МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования  
«Гомельский государственный технический университет имени П.О. Сухого»

Факультет автоматизированных и информационных систем

Кафедра «Информационные технологии»

Специальность 1-40 04 01 «Информатика и технологии программирования»

Отчет  
 по лабораторной работе №1  
по дисциплине «Методы трансляции»  
на тему: «Строки языка»

Составил: студент гр. ИП-31 22.02.2023 Коваленко А.И.

(подпись, дата)

Принял доцент кафедры ИТ Кравченко О.А.

(подпись, дата)

Гомель 2023

Лабораторная работа №1

Строки языка

*Цель работы: усвоить операции обработки строк языка*

*Задания*:

1. Разработать алгоритм, написать и отладить программу, определяющую принадлежность строки *S* языку *L*.

Программа (с графическим интерфейсом) должна обеспечивать чтение из текстового файла и с консоли символов заданного языка (алфавит языка должен содержать не менее семи символов) и при вводе с консоли некоторой строки выдавать ответ, принадлежит ли эта строка языку или нет. Например, для языка *L=*{*c*, *i*, *n*, *s*, *r*, *t*, *u*, *o*} при вводе строки *construction* ответ – «Принадлежит», а для строки *structure* – «Не принадлежит».

2. Разработать алгоритм, написать и отладить программу получения всех правильных префиксов, суффиксов и подстрок строки *S* языка *L*.

Программа должна обеспечивать чтение из текстового файла и с консоли символов заданного языка (не менее семи символов) и одной строки (*S*) этого языка. Графический интерфейс программы должен предоставлять пользователю возможность выбрать один из трех вариантов: префиксы, суффиксы и подстроки. Для каждого из указанных вариантов программа должна выводить на консоль и в текстовый файл все правильные ответы в виде строк (все префиксы, все суффиксы или все подстроки строки *S*).

3. Разработать алгоритм, написать и отладить программу определения наличия и вида подпоследовательности *S*1 в строке *S* языка *L*.

Программа (с графическим интерфейсом) должна обеспечивать чтение из текстового файла и с консоли символов заданного языка (не менее семи символов), и двух строк этого языка (*S* и *S*1). Возможные результаты выполнения программы:

* вторая подстрока (*S*1) не является подпоследовательностью первой строки (*S*);
* вторая подстрока (*S*1) является подпоследовательностью первой строки (*S*);
* вторая подстрока (*S*1) является префиксом первой строки (*S*);
* вторая подстрока (*S*1) является суффиксом первой строки (*S*);
* вторая подстрока (*S*1) является подстрокой первой строки (*S*).

**Тесты:**

Таблица 1 – Тестирование задания 1

|  |  |  |  |
| --- | --- | --- | --- |
| **Строка S** | **Алфавит языка L** | **Ожидаемый результат** | **Результат** |
| 1 | 2 | 3 | 4 |
| nastya | a b c d r f g h I j k | Не принадлежит | Не принадлежит |
| graf | a b c d r f g h I j k n | Принадлежит | Принадлежит |
| граф | а б в г д е ж з и | Не принадлежит | Не принадлежит |
| строка | с м в т и р ш о ж к ч а | Принадлежит | Принадлежит |
| 123 | 1 2 3 4 5 6 7 | Принадлежит | Принадлежит |

Таблица 2 – Тестирование задания 2

|  |  |  |  |
| --- | --- | --- | --- |
| **Строка S** | **Префиксы** | **Суффиксы** | **Подстроки** |
| 1 | 2 | 3 | 4 |
| graf | gr  gra  graf | f  af  raf  graf | g  gr  gra  r  ra  raf  a  af  f |
| строка | ст  стр  стро  строк  строка | а  ка  ока  рока  трока  строка | с  ст  стр  стро  строк  т  тр  тро  трок  трока  р  ро  рок  рока  о  ок  ока  к  ка  а |

Продолжение таблицы 2 – Тестирование задания 2

|  |  |  |  |
| --- | --- | --- | --- |
| 1 | 2 | 3 | 4 |
| 123 | 12  123 | 3  23  123 | 1  12  2  23  3 |

Таблица 3 – Тестирование задания 3

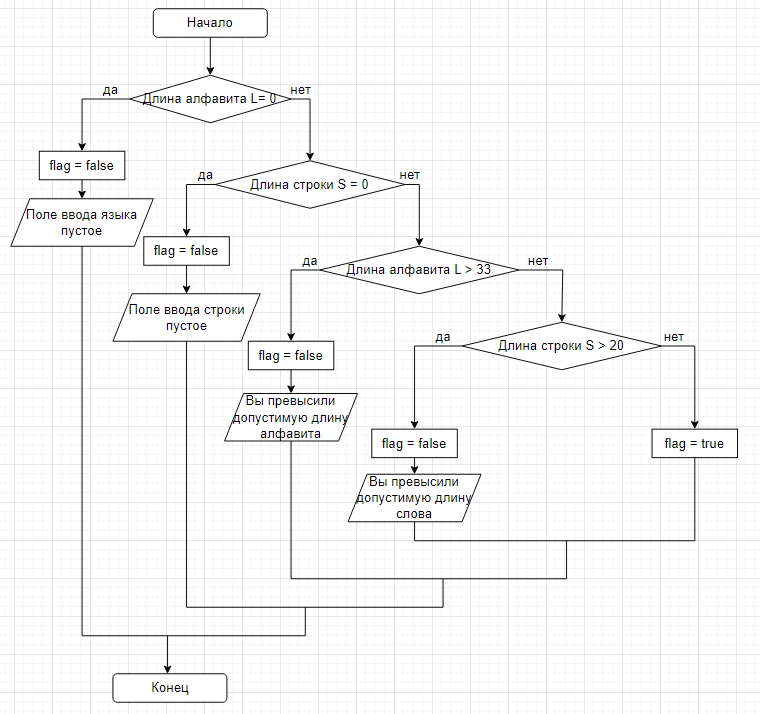
|  |  |  |
| --- | --- | --- |
| **Строка S1** | **Строка S2** | **Результат** |
| winner | win | Вторая подстрока S2 является префиксом первой строки S1.  Вторая подстрока S2 является подстрокой первой строки S1.  Вторая подстрока S2 является подпоследовательностью первой строки S1 |
| строка | строка | Вторая подстрока S2 является суффиксом первой строки S1.  Вторая подстрока S2 является префиксом первой строки S1.  Вторая подстрока S2 является подпоследовательностью первой строки S1 |

**Графические схемы алгоритмов:**

**Задание 1:**

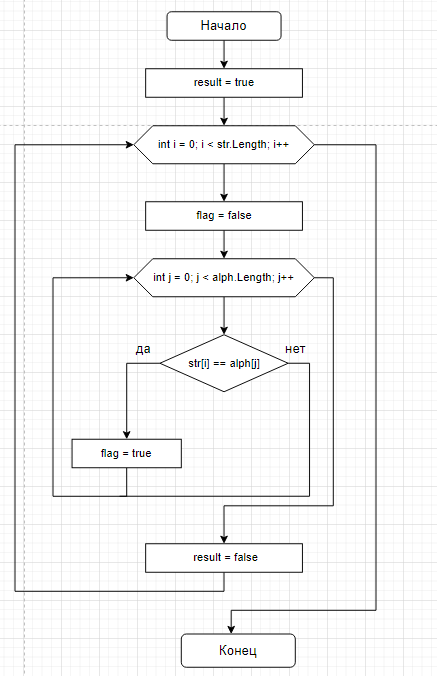
В графической схеме 1 представлен алгоритм проверки корректности введенных данных.

**Графическая схема 1**



В графической схеме 2 представлен алгоритм функции определения принадлежности введённой строки S языку L. Функция принимает на входе S и L типа string, на выходе возвращает результат типа bool.

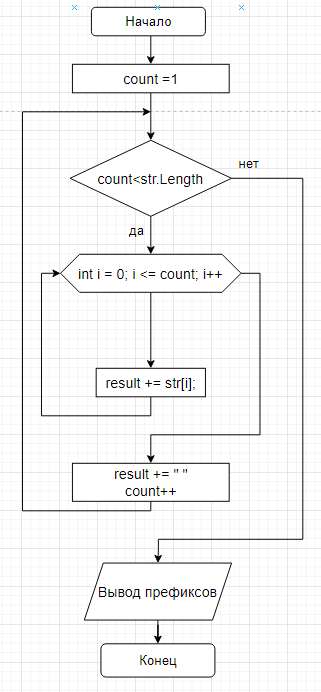
**Графическая схема 2**



**Задание 2:**

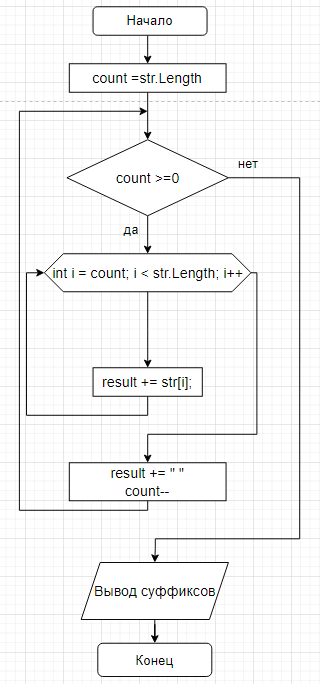
В графической схеме 3 представлен алгоритм определения всех префиксов введённой строки S.

**Графическая схема 3**



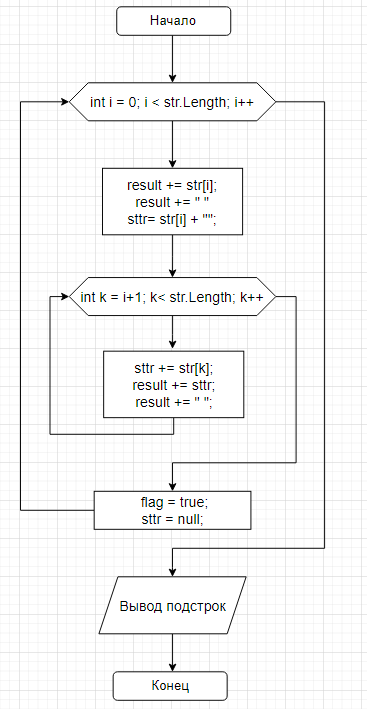
В графической схеме 4 представлен алгоритм определения всех суффиксов введённой строки S.

**Графическая схема 4**



В графической схеме 5 представлен алгоритм определения всех подстрок введённой строки S.

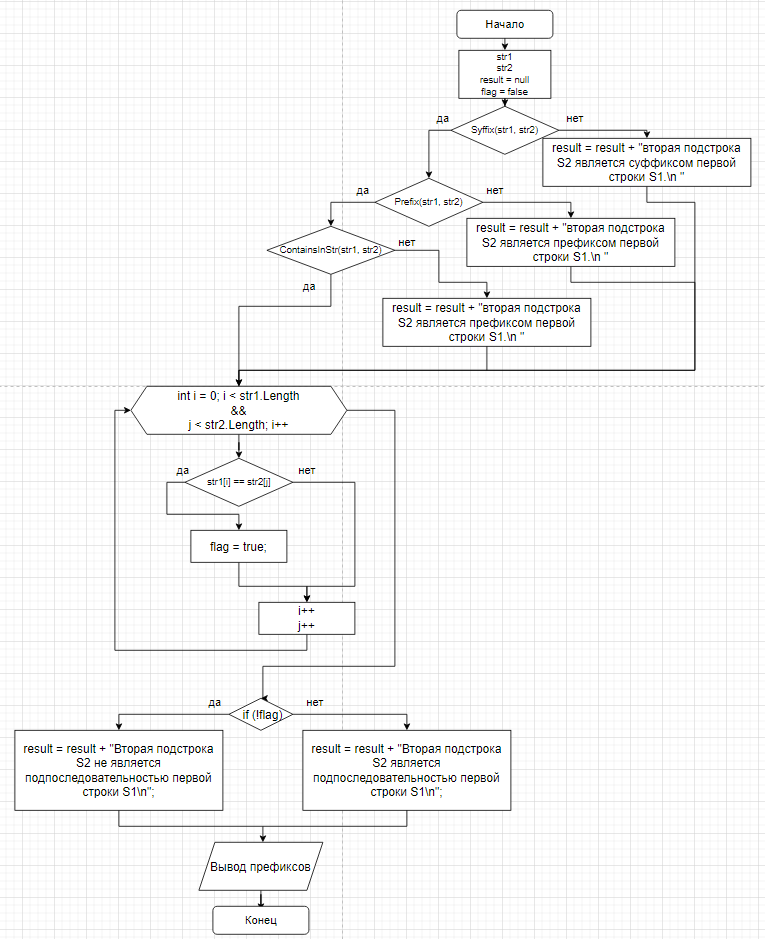
**Графическая схема 5**



**Задание 3:**

В графической схеме 6 представлен алгоритм определения является ли подстрока (*S*1) суффиксом, префиксом, подстрокой и подпоследовательностью строки (*S*2).

**Графическая схема 6**



**Результат выполнения программы:**

На рисунке 1 представлен результат определения принадлежности введённой строки S языку L.

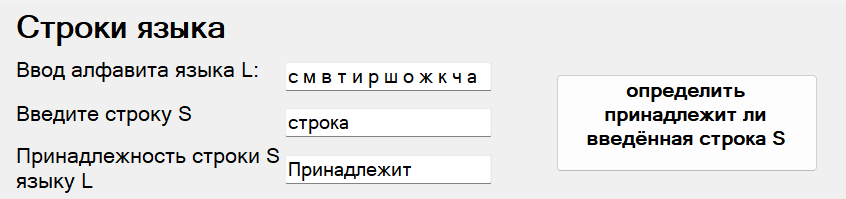


Рисунок 1 – Определение принадлежности введённой строки S языку L.

На рисунке 2 представлен результат определения всех префиксов введённой строки S.

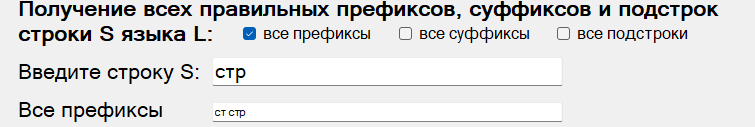


Рисунок 2 – Определение всех префиксов введённой строки S.

На рисунке 3 представлен результат определения всех суффиксов введённой строки S.

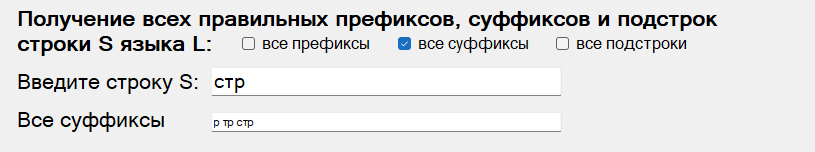


Рисунок 3 – Определение всех суффиксов введённой строки S.

На рисунке 4 представлен результат определения всех подстрок введённой строки S.

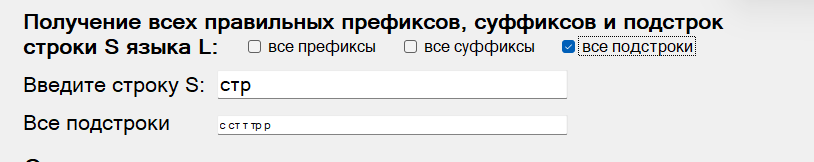


Рисунок 4 – Определение всех подстрок введённой строки S.

На рисунке 5 представлен результат определения является ли подстрока (*S*1) суффиксом, префиксом, подстрокой и подпоследовательностью строки (*S*2).

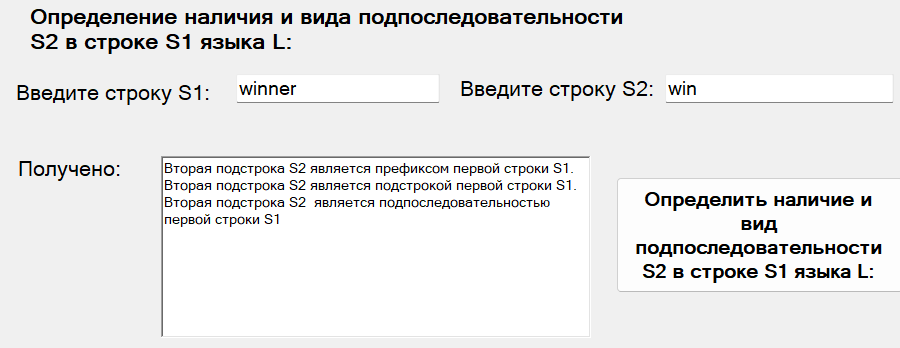


Рисунок 5 – Определение является ли подстрока S2 суффиксом, подстрокой, префиксом и подпоследовательностью строки S1.

**Листинг программы:**

**Form1.cs**

// Разработал студент ИП-31 Коваленко Анастасия Игоревна. Дата рождения: 12.02.2003. Место рождения: город Витебск

using System;

using System.Windows.Forms;

namespace first

{

public partial class Form1 : Form

{

public string alphabet; // алфавит языка

public Form1()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

alphabet = textBox1.Text; //получаем данные о введенном алфавите

string inputS = textBox2.Text; //получаем данные о введенной строке

textBox3.Clear();

try

{

if (alphabet == "") //проверка на пустое поле алфавита

throw new Exception("Поле ввода языка пустое");

if (inputS == "") //проверка на пустое поле строки

throw new Exception("Поле ввода строки пустое");

if (alphabet.Length > 33) //проверка на длину алфавита

throw new Exception("Вы превысили допустимую длину алфавита");

if (inputS.Length > 20)//проверка на длину слова

throw new Exception("Вы превысили допустимую длину слова");

else //проверка на принадлежность

{

if (ChekingForTask1(alphabet, inputS)) textBox3.Text = "Принадлежит";//вызов функции проверки вхождения слова в алфавит

else textBox3.Text = "Не принадлежит";

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

//Функция проверки вхождения слова в алфавит

//alph - входной параметр функции, для работы со значением алфавита

//str - входной параметр функции, для работы со значением строки

static public bool ChekingForTask1(string alph, string str)

{

bool result = true; //переменная для передачи результата, true - принадлежит, false - не принадлежит

//проверяет каждый символ из строки S для определения принадлежности к алфавиту

for (int i = 0; i < str.Length; i++)

{

bool flag = false;

for (int j = 0; j < alph.Length; j++)

{

if (str[i] == alph[j]) flag = true;

}

if (flag == false) result = false;

}

return result;

}

private void Form1\_Load(object sender, EventArgs e)

{

}

/// Обработчик события при нажатии на checkBox "Все префиксы" для проверки принадлежности S алфавиту

/// Данная функция анализирует введенные алфавит и строку, перебирает все варианты префиксов в строки на префиксы

private void checkBox1\_CheckedChanged(object sender, EventArgs e)

{

checkBox2.Checked = false;

checkBox3.Checked = false;

textBox5.Clear();

string str = textBox4.Text;

string result = null;

textBox5.WordWrap = true;

alphabet = textBox1.Text;

try

{

if (alphabet == "")

throw new Exception("Поле ввода языка пустое");

if (alphabet.Length > 40)

throw new Exception("Вы превысили допустимую длину языка");

if (str.Length == 0)

throw new Exception("Поле ввода строки пустое");

if(str.Length > 15)

throw new Exception("Вы превысили допустимую длину строки");

else

{

//// проверка принадлежности строки алфавиту

if (!ChekingForTask1(alphabet, str))

throw new Exception("Строка не принадлежит алфавиту языка");

else

{

////делим строку на префиксы

int count = 1;

while (count<str.Length)

{

for (int i = 0; i <= count; i++)

{

result += str[i];

}

result += " ";

count++;

}

textBox5.Text = result;

label5.Text = "Все префиксы";

}

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

/// Обработчик события при нажатии на checkBox "Все суффиксы" для проверки принадлежности S алфавиту

/// Данная функция анализирует введенные алфавит и строку, перебирает все варианты суффиксов строки

private void checkBox2\_CheckedChanged(object sender, EventArgs e)

{

checkBox1.Checked = false;

checkBox3.Checked = false;

textBox5.Clear();

string str = textBox4.Text;

string result = null;

textBox5.WordWrap = true;

alphabet = textBox1.Text;

try

{

if (alphabet == "")

throw new Exception("Поле ввода языка пустое");

if (alphabet.Length > 40)

throw new Exception("Вы превысили допустимую длину языка");

if (str.Length == 0)

throw new Exception("Поле ввода строки пустое");

if (str.Length > 15)

throw new Exception("Вы превысили допустимую длину строки");

else

{

//// проверка принадлежности строки алфавиту

bool flag1 = false;

if (!ChekingForTask1(alphabet, str))

throw new Exception("Строка не принадлежит алфавиту языка");

else

{

int count = str.Length;

////делим строку на суффиксы

while (count >=0)

{

for (int i = count; i < str.Length; i++)

{

result += str[i];

}

if (flag1)

result += " ";

count--;

flag1 = true;

}

textBox5.Text = result;

label5.Text = "Все суффиксы";

}

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

/// Обработчик события при нажатии на checkBox "Все подстроки" для проверки принадлежности S алфавиту

/// Данная функция анализирует введенные алфавит и строку, перебирает все варианты подстрок строки

private void checkBox3\_CheckedChanged(object sender, EventArgs e)

{

checkBox1.Checked = false;

checkBox2.Checked = false;

textBox5.Clear();

string str = textBox4.Text;

string result = null;

textBox5.WordWrap = true;

alphabet = textBox1.Text;

try

{

if (alphabet == "")

throw new Exception("Поле ввода языка пустое");

if (alphabet.Length > 40)

throw new Exception("Вы превысили допустимую длину языка");

if (str.Length == 0)

throw new Exception("Поле ввода строки пустое");

if (str.Length > 15)

throw new Exception("Вы превысили допустимую длину строки");

else

{

//// проверка принадлежности строки алфавиту

if (!ChekingForTask1(alphabet, str))

throw new Exception("Строка не принадлежит алфавиту языка");

else

{

string sttr = null;

bool flag = false;

///циклы для перебора всех подстрок строки

for (int i = 0; i < str.Length; i++)

{

result += str[i];

result += " ";

sttr = str[i] + "";

for (int k = i+1; k < str.Length; k++)

{

if (k == str.Length - 1 && !flag)

break;

sttr += str[k];

result += sttr;

result += " ";

}

flag = true;

sttr = null;

}

textBox5.Text = result;

label5.Text = "Все подстроки";

}

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

/// Обработчик события при нажатии на кнопку "Определить" для определения вида подпоследовательности S2 в строке S1

/// Данная функция анализирует введенные алфавит и строку, определяет виды подпоследовательностей

private void button4\_Click(object sender, EventArgs e)

{

richTextBox1.Clear();

string str1 = textBox7.Text;

string str2 = textBox6.Text;

alphabet = textBox1.Text;

{

/// анализ строки

if (alphabet == "")

throw new Exception("Поле ввода языка пустое");

if (alphabet.Length > 40)

throw new Exception("Вы превысили допустимую длину языка");

if (str1.Length == 0)

throw new Exception("Поле ввода строки S1 пустое");

if (str1.Length > 15)

throw new Exception("Вы превысили допустимую длину строки S1");

if (str2.Length == 0)

throw new Exception("Поле ввода строки S2 пустое");

if (str2.Length > 15)

throw new Exception("Вы превысили допустимую длину строки S2");

else

{

// проверка принадлежности строк алфавиту

if (!ChekingForTask1(alphabet, str1))

throw new Exception("Строка S1 не принадлежит алфавиту языка");

if (!ChekingForTask1(alphabet, str2))

throw new Exception("Строка S2 не принадлежит алфавиту языка");

else if (str1.Length<str2.Length)

{

MessageBox.Show("Вторая строка не является подстрокой первой строки\n");

textBox7.Clear();

textBox6.Clear();

}

else

{

string result = null;

/// проверка, является ли S2 суффиксом

if (Syffix(str1, str2))

result += "Вторая подстрока S2 является суффиксом первой строки S1.\n";

/// проверка, является ли S2 префиксом

if (Prefix(str1, str2))

result = result + "Вторая подстрока S2 является префиксом первой строки S1.\n";

/// проверка, является ли S2 подстрокой

if (ContainsInStr(str1, str2))

result = result + "Вторая подстрока S2 является подстрокой первой строки S1.\n";

bool flag = false;

int j = 0;

/// цикл для проверки, является ли S2 подполедовательностью строки S1

for (int i = 0; i < str1.Length && j < str2.Length; i++)

{

if (str1[i] == str2[j])

{

flag = true;

j++;

}

else

flag = false;

}

if (!flag)

result = result + "Вторая подстрока S2 не является подпоследовательностью первой строки S1\n";

else

result = result + "Вторая подстрока S2 является подпоследовательностью первой строки S1\n";

richTextBox1.Text = result;

}

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

//функция проверки, является ли S2 суффиксом S1

public bool Syffix(string s1, string s2)

{

int j = s1.Length -1;

bool flag = true;

int count = s1.Length - 1;

for (int i = s2.Length -1; i >=0; i--)

{

if (s1[j] != s2[i])

flag = false;

j--;

}

return flag;

}

//функция проверки, является ли S2 префиксом S1

public bool Prefix(string s1, string s2)

{

bool flag = true;

for (int i = 0; i < s2.Length; i++)

{

if (s1[i] != s2[i])

flag = false;

}

return flag;

}

/// функция проверки, является ли S2 подстрокой S1

public bool ContainsInStr(string s1, string s2)

{

if (s1.Contains(s2))

return true;

else

return false;

}

}

}

**Вывод:** в ходе данной лабораторной работы были усвоены операции над языками в лексическом анализе, термины для частей строк.