

Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

Пояснительная записка

к курсовой работе

по курсу «Программирование»

на тему «Разработка программы сложной структуры методом
нисходящего программирования»

Выполнил:

студент группы 21BB2:

Сагатов А.К.

Принял:

Голотенков Н.О.

от

08.06.22

Пенза 2022

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет Вычислительной техники

Кафедра "Вычислительная техника"

"УТВЕРЖДАЮ"

Зав. кафедрой ВТ



М.А. Митрохин

« ____ » _____ 20__ г

ЗАДАНИЕ

на курсовое проектирование по курсу

Программирование

Студенту Салателаву Артёму Викторовичу Группа 21 В В 2

Тема проекта: *Разработка программы сложной структуры методом нисходящего программирования.*

Исходные данные (технические требования) на проектирование

Приложение: переводчик (словарь)

Функции: возможность поиска слов по словарю, их перевод прямой и обратной. Запись новых слов в словарь.

Обязательные требования к программе:

1. Многомодульность.
2. Использование сложных типов данных – структур, списков, файлов.
3. Режим работы видеосистемы – текстовый / графический.
4. Устройство ввода информации – клавиатура, мышь.
5. Пользовательский интерфейс должен быть построен на основе меню и панели инструментов.
6. Наличие заставки.
7. Операционная система MS Windows.
8. Язык программирования – Си и Ассемблер.
9. Среда разработки ПО – Microsoft Visual Studio.

Объем работы по курсу

1. Расчетная часть

Разработка программы.

2. Графическая часть

Схема данных, схема ресурсов системы, схема работы системы, иерархическая структура программы, схема взаимодействия программы.

3. Экспериментальная часть

Отладка программы.

Срок выполнения проекта по разделам

В соответствии с графиком.

Дата выдачи задания " 22 " Февраль 2022

Дата защиты проекта " 08 " 06. 22

Руководитель Гелометков Н.О.

Задание получил " 22 " Февраль 2022 г.

Студент С.А. Сагитов А.К.

Содержание

Введение	6
1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ.....	7
2. ТЕХНИЧЕСКОЕ ЗАДАНИЕ.....	8
2.1 Введение.....	8
2.1.1 Наименование продукта	8
2.1.2 Краткая характеристика области применения.....	8
2.2 Основание для разработки	8
2.2.1 Документ, на основании которого ведется разработка	8
2.2.2 Организация, утвердившая документ	8
2.3 Требования к разработке	8
2.3.1 Требования к функциональным характеристикам	8
2.3.2 Требования к составу и параметрам технических средств....	9
2.3.3 Требования к информационной и программной совместимости	9
2.4 Требования к программной документации.....	9
2.5 Стадии и этапы разработки	10
2.6 Порядок контроля и приема	10
3. РАЗРАБОТКА МОДЕЛИ СИСТЕМЫ.....	11
3.1 Диаграмма классов	7
3.2 Схема работы программы.....	15
4. ОПИСАНИЕ ПРОГРАММЫ.....	16
4.1 Общие сведения	16
4.2 Функциональное назначение	16

4.3	Используемые технические средства	16
4.4	Вызов и загрузка	16
4.5	Входные данные.....	16
4.6	Выходные данные	17
5.	ОТЛАДКА И ТЕСТИРОВАНИЕ.....	17
6.	ПРОГРАММА И МЕТОДИКА ИСПЫТАНИЙ.....	18
5.1	Объект испытаний	18
5.2	Требования к программе	18
5.3	Требования к программной документации.....	18
5.4	Средства и порядок испытаний.....	18
5.5	Методы испытаний	19
	ЗАКЛЮЧЕНИЕ	21
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	22
	Приложение А	20
	Приложение Б	41

Введение

В современном мире языковой барьер может стать большим препятствием общения. Самый доступный и оперативный способ устранения этого препятствия – машинный перевод текста.

Программа «Переводчик текстов» позволяет осуществлять перевод текстов с английского языка на русский и обратно.

Пояснительная записка содержит анализ предметной области, техническое задание, описание программы, программу и методику испытаний, описание применения.

1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

Предметной областью для данной разработки является перевод текстов с английского языка на русский и обратно.

Интерфейс приложения имеет схожие черты с интерфейсом приложения *Stylus*. Словарь системы *Stylus* состоит из слов. Под словом понимается цепочка букв от пробела до пробела. Каждому слову входного языка в словаре программы соответствует одно слово другого языка. Помимо этого, в программе реализованы функции: записи перевода неизвестных слов в словарь и открытие словаря.

2. ТЕХНИЧЕСКОЕ ЗАДАНИЕ

2.1 Введение

2.1.1 Наименование продукта

Программа «Переводчик текстов» позволяет осуществлять перевод текстов с английского языка на русский и обратно.

2.1.2 Краткая характеристика области применения

Данная программа является средством перевода текстов.

2.2 Основание для разработки

2.2.1 Документ, на основании которого ведется разработка

Задание на курсовую работу по дисциплине "Программирование на языках высокого уровня".

2.2.2 Организация, утвердившая документ

Пензенский Государственный Университет.

2.3 Требования к разработке

2.3.1 Требования к функциональным характеристикам

1. Возможность ввода текста в окно источника.
2. Возможность выбора направления перевода (с английского языка на русский и обратно).
3. Проверка выбора направления перевода. Если направление перевода не было выбрано, то программа выведет на экран сообщение «Выберите направление перевода».
4. Перевод текста и вывод переведённого текста в окно результата.
5. Запись в электронный словарь перевода неизвестного слова. В случае, если программе неизвестен перевод какого-либо слова, то программа попросит пользователя ввести перевод и запишет его в электронный словарь.
6. Открытие словаря в зависимости от выбора направления перевода
7. Проверка направления перевода и введённого текста. В случае несовпадения направления перевода и какого-либо введённого слова, оно будет перенесено из окна источника в окно результата.

8. Возможность редактирования текста как в окне источника перевода, так и в окне результата
9. Возможность работы с буфером обмена (копирование, вырезание, вставка)
10. Вызов помощи
11. Простой и удобный интерфейс.

2.4 Требования к программной документации

Предварительный состав программной документации:

- "Техническое задание", текст программы с комментариями, описание алгоритма и программы.
- "Руководство пользователя".

3. РАЗРАБОТКА МОДЕЛИ СИСТЕМЫ

3.1 Диаграмма классов

Для решения поставленной задачи необходимы следующие классы: TForm, TButton, TTextBox, TRadioButton, TLabel.

Диаграмма классов представлена на рисунке 1.

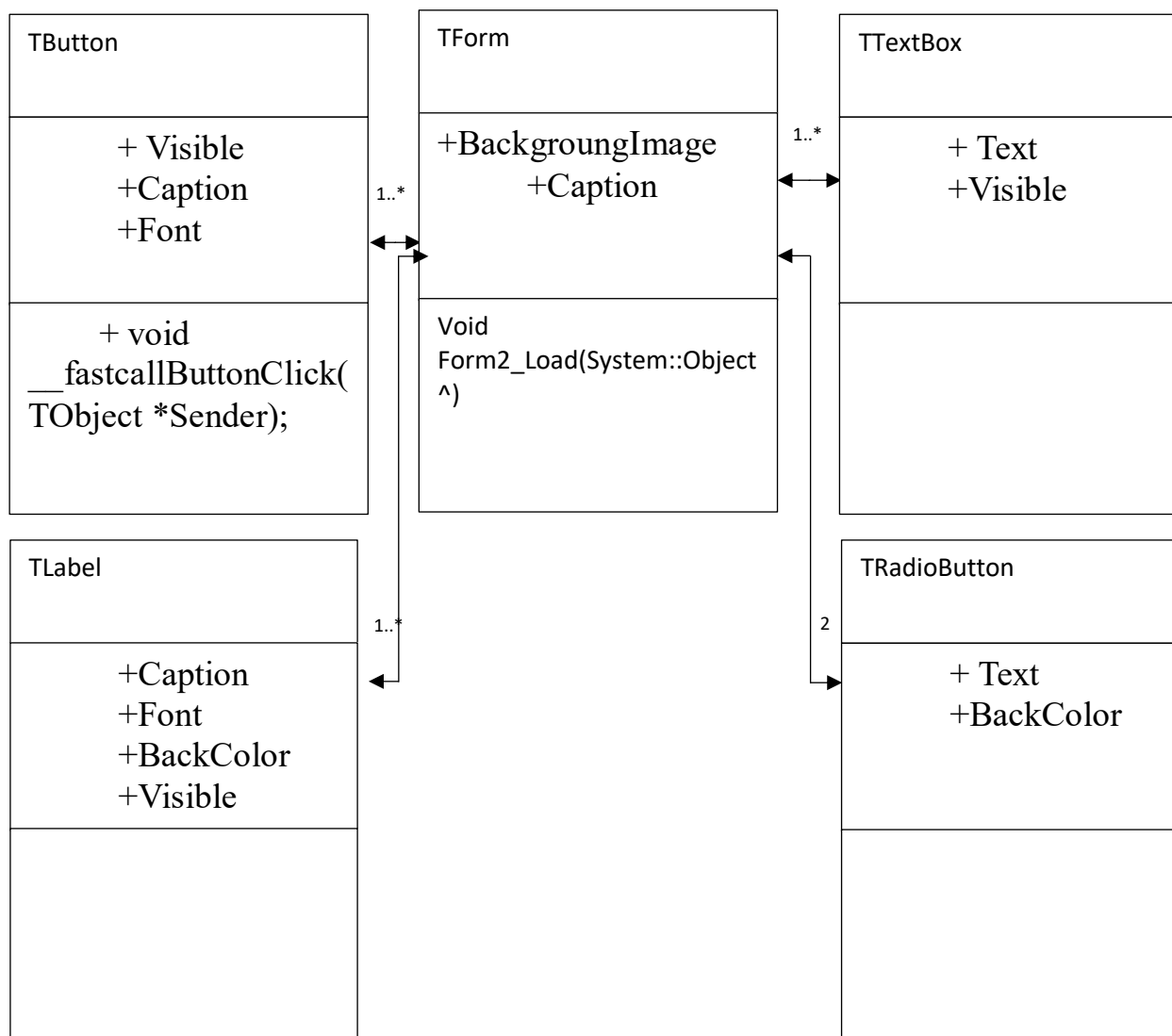


Рисунок 1 – Диаграмма классов

Класс TForm имеет следующие атрибуты:

Caption – текст на заголовке формы;

BackgroundImage – изображение на форме;

Класс TForm имеет следующий метод:

Обработчик события создания формы System::Void Form2_Load(System::Object^ sender, System::EventArgs^ e)

Класс TButton имеет следующие атрибуты:

Caption – текст на кнопке;

Visible – свойство видимости;

Font– свойства текста на кнопке;

Класс TButton имеет следующие методы:

Обработчик события клика System::Void button1_Click(System::Object^ sender, System::EventArgs^ e);

Обработчик события клика System::Void button2_Click(System::Object^ sender, System::EventArgs^ e);

Обработчик события клика System::Void button3_Click(System::Object^ sender, System::EventArgs^ e);

Обработчик события клика System::Void button4_Click(System::Object^ sender, System::EventArgs^ e);

Обработчик события клика System::Void button5_Click(System::Object^ sender, System::EventArgs^ e);

Обработчик события клика System::Void button1_Click(System::Object^ sender, System::EventArgs^ e);

Класс TTextBox имеет следующие атрибуты:

Text – текст, находящийся в поле объекта;

Visible – свойство видимости;

Класс TLabel имеет следующие атрибуты:

Caption – текст на метке;

Visible – свойство видимости;

BackColor– фон на метке;

Font – свойства текста;

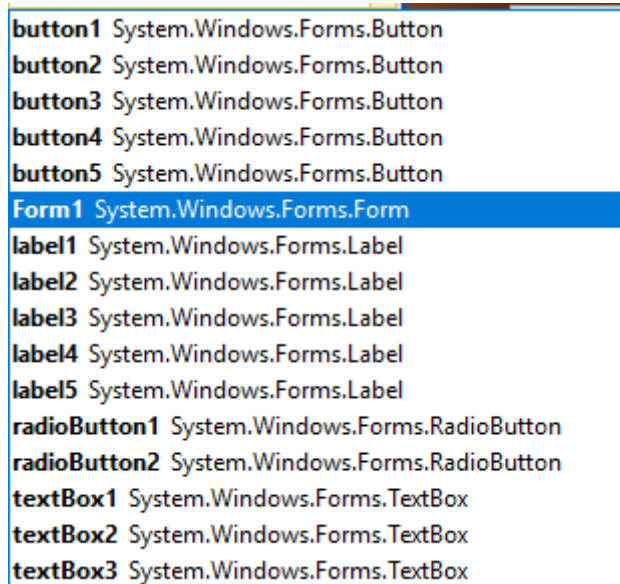
Класс TRadioButton имеет следующие атрибуты:

BackColor– фон на метке;

Text – текст на метке;

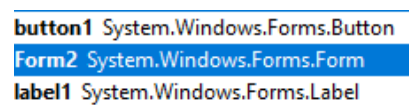
Структура формы Form1 представлена на рисунке 2.

Структура формы Form2 представлена на рисунке 3.



button1	System.Windows.Forms.Button
button2	System.Windows.Forms.Button
button3	System.Windows.Forms.Button
button4	System.Windows.Forms.Button
button5	System.Windows.Forms.Button
Form1	System.Windows.Forms.Form
label1	System.Windows.Forms.Label
label2	System.Windows.Forms.Label
label3	System.Windows.Forms.Label
label4	System.Windows.Forms.Label
label5	System.Windows.Forms.Label
radioButton1	System.Windows.Forms.RadioButton
radioButton2	System.Windows.Forms.RadioButton
textBox1	System.Windows.Forms.TextBox
textBox2	System.Windows.Forms.TextBox
textBox3	System.Windows.Forms.TextBox

Рисунок 2 – Структура формы Form1



button1	System.Windows.Forms.Button
Form2	System.Windows.Forms.Form
label1	System.Windows.Forms.Label

Рисунок 3 – Структура формы Form2

На Form1 размещены следующие объекты:

- Button1 – кнопка перевода введённого текста.
- Button2 – кнопка записи перевода неизвестного слова в словарь.
- Button3 – кнопка закрытия окна для записи перевода неизвестного слова в словарь.
- Button4 – кнопка выхода из программы.
- Button5 – кнопка вызова помощи.
- TextBox1 – однострочный редактор для ввода текста, который необходимо перевести.
- TextBox2 – однострочный редактор для вывода переведённого текста

- TextBox3 – однострочный редактор для ввода перевода неизвестного программе слова.
- Label1 – сообщение о том, что пользователю необходимо ввести текст, который он хочет перевести.
- Label2 – сообщение о том, что программе неизвестен перевод какого-либо слова.
- Label3 – сообщение о том, что пользователю необходимо ввести перевод неизвестного программе слова.
- Label4 – сообщение о выборе направления перевода.
- RadioButton1 – метка выбора направления перевода с Английского языка на Русский.
- RadioButton2 – метка выбора направления перевода с Русского языка на Английский.

На Form2 размещены следующие объекты:

- Button1 – кнопка закрытия помощи.
- Label1 – информация о пользовании переводчиком.

Расположение объектов формы Form1 представлено на рисунке 4.

Расположение объектов формы Form2 представлено на рисунке 5.



Рисунок 4 – Расположение объектов формы Form1

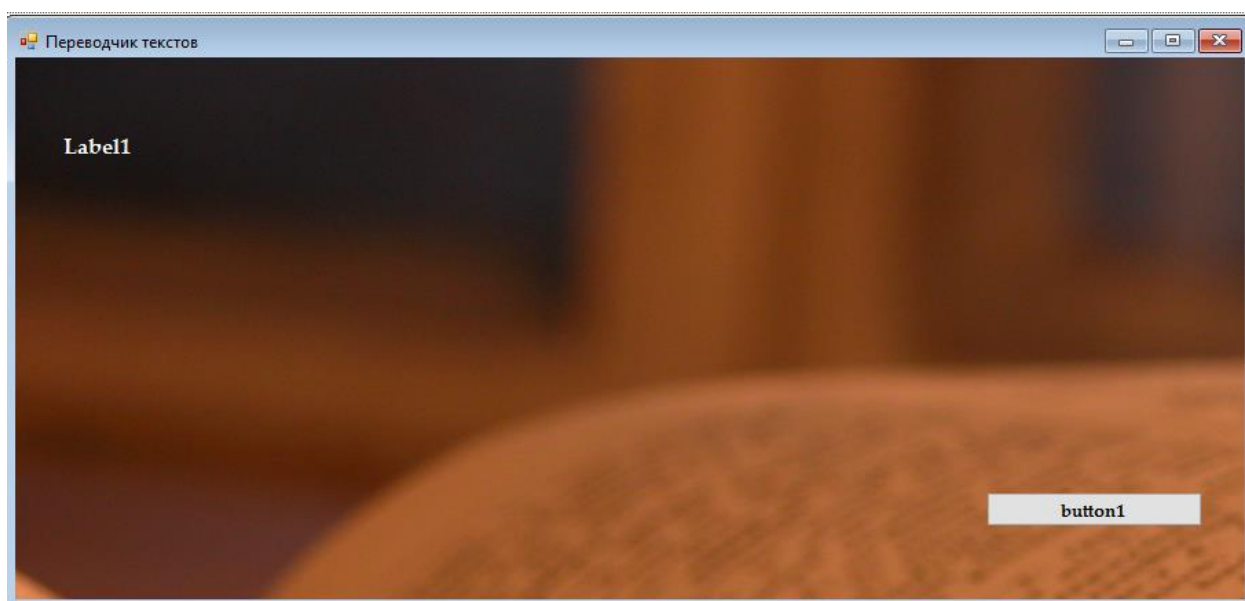


Рисунок 5 – Расположение объектов формы Form2

СХЕМА РАБОТЫ ПРОГРАММЫ

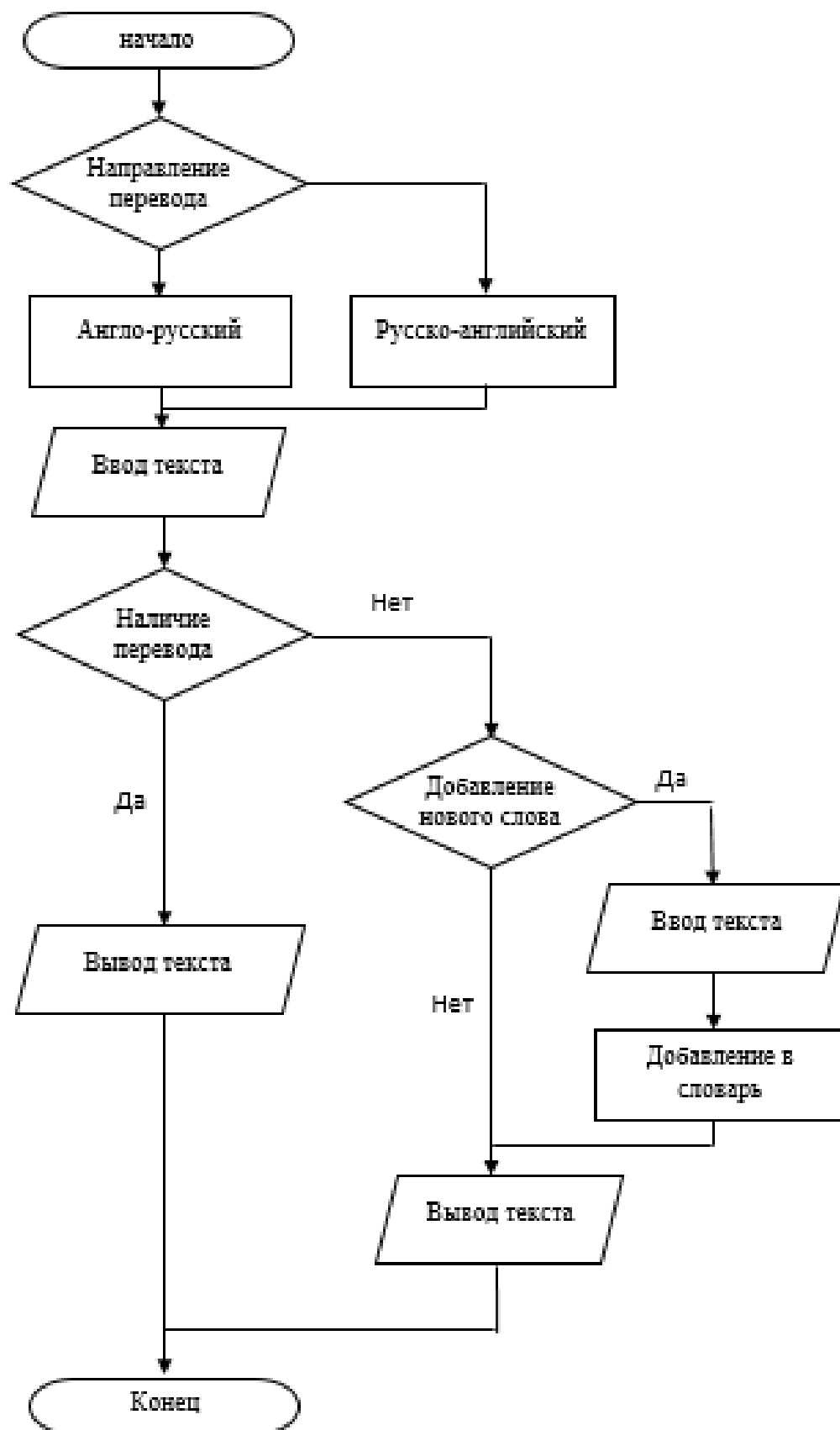


Рис. 6

4. ОПИСАНИЕ ПРОГРАММЫ

4.1 Общие сведения

Программа «Переводчик текстов» создана на языке C++ со вставкой на языке Ассемблер в среде разработки Microsoft Visual C++.

Текст программы представлен в приложении А.

Приложение состоит из файла исходного кода `unit1.cpp`, главной формы `Form1`, второстепенной формы `Form2` и файла проекта `Project1`.

4.2 Функциональное назначение

Программа является средством перевода текстов с английского языка на русский и обратно.

4.3 Используемые технические средства

Программа предназначена для работы на x64-совместимых персональных компьютерах, работающих под управлением операционной системы Microsoft Windows версии XP или выше и имеющих следующие минимальные характеристики: тактовая частота процессора – не ниже 1,8 ГГц; оперативная память – 2 Гб; видеоадаптер с минимальным разрешением 720p (1280 на 720 пикселей), стандартная клавиатура, манипулятор "мышь". Для корректной работы программы необходимо наличие установленной библиотеки NetFramework версии 4.0 или выше.

4.4 Вызов и загрузка

Для начала работы программы необходимо открыть файл проекта `windowsforms.exe`. Проект содержится в каталоге *Курсовой проект*.

4.5 Входные данные

Входными данными является текст на английском или русском языке, который необходимо перевести.

4.6 Выходные данные

Выходными данными является переведённый программой текст.

5. ОТЛАДКА И ТЕСТИРОВАНИЕ

В качестве среды разработки была выбрана программа Visual Studio 2010. Программа предоставляет все средства, необходимые при разработке и отладке модулей и программ. В ходе

отладки применялись такие возможности, как точка остановки, выполнение кода по шагам, анализ содержимого локальных и глобальных переменных, анализ содержимого памяти.

Отладчик — это узкоспециализированное средство разработки, которое присоединяется к работающему приложению и позволяет проверять код. В документации по отладке для Visual Studio именно это обычно подразумевается под отладкой. Отладчик является важнейшим инструментом для поиска и устранения ошибок в приложениях.

Тестирование проводилось в рабочем порядке, в процессе разработки, после завершения написания программы.

6. ПРОГРАММА И МЕТОДИКА ИСПЫТАНИЙ

5.1 Объект испытаний

Объектом испытаний является программа *Переводчик текстов*, предназначенная для автоматического лингвистического перевода текстов с английского языка на русский и обратно.

5.2 Требования к программе

Программа должна выполнять все функции, указанные в техническом задании. Работа программы не должна приводить к сбою работы компьютера.

5.3 Требования к программной документации

Состав программной документации, предъявляемой на испытания:

- Техническое задание
- Описание программы
- Программа и методика испытаний
- Описание применения
- Текст программы

5.4 Средства и порядок испытаний

Испытания проводятся на x64-совместимом персональном компьютере, работающем под управлением операционной системы Microsoft Windows версии 10 и имеющем следующие характеристики: тактовая частота процессора – 2,3 ГГц; оперативная память – 4 Гбайт. Для корректной работы программы установлена библиотека NetFramework версии 4.0.

Для проверки правильности работы системы был принят следующий порядок действий:

- а) запуск программы на выполнение;
- б) проведение тестирования программы по тестам, описанным в пункте 5.5;
- в) сравнение результата работы программы с ожидаемым результатом;

г) оценка работоспособности программы по результатам тестирования.

5.5 Методы испытаний

Для контроля корректной работы программы составлены тестовые наборы данных, которые представляет из себя набор действий производимых в программе. Путём сравнения полученного результата с ожидаемыми результатами делается вывод о корректности работы программы.

Тестовые наборы данных представлены в таблицах 1, 2, 3.

Таблица 1 — Тестовый набор данных (текст на английском языке)

Текст	Перевод	Результат
Student	Студент	Рис Б.1

Таблица 2 — Тестовый набор данных (текст на русском языке)

Текст	Перевод	Результат
Студент	Student	Рис Б.2

Таблица 3 — Тестовый набор данных (программа не знает перевода)

Текст	Перевод	Результат
Академия	“Программа не знает перевод данного слова”	Рис Б.3

Результаты испытаний работы программы находятся в Приложении Б.

Изначально, при запуске программы будет загружено окно для ввода текста, который необходимо перевести, выбор направления перевода, окно для вывода переведённого текста, кнопка «Перевести», кнопка «Помощь», кнопка «Открыть словарь».

Для перевода необходимо ввести текст в окно источника перевода и нажать кнопку «Перевести». После этого в окне вывода появится переведённый текст. В случае, если программа не знает перевод какого-либо слова, то на экране появится сообщение об этом и окно для ввода перевода неизвестного программе слова. Если пользователь знает перевод, то ему необходимо ввести перевод и нажать кнопку «Записать», если нет, то нажать кнопку «Не знаю перевода данного слова».

Для открытия словаря необходимо нажать на кнопку «Открыть словарь». После открытия пользователь может дописать в словарь или удалить из словаря слова.

Для вызова помощи, пользователю необходимо нажать кнопку «Помощь».

Для закрытия программы необходимо кликнуть на кнопку «Выход». Ниже на рисунке 7 представлена схема взаимодействия пользователя с программой:

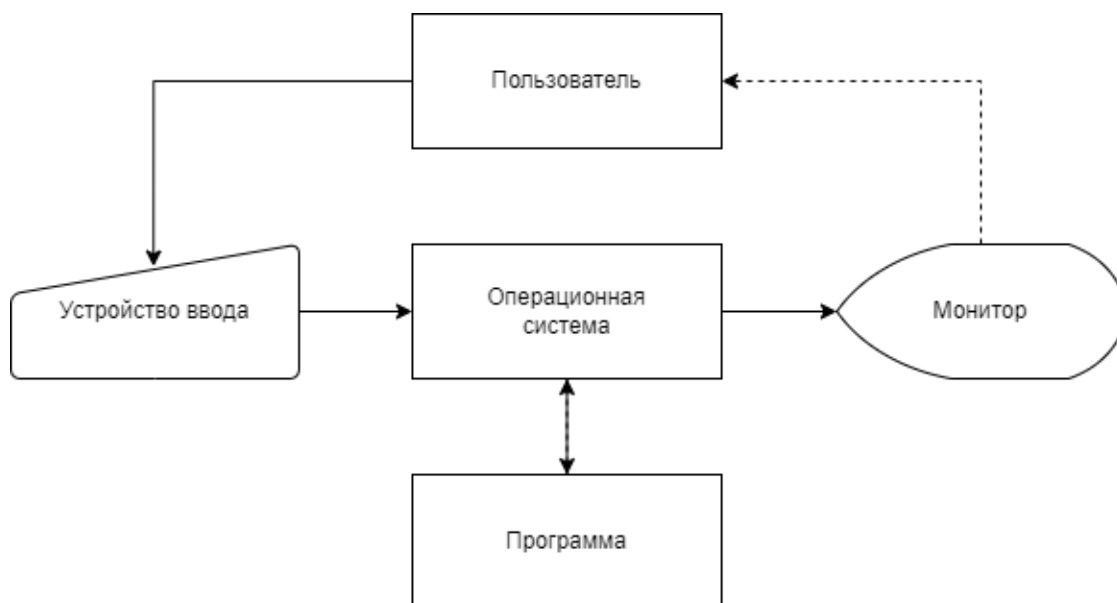


Рис. 7

ЗАКЛЮЧЕНИЕ

В результате выполнения курсового проекта был проанализирован процесс работы переводчика текстов, разработан алгоритм решения поставленной задачи по переводу текста с английского языка на русский и обратно, по разработанному алгоритму была написано приложение, реализующая работу переводчика текстов.

Были проведены испытания программы, которые позволили проверить соответствие работы программы всем требованиям технического задания.

Проведенные испытания показали, что программа удовлетворяет требованиям технического задания.

Оформлена документация, содержащая в себе описание требований к программе, методы тестирования программы, алгоритм решения задачи, описание и результаты тестирования программы, описание работы с программой, текст программы, выводы по проведённой работе, приложения к документации.

За время работы над курсовым проектом был повышен уровень практических навыков программирования на языке *C++* и *Ассемблер*.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Якобсон А., Буч Г., Рамбо Дж. Унифицированный процесс разработки программного обеспечения. – СПб.: Питер, 2002. – 496 с.
2. Секунов Н.Ю. Самоучитель Visual C++ 6. – СПб.: БХВ – Петербург, 2003. – 960 с.
3. Франка П. C++: учебный курс. – СПб.: Питер, 2004. – 522 с.
4. Довбуш Г.Ф. Visual C++ на примерах / Г.Ф. Довбуш, А.Д. Хомоненко / под ред. проф. А.Д. Хомоненко. – СПб.: БХВ – Петербург, 2007. – 528 с.

ТЕКСТ ПРОГРАММЫ

Приложение А

(обязательное)

```

#pragma once

#pragma managed(push, off)

int mystrcat(char* d, char* s)
{
    __asm
    {
        mov esi, s
        mov ecx, -1
        cld
        L : inc ecx //считаем длину строки, которую копируем
        lodsb
        cmp al, 0
        jne L
        mov edi, d
        add edi, ecx //прибавляем эту длину, к исходной строке
        cld
        rep movsb //копируем
    }
}

#pragma managed(pop)

#include "stdafx.h"
#include "iostream"
#include "cstdlib"
#include "iomanip"
#include "conio.h"
#include "windows.h"
#include "stdlib.h"
#include "string"
#include "fstream"
#include "stdio.h"
#include <sstream>
#include "Form2.h"

```

```

#using <System.dll>

namespace Курсовая {

    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;
    using namespace std;

    /// <summary>
    /// Сводка для Form1
    /// </summary>
    public ref class Form1 : public System::Windows::Forms::Form
    {
    public:
        Form1(void)
        {
            InitializeComponent();
            //
            //TODO: добавьте код конструктора
            //
        }

    protected:
        /// <summary>
        /// Освободить все используемые ресурсы.
        /// </summary>
        ~Form1()
        {
            if (components)
            {
                delete components;
            }
        }
    }
}

```

```

private: System::Windows::Forms::Button^ button1;
protected:
private: System::Windows::Forms::TextBox^ textBox1;
private: System::Windows::Forms::TextBox^ textBox2;

private: System::Windows::Forms::RadioButton^ radioButton2;
private: System::Windows::Forms::Label^ label1;
private: System::Windows::Forms::Label^ label2;
private: System::Windows::Forms::Label^ label3;
private: System::Windows::Forms::Label^ label4;

private: System::Windows::Forms::TextBox^ textBox3;
private: System::Windows::Forms::Button^ button2;
private: System::Windows::Forms::Button^ button3;
private: System::Windows::Forms::Button^ button4;
private: System::Windows::Forms::Button^ button5;
private: System::Windows::Forms::RadioButton^ radioButton1;
private: System::Windows::Forms::Button^ button6;

private:
    /// <summary>
    /// Требуется переменная конструктора.
    /// </summary>
    System::ComponentModel::Container^ components;

```

#pragma region Windows Form Designer generated code

```

    /// <summary>
    /// Обязательный метод для поддержки конструктора - не изменяйте
    /// содержимое данного метода при помощи редактора кода.
    /// </summary>
    void InitializeComponent(void)
    {
        this->button1 = (gcnew System::Windows::Forms::Button());
        this->textBox1 = (gcnew System::Windows::Forms::TextBox());
        this->textBox2 = (gcnew System::Windows::Forms::TextBox());
        this->radioButton2 = (gcnew System::Windows::Forms::RadioButton());
        this->label1 = (gcnew System::Windows::Forms::Label());
    }

```

```

this->label2 = (gcnew System::Windows::Forms::Label());
this->label3 = (gcnew System::Windows::Forms::Label());
this->label4 = (gcnew System::Windows::Forms::Label());
this->textBox3 = (gcnew System::Windows::Forms::TextBox());
this->button2 = (gcnew System::Windows::Forms::Button());
this->button3 = (gcnew System::Windows::Forms::Button());
this->button4 = (gcnew System::Windows::Forms::Button());
this->button5 = (gcnew System::Windows::Forms::Button());
this->radioButton1 = (gcnew System::Windows::Forms::RadioButton());
this->button6 = (gcnew System::Windows::Forms::Button());
this->SuspendLayout();

//
// button1
//
this->button1->Font = (gcnew System::Drawing::Font(L"Modern No. 20",
11.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
    static_cast<System::Byte>(0)));
this->button1->Location = System::Drawing::Point(260, 190);
this->button1->Margin = System::Windows::Forms::Padding(4);
this->button1->Name = L"button1";
this->button1->Size = System::Drawing::Size(117, 28);
this->button1->TabIndex = 0;
this->button1->Text = L"Перевести";
this->button1->UseVisualStyleBackColor = true;
this->button1->Click += gcnew System::EventHandler(this,
&Form1::button1_Click);
//
// textBox1
//
this->textBox1->Location = System::Drawing::Point(163, 57);
this->textBox1->Margin = System::Windows::Forms::Padding(4);
this->textBox1->Multiline = true;
this->textBox1->Name = L"textBox1";
this->textBox1->Size = System::Drawing::Size(317, 125);
this->textBox1->TabIndex = 1;
//
// textBox2
//

```

```

this->textBox2->Location = System::Drawing::Point(163, 225);
this->textBox2->Margin = System::Windows::Forms::Padding(4);
this->textBox2->Multiline = true;
this->textBox2->Name = L"textBox2";
this->textBox2->Size = System::Drawing::Size(317, 116);
this->textBox2->TabIndex = 2;
//
// radioButton2
//
this->radioButton2->AutoSize = true;
this->radioButton2->BackColor =
System::Drawing::SystemColors::ActiveCaption;
this->radioButton2->Font = (gcnew System::Drawing::Font(L"Palatino
Linotype", 11.25F, System::Drawing::FontStyle::Bold,
System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(0)));
this->radioButton2->Location = System::Drawing::Point(495, 117);
this->radioButton2->Margin = System::Windows::Forms::Padding(4);
this->radioButton2->Name = L"radioButton2";
this->radioButton2->Size = System::Drawing::Size(236, 30);
this->radioButton2->TabIndex = 4;
this->radioButton2->TabStop = true;
this->radioButton2->Text = L"Русский-Английский";
this->radioButton2->UseVisualStyleBackColor = false;
//
// label1
//
this->label1->AutoSize = true;
this->label1->BackColor = System::Drawing::Color::Transparent;
this->label1->Font = (gcnew System::Drawing::Font(L"Palatino
Linotype", 14.25F, System::Drawing::FontStyle::Bold,
System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(204)));
this->label1->ForeColor =
System::Drawing::SystemColors::ButtonHighlight;
this->label1->Location = System::Drawing::Point(156, 11);
this->label1->Margin = System::Windows::Forms::Padding(4, 0, 4, 0);
this->label1->Name = L"label1";
this->label1->Size = System::Drawing::Size(347, 32);
this->label1->TabIndex = 5;

```

```

        this->label1->Text = L"Введите текст для перевода";
        //
        // label2
        //
        this->label2->AutoSize = true;
        this->label2->BackColor = System::Drawing::Color::Transparent;
        this->label2->Font = (gcnew System::Drawing::Font(L"Palatino
Linotype", 14.25F, System::Drawing::FontStyle::Bold,
System::Drawing::GraphicsUnit::Point,
                static_cast<System::Byte>(204)));
        this->label2->ForeColor =
System::Drawing::SystemColors::ButtonHighlight;
        this->label2->Location = System::Drawing::Point(855, 11);
        this->label2->Margin = System::Windows::Forms::Padding(4, 0, 4, 0);
        this->label2->Name = L"label2";
        this->label2->Size = System::Drawing::Size(79, 32);
        this->label2->TabIndex = 6;
        this->label2->Text = L"label2";
        this->label2->Visible = false;
        //
        // label3
        //
        this->label3->AutoSize = true;
        this->label3->BackColor = System::Drawing::Color::Transparent;
        this->label3->Font = (gcnew System::Drawing::Font(L"Palatino
Linotype", 14.25F, System::Drawing::FontStyle::Bold,
System::Drawing::GraphicsUnit::Point,
                static_cast<System::Byte>(204)));
        this->label3->ForeColor =
System::Drawing::SystemColors::ButtonHighlight;
        this->label3->Location = System::Drawing::Point(939, 46);
        this->label3->Margin = System::Windows::Forms::Padding(4, 0, 4, 0);
        this->label3->Name = L"label3";
        this->label3->Size = System::Drawing::Size(366, 32);
        this->label3->TabIndex = 7;
        this->label3->Text = L"Введите перевод, если знаете";
        this->label3->Visible = false;
        //
        // label4
        //

```

```

        this->label4->AutoSize = true;

        this->label4->BackColor = System::Drawing::Color::Transparent;

        this->label4->Font = (gcnew System::Drawing::Font(L"Palatino
Linotype", 11.25F, System::Drawing::FontStyle::Bold,
System::Drawing::GraphicsUnit::Point,

                static_cast<System::Byte>(204)));

        this->label4->ForeColor =
System::Drawing::SystemColors::ButtonHighlight;

        this->label4->Location = System::Drawing::Point(489, 57);
        this->label4->Margin = System::Windows::Forms::Padding(4, 0, 4, 0);
        this->label4->Name = L"label4";
        this->label4->Size = System::Drawing::Size(330, 26);
        this->label4->TabIndex = 8;
        this->label4->Text = L"Выберите направление перевода";
//
// textBox3
//
        this->textBox3->Location = System::Drawing::Point(945, 81);
        this->textBox3->Margin = System::Windows::Forms::Padding(4);
        this->textBox3->Multiline = true;
        this->textBox3->Name = L"textBox3";
        this->textBox3->Size = System::Drawing::Size(380, 136);
        this->textBox3->TabIndex = 10;
        this->textBox3->Visible = false;
//
// button2
//
        this->button2->Font = (gcnew System::Drawing::Font(L"Modern No. 20",
11.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,

                static_cast<System::Byte>(0)));

        this->button2->Location = System::Drawing::Point(1095, 225);
        this->button2->Margin = System::Windows::Forms::Padding(4);
        this->button2->Name = L"button2";
        this->button2->Size = System::Drawing::Size(100, 28);
        this->button2->TabIndex = 11;
        this->button2->Text = L"Записать";
        this->button2->UseVisualStyleBackColor = true;
        this->button2->Visible = false;

```

```

        this->button2->Click += gcnew System::EventHandler(this,
&Form1::button2_Click);
        //
        // button3
        //
        this->button3->Font = (gcnew System::Drawing::Font(L"Modern No. 20",
11.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(0)));
        this->button3->Location = System::Drawing::Point(988, 261);
        this->button3->Margin = System::Windows::Forms::Padding(4);
        this->button3->Name = L"button3";
        this->button3->Size = System::Drawing::Size(305, 28);
        this->button3->TabIndex = 12;
        this->button3->Text = L"Не знаю перевода данного слова";
        this->button3->UseVisualStyleBackColor = true;
        this->button3->Visible = false;
        this->button3->Click += gcnew System::EventHandler(this,
&Form1::button3_Click);
        //
        // button4
        //
        this->button4->Font = (gcnew System::Drawing::Font(L"Modern No. 20",
11.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(0)));
        this->button4->Location = System::Drawing::Point(260, 350);
        this->button4->Margin = System::Windows::Forms::Padding(4);
        this->button4->Name = L"button4";
        this->button4->Size = System::Drawing::Size(100, 28);
        this->button4->TabIndex = 13;
        this->button4->Text = L"Выход";
        this->button4->UseVisualStyleBackColor = true;
        this->button4->Click += gcnew System::EventHandler(this,
&Form1::button4_Click);
        //
        // button5
        //
        this->button5->Font = (gcnew System::Drawing::Font(L"Modern No. 20",
11.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(0)));
        this->button5->Location = System::Drawing::Point(495, 155);

```

```

this->button5->Margin = System::Windows::Forms::Padding(4);
this->button5->Name = L"button5";
this->button5->Size = System::Drawing::Size(100, 28);
this->button5->TabIndex = 14;
this->button5->Text = L"Помощь";
this->button5->UseVisualStyleBackColor = true;
this->button5->Click += gcnew System::EventHandler(this,
&Form1::button5_Click);
//
// radioButton1
//
this->radioButton1->AutoSize = true;
this->radioButton1->BackColor =
System::Drawing::SystemColors::ActiveCaption;
this->radioButton1->Font = (gcnew System::Drawing::Font(L"Palatino
Linotype", 11.25F, System::Drawing::FontStyle::Bold,
System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(0)));
this->radioButton1->Location = System::Drawing::Point(495, 86);
this->radioButton1->Margin = System::Windows::Forms::Padding(4);
this->radioButton1->Name = L"radioButton1";
this->radioButton1->Size = System::Drawing::Size(236, 30);
this->radioButton1->TabIndex = 3;
this->radioButton1->TabStop = true;
this->radioButton1->Text = L"Английский-Русский";
this->radioButton1->UseVisualStyleBackColor = false;
//
// button6
//
this->button6->Font = (gcnew System::Drawing::Font(L"Modern No. 20",
11.25F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
static_cast<System::Byte>(0)));
this->button6->Location = System::Drawing::Point(603, 155);
this->button6->Margin = System::Windows::Forms::Padding(4);
this->button6->Name = L"button6";
this->button6->Size = System::Drawing::Size(163, 28);
this->button6->TabIndex = 15;
this->button6->Text = L"Открыть словарь";
this->button6->UseVisualStyleBackColor = true;

```

```

        this->button6->Click += gcnew System::EventHandler(this,
&Form1::button6_Click);

        //
        // Form1
        //

        this->AutoScaleDimensions = System::Drawing::SizeF(8, 16);
        this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
        this->BackColor =
System::Drawing::Color::FromArgb(static_cast<System::Int32>(static_cast<System::Byte>(128
)), static_cast<System::Int32>(static_cast<System::Byte>(64)),
        static_cast<System::Int32>(static_cast<System::Byte>(0)));
        this->ClientSize = System::Drawing::Size(1567, 606);
        this->Controls->Add(this->button6);
        this->Controls->Add(this->radioButton1);
        this->Controls->Add(this->button5);
        this->Controls->Add(this->radioButton2);
        this->Controls->Add(this->button4);
        this->Controls->Add(this->button3);
        this->Controls->Add(this->button2);
        this->Controls->Add(this->textBox3);
        this->Controls->Add(this->label4);
        this->Controls->Add(this->label3);
        this->Controls->Add(this->label2);
        this->Controls->Add(this->label1);
        this->Controls->Add(this->textBox2);
        this->Controls->Add(this->textBox1);
        this->Controls->Add(this->button1);
        this->Margin = System::Windows::Forms::Padding(4);
        this->Name = L"Form1";
        this->Text = L"Переводчик текстов";
        this->Load += gcnew System::EventHandler(this, &Form1::Form1_Load);
        this->ResumeLayout(false);
        this->PerformLayout();

    }

    String^ s4; // переменная для записи перевода неизвестного слова
    string SystemToStl(String^ s) // функция для перевода значений переменных
из String^ в string
    {

```

```

        using namespace Runtime::InteropServices;

        const char* ptr = (const
char*)(Marshal::StringToHGlobalAnsi(s)).ToPointer();

        return string(ptr);

    }

#pragma endregion

private: System::Void button1_Click(System::Object^ sender, System::EventArgs^ e)
//кнопка "Перевести"
{
    String^ s;
    string s1, s2, s3, naz;
    string f, n;
    int c;

    if (radioButton1->Checked) // открытие словаря в зависимости от направления
перевода
        naz = "Англо-русский.txt";
    else
        if (radioButton2->Checked)
            naz = "Русско-Английский.txt";
        else
        {
            //выбора перевода
            MessageBox::Show("Выберите направление перевода"); // проверка

            textBox1->Text = "";
            s = "";

        }

    ifstream file(naz); // открытие файла (словаря) для считывания
    s = textBox1->Text;
    s1 = System::ToString(s);
    istream iss(s1);
    while (iss >> s2) // считывание строки по одному слову
    {
        file.clear();
        file.seekg(0);

        if ((radioButton1->Checked) & ((s2[0] >= 'A') & (s2[0] <= 'Я'))) //
проверка соответствия направления перевода и введённого текста
        {
            //s3 = s3 + " " + s2;

```

```

char *_s3, *_s2;
for (int i = 0; i < s3.size(); i++)
{
    _s3[i] = s3[i];
}

s3 = mystrcat(_s3, " ");

for (int i = 0; i < s3.size(); i++)
{
    _s3[i] = s3[i];
}

for (int i = 0; i < s2.size(); i++)
{
    _s2[i] = s2[i];
}

s3 = mystrcat(_s3, _s2);

continue;
}
if ((radioButton2->Checked) & ((s2[0] >= 'A') & (s2[0] <= 'z')))
{
    s3 = s3 + " " + s2;
    continue;
}
if (radioButton1->Checked & (s2[0] < 'A' || s2[0] > 'z')) // вывод
символов, не являющихся буквами
{
    s3 = s3 + " " + s2;
    continue;
}
else
{
    if (radioButton2->Checked & (s2[0] < 'A' || s2[0] > 'я'))
    {
        s3 = s3 + " " + s2;
    }
}

```

```

        continue;
    }
}
do
{
    c = 0;
    file >> f; // считывание слова из файла
    if (((('A' <= f[0]) && (f[0] <= 'Z')) || (('A' <= f[0]) &&
(f[0] <= 'Я')))) //сравнение больших и малых букв
        f[0] = f[0] + 32;
    if (((('A' <= s2[0]) && (s2[0] <= 'Z')) || (('A' <= s2[0]) &&
(s2[0] <= 'Я'))))
        s2[0] = s2[0] + 32;
    if ((s2 == f) & (f != " ")) //сравнение введённого слова и
слова, считанного из словаря
    {
        c = 1;
        file >> f; //считывание перевода из файла
        s3 = s3 + " " + f;
        break;
    }
} while (!file.eof()); //цикл(пока не конец файла)
if (c == 0)
{
    s4 = gcnew System::String(s2.c_str());
    label2->Text = "К сожалению в словаре нет перевода слова " +
"" + s4 + ""; //программа не знает перевода какого-либо слова
    textBox3->Visible = true; //появление объектов на форме
    label2->Visible = true;
    label3->Visible = true;
    button2->Visible = true;
    button3->Visible = true;
}
}
textBox2->Text = gcnew System::String(s3.c_str()); //вывод переведённого
текста
}

private: System::Void button2_Click(System::Object^ sender, System::EventArgs^ e)
//кнопка "Записать"

```

```

{
    String^ s;
    string s1, s3, naz, naz1;
    if (radioButton1->Checked) //открытие словаря в зависимости от направления
перевода
    {
        naz = "Англо-русский.txt";
        naz1 = "Русско-Английский.txt";
    }
    if (radioButton2->Checked)
    {
        naz = "Русско-Английский.txt";
        naz1 = "Англо-русский.txt";
    }
    s = textBox3->Text;
    ofstream file(naz, ios_base::app); //открытие файла(словаря) для дозаписи
    ofstream file1(naz1, ios_base::app); //открытие файла(словаря) для дозаписи
    s1 = SystemToStl(s);
    s3 = SystemToStl(s4);
    file.eof(); //переход в конец файла
    file << "\n" << s3; //запись неизвестного программе слова
    file << ' ' << s1; //запись перевода неизвестного программе слова
    file1.eof(); //переход в конец файла
    file1 << "\n" << s1; //запись перевода неизвестного программе слова
    file1 << ' ' << s3; //запись неизвестного программе слова
    textBox3->Text = "";
    textBox3->Visible = false; //исчезновение объектов с формы
    label2->Visible = false;
    label3->Visible = false;
    button2->Visible = false;
    button3->Visible = false;
}

private: System::Void button3_Click(System::Object^ sender, System::EventArgs^ e)
//кнопка "Не знаю перевода данного слова"
{
    textBox3->Visible = false; //исчезновение объектов с формы
    label2->Visible = false;
    label3->Visible = false;
}

```

```

        button2->Visible = false;
        button3->Visible = false;
    }

    private: System::Void button4_Click(System::Object^ sender, System::EventArgs^ e)
//кнопка "Выход"
    {
        Form1::Close(); //закрытие формы
    }

    private: System::Void button5_Click(System::Object^ sender, System::EventArgs^ e)
//кнопка "Помощь"
    {
        Form2^ secondForm = gcnew Form2(); //переход на вторую форму
        secondForm->ShowDialog();
    }

    private: System::Void button6_Click(System::Object^ sender, System::EventArgs^ e)
//кнопка "Открыть словарь"
    {
        string naz;
        if (radioButton1->Checked) // открытие словаря в зависимости от направления
перевода
            system("notepad.exe Англо-русский.txt");
        else if (radioButton2->Checked)
            system("notepad.exe Русско-Английский.txt");
        else
        {
            MessageBox::Show("Выберите направление перевода"); // проверка выбора
направления перевода
        }
    }

    private: System::Void Form1_Load(System::Object^ sender, System::EventArgs^ e) {
    }
};
}

```

#pragmaonce

```

#include"stdafx.h"
#include"iostream"
#include"cstdlib"
#include"iomanip"
#include"conio.h"
#include"windows.h"
#include"stdlib.h"
#include"string"
#include"fstream"
#include"stdio.h"
#include<sstream>

namespaceКурсовая {

    usingnamespace System;
    usingnamespaceSystem::ComponentModel;
    usingnamespaceSystem::Collections;
    usingnamespaceSystem::Windows::Forms;
    usingnamespaceSystem::Data;
    usingnamespaceSystem::Drawing;
    usingnamespace std;
    /// <summary>
    /// Сводкадля Form2
    /// </summary>
    publicrefclass Form2 :public System::Windows::Forms::Form
    {
    public:
        Form2(void)
        {
            InitializeComponent();
            //
            //TODO: добавьтекодконструктора
            //
        }

    protected:

```

```

    /// <summary>
    /// Освободить все используемые ресурсы.
    /// </summary>
    ~Form2()
    {
        if (components)
        {
            delete components;
        }
    }

private: System::Windows::Forms::Label^ label1;
protected:
private: System::Windows::Forms::Button^ button1;

private:
    /// <summary>
    /// Требуется переменная конструктора.
    /// </summary>
    System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
    /// <summary>
    /// Обязательный метод для поддержки конструктора - не изменяйте
    /// содержимое данного метода при помощи редактора кода.
    /// </summary>
    void InitializeComponent(void)
    {
        System::ComponentModel::ComponentResourceManager^ resources = (gcnew
System::ComponentModel::ComponentResourceManager(Form2::typeid));

        this->label1 = (gcnew System::Windows::Forms::Label());
        this->button1 = (gcnew System::Windows::Forms::Button());
        this->SuspendLayout();

        //
        // label1
        //
        this->label1->AutoSize = true;
        this->label1->BackColor = System::Drawing::Color::Transparent;

```

```

        this->label1->Font = (gcnew System::Drawing::Font(L"Palatino
Linotype", 12, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->label1->ForeColor = System::Drawing::SystemColors::Window;
        this->label1->Location = System::Drawing::Point(131, 37);
        this->label1->Name = L"label1";
        this->label1->Size = System::Drawing::Size(0, 22);
        this->label1->TabIndex = 0;
        //
        // button1
        //
        this->button1->Font = (gcnew System::Drawing::Font(L"Palatino
Linotype", 9.75F, System::Drawing::FontStyle::Bold, System::Drawing::GraphicsUnit::Point,
        static_cast<System::Byte>(204)));
        this->button1->Location = System::Drawing::Point(806, 300);
        this->button1->Name = L"button1";
        this->button1->Size = System::Drawing::Size(157, 25);
        this->button1->TabIndex = 1;
        this->button1->Text = L"Вернутьсяобратно";
        this->button1->UseVisualStyleBackColor = true;
        this->button1->Click += gcnew System::EventHandler(this,
&Form2::button1_Click);
        //
        // Form2
        //
        this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
        this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;
        this->BackgroundImage = (cli::safe_cast<System::Drawing::Image^
>(resources->GetObject(L"$this.BackgroundImage"))));
        this->ClientSize = System::Drawing::Size(975, 497);
        this->Controls->Add(this->button1);
        this->Controls->Add(this->label1);
        this->Name = L"Form2";
        this->Text = L"Помощь";
        this->Load += gcnew System::EventHandler(this, &Form2::Form2_Load);
        this->ResumeLayout(false);
        this->PerformLayout();
    }

```

```

#pragma endregion

    private: System::Void button1_Click(System::Object^ sender, System::EventArgs^
e) // кнопка "Вернуться обратно"
    {
        Form2::Close (); // кнопка "Закрытие формы"
    }

    private: System::Void Form2_Load(System::Object^ sender, System::EventArgs^ e)
// помощь
    {
        label1->Text="Данная программа предназначена для перевода
текста\nДля перевода вам необходимо выполнить следующие действия:\n1)Выбрать тип
перевода:\n    а)С Английского языка на Русский\n    б)С Русского языка на Английский\n2)
Ввести слово
или текст, который вы хотите перевести в верхнее окно на экране\n3) Нажать кнопку
'Перевести'\n4)На экране в нижнем окне появится перевод\n5)В случае ввода неизвестного
слова, программа попросит вас ввести перевод, если он вам известен"; // Вывод помощи
    }

};

}

```


Приложение Б

(обязательное)

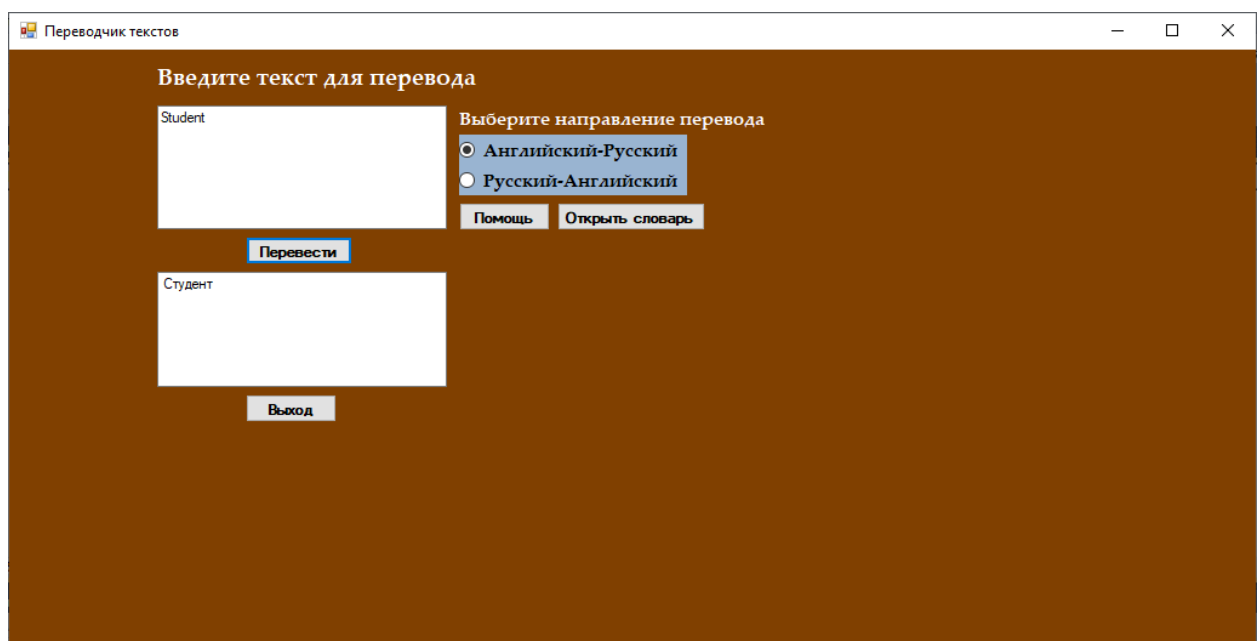


Рисунок Б1. Тестовый вариант (текст на английском языке) №1

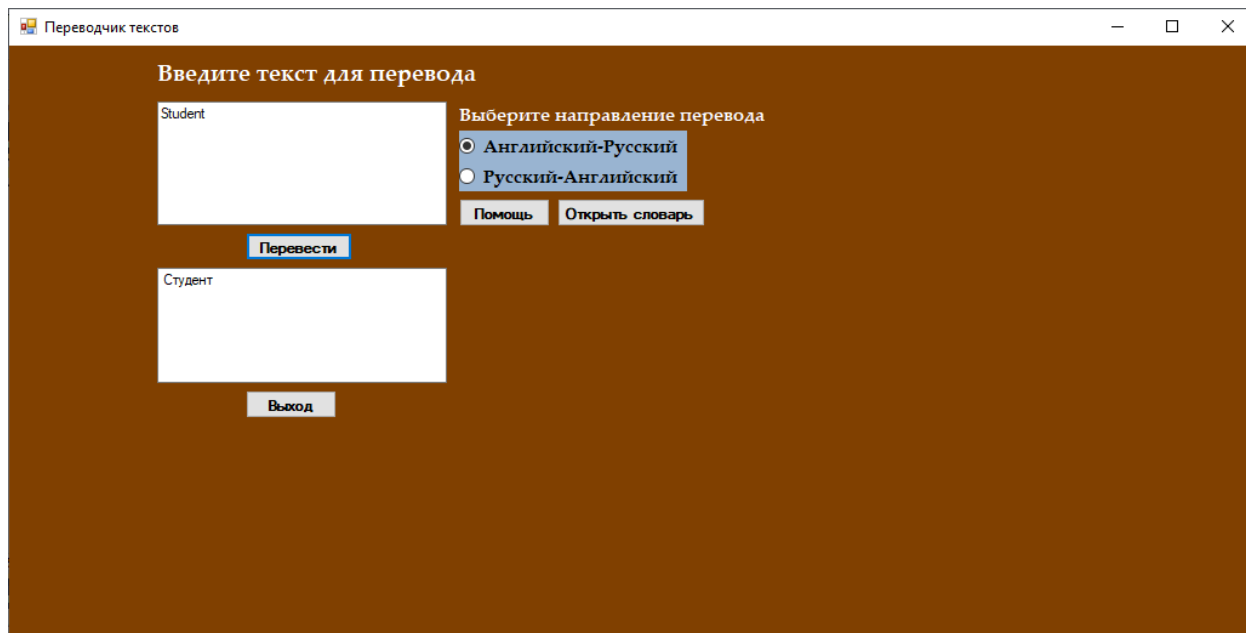


Рисунок Б2. Тестовый вариант (текст на русском языке) №2

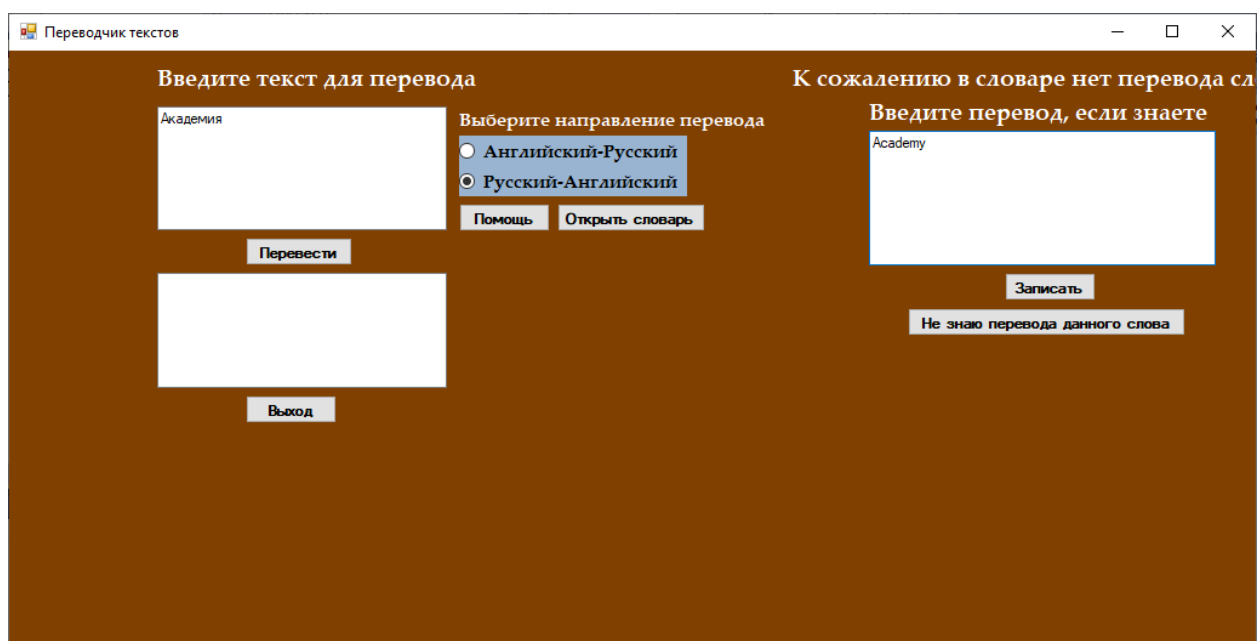


Рисунок Б3. Тестовый вариант (программа не знает перевода слова) №3

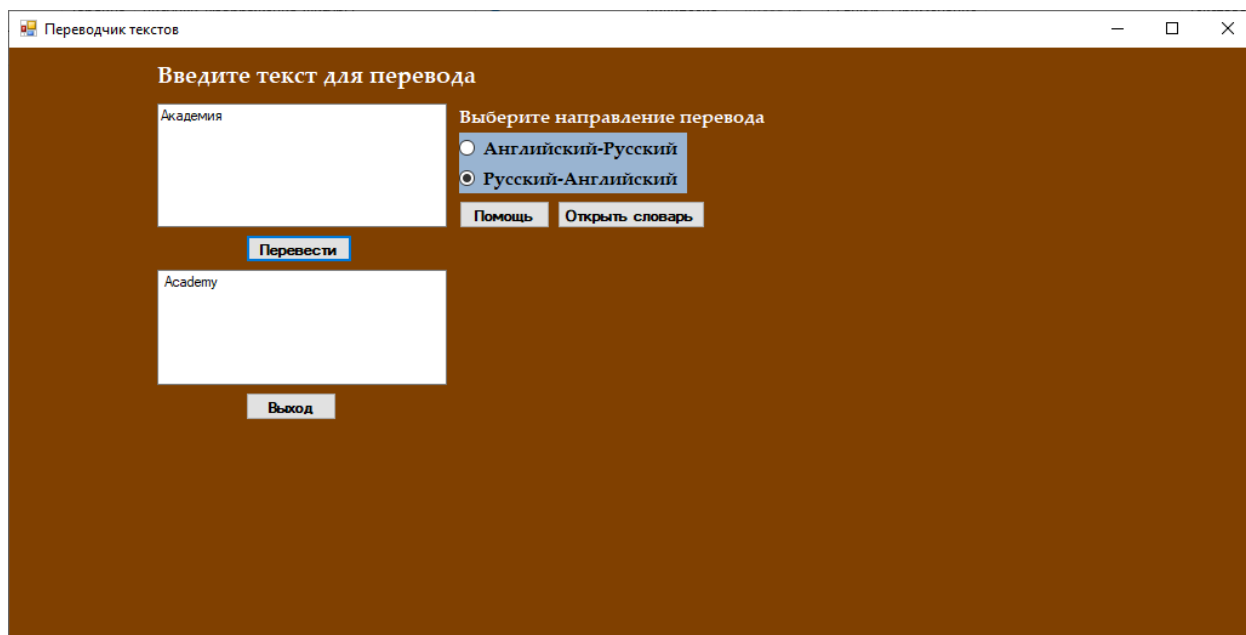


Рисунок Б4. Тестовый вариант (перевод после записи в словарь) №3

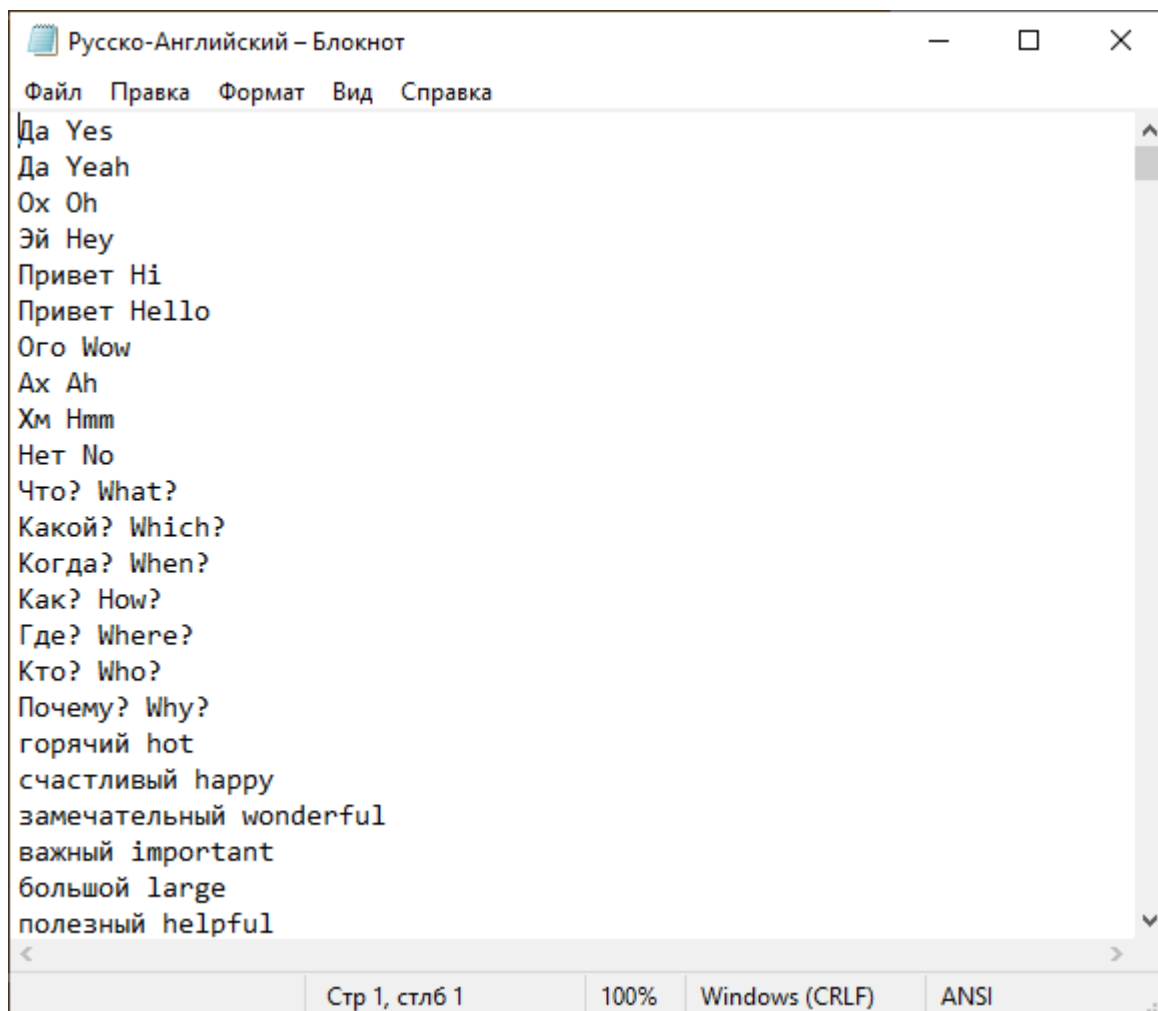


Рисунок Б5. Открытие словаря

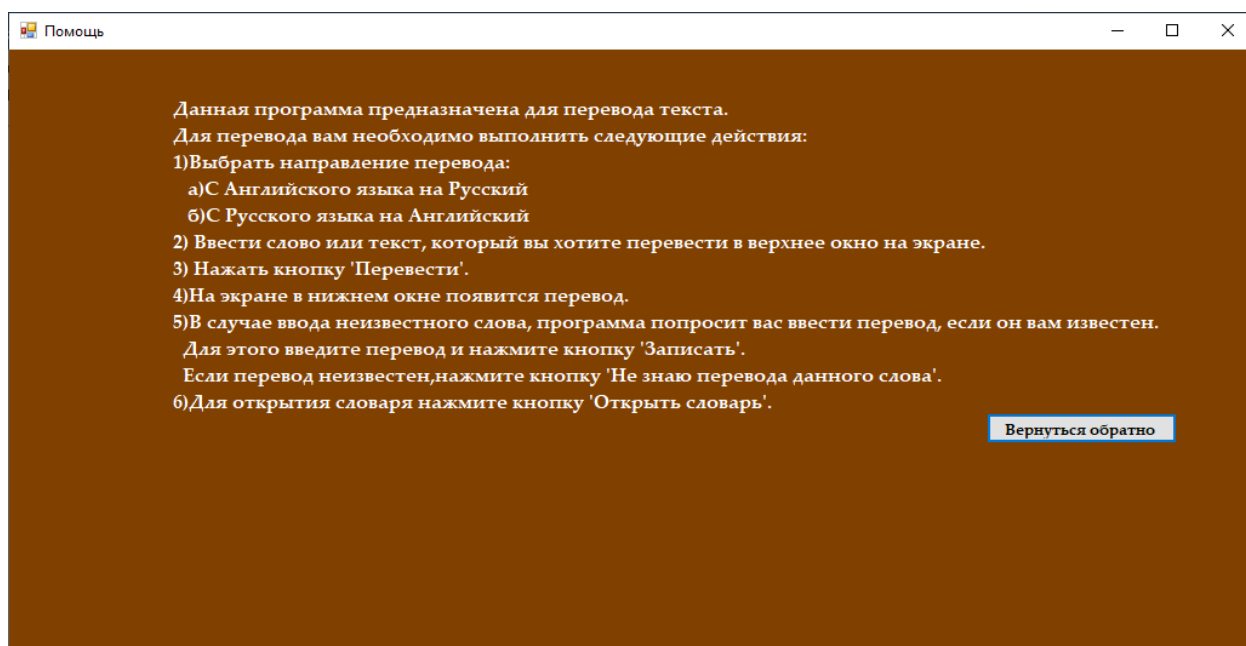


Рисунок Б6. Вызов помощи