Министерство образования и науки Российской Федерации  
Федеральное‌ ‌государственное‌ ‌бюджетное‌ ‌образовательное‌ ‌учреждение‌

высшего‌ ‌образования‌

**«Пермский национальный исследовательский  
политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**ОТЧЁТ**

**по лабораторной работе №14**

Дисциплина: «Основы теории алгоритмов и структуры данных»

Тема: Методы поиска: метод Кнутта-Мориса Пратта, метод Бойера-Мура

Вариант 17

Выполнила:

Студент группы Рис-20-1б

Томилов Владислав Алексеевич

Проверила:

Доцент кафедры ИТАС

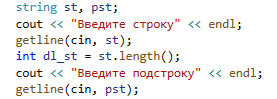
Полякова О. А.

**Пермь, 2021**

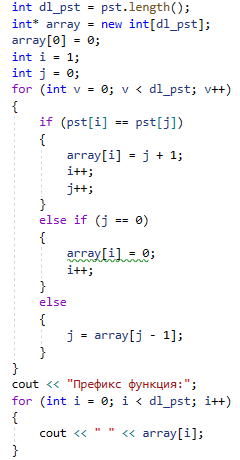
**Анализ задачи**

**1.** Для решения задачи необходимо разработать функцию int main():

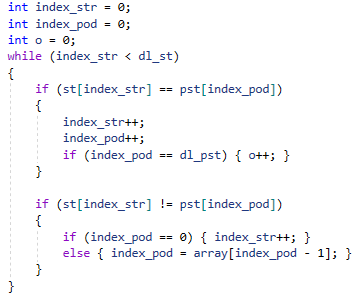
**1.1.** Пользователь вводит строку и подстроку в консоль, чтение строк происходит через оператор getline()



**1.2.** Создается массив префикс-функции и заполняется с помощью условия префиксов. Префикс-функция для і-го символа образа возвращает значение, равное максимальной длине совпадающих префикса и суффикса подстроки в образе, которая заканчивается i-м символом. Это значение будем хранить в array[i]. В конце массив префикс-функции выводится в консоль.



**1.3.** После получения массива префикс-функции производится поиск подстроки в строке и подсчитывается количество совпадений.



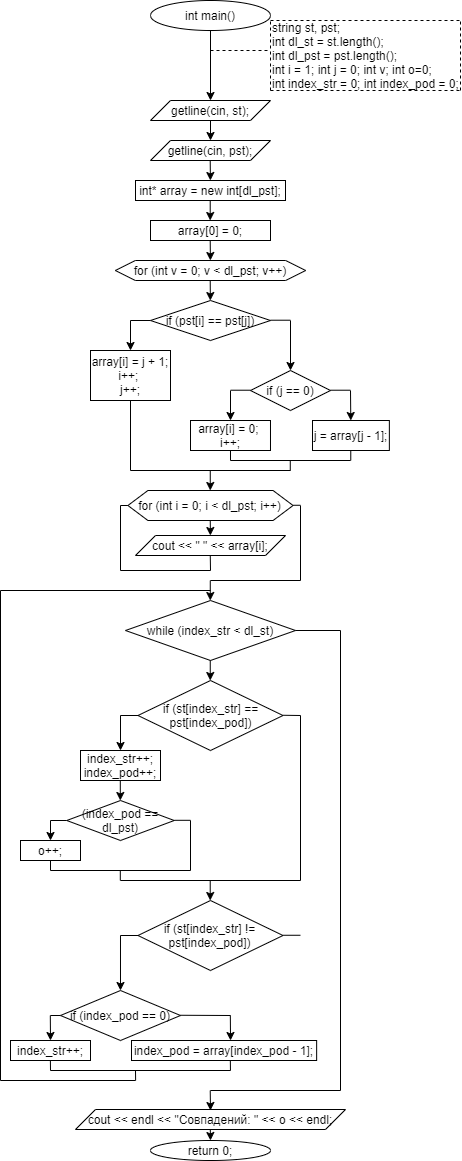
**1.4.** В итоге, выводится в консоль количество совпадений в строке.

**2.**В ходе работы были использованы следующие типы данных:

**2.1.** string – строка, подстрока

2.2. int – длина строки и длина подстроки

2.3 Динамический массив int – массив значений префикс-функции



**Код программы**

# include <string>

# include <iostream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Russian");

system("chcp 1251 > nul");

string st, pst;

cout << "Введите строку" << endl;

getline(cin, st);

int dl\_st = st.length();

cout << "Введите подстроку" << endl;

getline(cin, pst);

int dl\_pst = pst.length();

int\* array = new int[dl\_pst];

array[0] = 0;

int i = 1;

int j = 0;

for (int v = 0; v < dl\_pst; v++)

{

if (pst[i] == pst[j])

{

array[i] = j + 1;

i++;

j++;

}

else if (j == 0){ array[i] = 0; i++;}

else

{

j = array[j - 1];

}

}

cout << "Префикс функция:";

for (int i = 0; i < dl\_pst; i++)

{

cout << " " << array[i];

}

int index\_str = 0;

int index\_pod = 0;

int o = 0;

while (index\_str < dl\_st)

{

if (st[index\_str] == pst[index\_pod])

{

index\_str++;

index\_pod++;

if (index\_pod == dl\_pst) { o++; }

}

if (st[index\_str] != pst[index\_pod])

{

if (index\_pod == 0) { index\_str++; }

else { index\_pod = array[index\_pod - 1]; }

}

}

cout << endl << "Совпадений: " << o << endl;

system("pause");

return 0;

}

**Работа программы**

