является базой ответов и заданий для практической работы (программа используется только преподавателем), другая программа предназначена для проверки ответов студента (используется только студентом).

СПИСОК СОКРАЩЕНИЙ

ВКР – Выпускная квалификационная работа;

ПТЦА – Прикладная теория цифровых автоматов;

ФАЛ – Функции алгебры логики;

Ф.И.О – Фамилия, Имя, Отчество.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Булева алгебра [Электронный ресурс] // ru.bmstu.wiki: Национальная библиотека им. Н.Э. Баумана Bauman National Library. URL: https://ru.bmstu.wiki/Булева_алгебра (дата обращения: 10.04.2018).

2. ГОСТ 2.316–2008 Единая система конструкторской документации. Правила нанесения надписей, технических требований и таблиц на графических документах. Общие положения. Взамен ГОСТ 2.316–68; дата введ. 01.07.2009. – Москва: Стандартинформ, 2009. – 12 с.

3. Основы алгебры и логики [Электронный ресурс] // e.sfu-kras.ru: Система электронного обучения СФУ. URL: https://e.sfu-kras.ru/pluginfile.php/84381/mod_resource/content/2/content/2/Лекция%201.pdf (дата обращения: 10.04.2018).

4. Система менеджмента качества. Общие требования к построению, изложению и оформлению документов учебной деятельности. СТО 4.2-07-2014. Красноярск: ИПЦ СФУ, 2014. – 60 с.

5. Windows Forms [Электронный ресурс] // docs.microsoft.com: Microsoft – официальная страница. URL: <https://docs.microsoft.com/ru-ru/dotnet/framework/winforms/> (дата обращения: 10.04.2018).

ПРИЛОЖЕНИЕ А

Пример отчета

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

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ №1
«Аксиомы и свойства алгебры логики»
Вариант 1

Преподаватель

подпись, дата

А.И. Постников

инициалы, фамилия

Студент КИ14-07Б 031401950

номер группы, зачетной книжки

подпись, дата

А.Н. Анисимова

инициалы, фамилия

Красноярск 2018

Цель работы: научиться пользоваться аксиомами законами и свойствами алгебры логики для преобразования логических функций.

Задание №1

Пользуясь аксиомами и свойствами алгебры логики, упростить выражение:

$$1. F = \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_4} + \overline{x_1} \cdot \overline{x_2} + \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_2} \cdot \overline{x_4} + \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} + \overline{x_1} \cdot \overline{x_2} + (\overline{x_2} + \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3}) \cdot (\overline{x_2} \cdot \overline{x_3} + \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4} + \overline{x_2} \cdot \overline{x_3} + \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4}) + \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} + \overline{x_1} \cdot \overline{x_3} \cdot \overline{x_4} + \overline{x_1} \cdot \overline{x_3}.$$

Ответ:

$$\overline{x_1 + x_2}$$

Задание №2

В данной ФАЛ с помощью правила де Моргана заменить знак конъюнкции на знак дизъюнкции, а знак дизъюнкции – на знак конъюнкции:

$$1. F = x_4 \vee \overline{x_1} \cdot \overline{x_2} \vee \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \vee \overline{\overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3} \cdot \overline{x_4}}.$$

Ответ:

$$\overline{\overline{\overline{x_4} \vee (\overline{x_1 + x_2}) \vee (\overline{x_1 + x_2 + x_3}) \vee (\overline{x_1 + x_2 + x_3 + x_4})}}.$$