Федеральное государственное автономное образовательное учреждение высшего образования «Пермский национальный исследовательский политехнический университет», ПНИПУ

ЛАБОРАТОРНАЯ РАБОТА

СТРОКИ

Выполнил: студент группы РИС-23-3б

Артем Владимирович Швецов

Проверила: доцент кафедры ИТАС

Ольга Андреевна Полякова

Пермь 2023

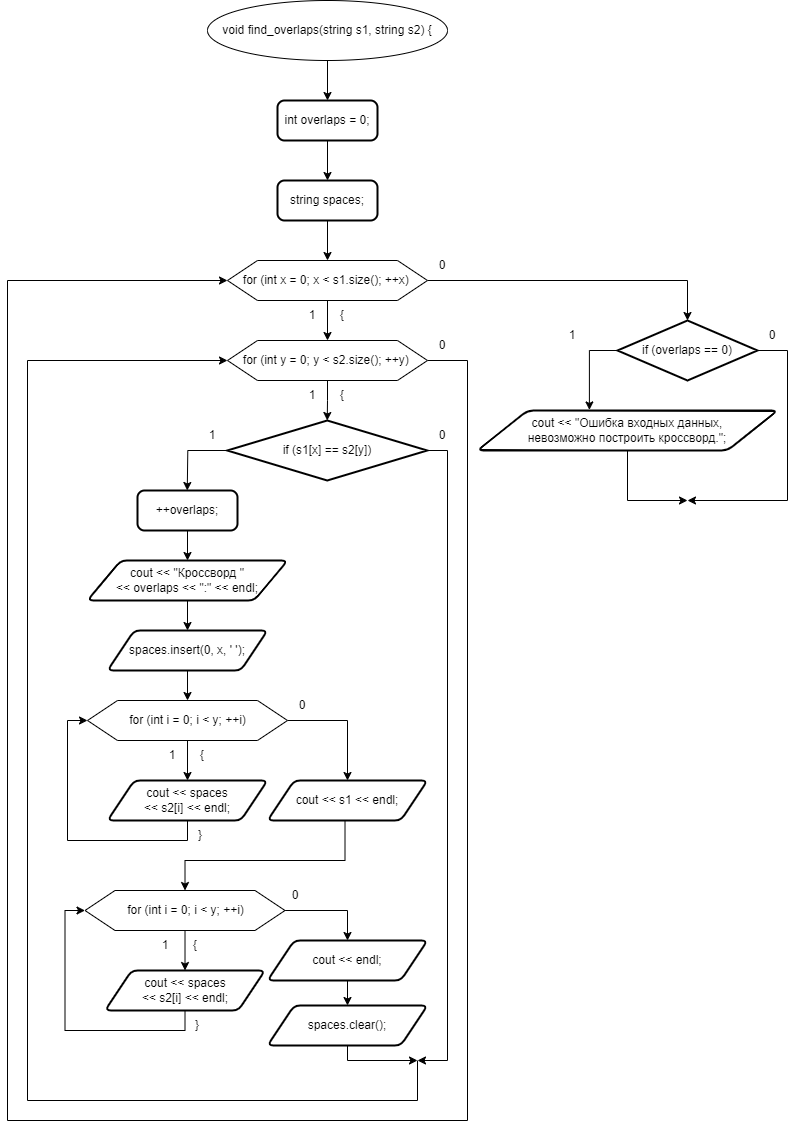
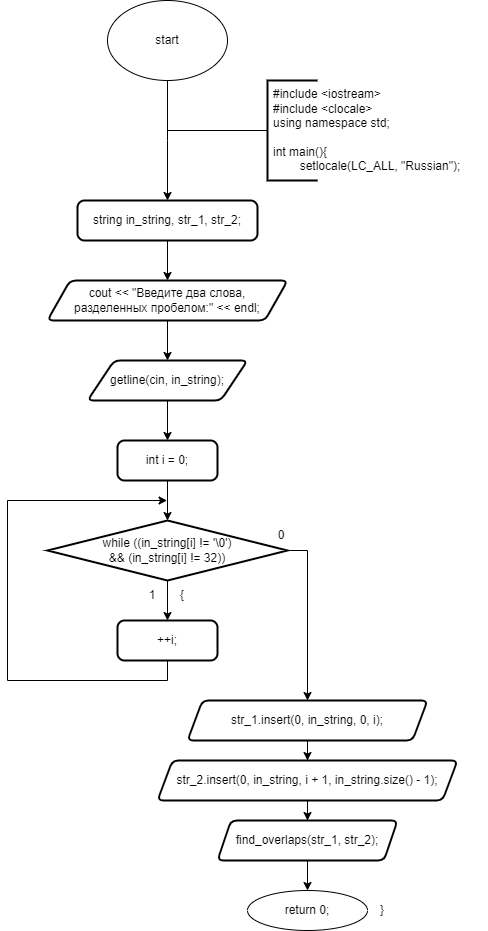
Постановка задачи

Из двух заданных с клавиатуры слов составить различные виды кроссвордов. При невозможности составить кроссворд (во введенных словах нет ни одной одинаковой буквы) напечатать сообщение об ошибке.

Анализ задачи

1. Для составления кроссворда необходимо наличие совпадений букв. Создается счетчик выводящий номер текущего кроссворда, если он равен 0, то вывести сообщение об ошибке.
2. Для поиска совпадений используется вложенный цикл. Внешний проходит по буквам первого слова, вложенный по буквам второго.
3. При нахождении совпадений выводятся слова составленные в кроссворд. Для вывода строк состоящих из одной буквы, перед буквами выводится количество пробелов, соответствующее индексу найденной буквы в первом слове.

Блок-схема



Код

#include <iostream>

#include <string>

using namespace std;

void find\_overlaps(string s1, string s2)

{

int overlaps = 0;

string spaces;

for (int x = 0; x < s1.size(); ++x)

{

for (int y = 0; y < s2.size(); ++y)

{

if (s1[x] == s2[y])

{

++overlaps;

cout << "Кроссворд " << overlaps << ":" << endl;

spaces.insert(0, x, ' ');

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

{

cout << spaces << s2[i] << endl;

}

cout << s1 << endl;

for (int i = y + 1; i < s2.size(); ++i)

{

cout << spaces << s2[i] << endl;

}

cout << endl;

spaces.clear();

}

}

}

if (overlaps == 0)

{

cout << "Ошибка входных данных, невозможно построить кроссворд.";

}

}

int main() {

string in\_string, str\_1, str\_2;

system("chcp 1251");

setlocale(LC\_ALL, "ru");

cout << "Введите два слова, разделенных пробелом:" << endl;

getline(cin, in\_string);

int i = 0;

while ((in\_string[i] != '\0') && (in\_string[i] != 32))

{

++i;

}

str\_1.insert(0, in\_string, 0, i);

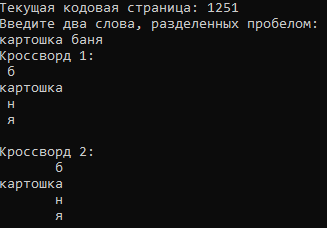
str\_2.insert(0, in\_string, i + 1, in\_string.size() - 1);

find\_overlaps(str\_1, str\_2);

return 0;

}

Решение



Выводы

В ходе работы, были исследованы основные способы работы со строками и методы строк, которые могут быть применены в будущих задачах.

Github

https://github.com/Hitikov/Lab\_6