

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»

КАФЕДРА ТЕОРЕТИЧЕСКОЙ И ПРИКЛАДНОЙ ИНФОРМАТИКИ

Лабораторная работа №4
по дисциплине «Основы программирования»

Факультет: ПМИ

Группа: ПМИ-03

Студенты: Сидоров Д.И., Малыгин С. А.

Преподаватель: Еланцева Е.Л.

- 1) Условие задачи: Данна непустая последовательность слов (не более 50), в каждом слове не более 20 символов. Слова разделены пробелом, за последним словом точка: в словах, оканчивающихся на ‘ing’, заменить это окончание на ‘ed’.

2) Анализ данных:

Входные данные: Стока – длиной не более 1000 символов.

Выходные данные: Изменённая строка.

Метод решения: Запустим цикл и будем вводить значения строки. После чего создадим ещё три строки : Fstr1 – строка содержащая “ ing ” , с помощью неё будем искать слова оканчивающиеся на ing в середине основной строки; Fstr2 - строка содержащая “ ing. ” , с помощью неё будем проверять последнее слово оканчивающееся точкой; И Cstr – строка содержащая “ed”, с помощью неё будем заменять найденные ing на ed. Запустим цикл, в котором будем идти по строке и искать совпадение i-го символа основной строки с первым символом строки Fstr1(то есть с “i”), и как только найдём, запустим вложенный цикл, в котором проверим оставшиеся три символа на совпадение со строкой Fstr1 или Fstr2. И в случае полного совпадения запустим функцию chang_str, в которой заменим ing на ed. Так как ing состоит из 3-х символов, а ed только из 2-х нужно будет также сдвинуть все последующие элементы строки на одну позицию влево. После чего выведем изменённую строку.

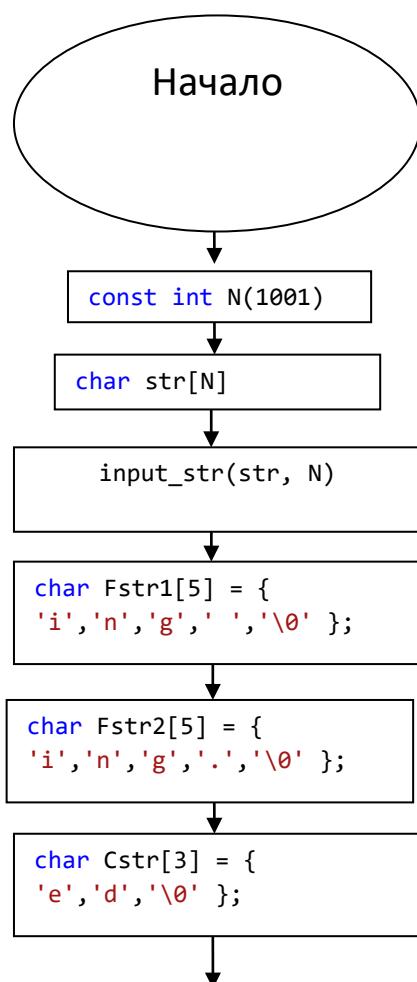
Можно выделить подпрограммы: input_str(str, MAX) – для ввода строки str с учётом её максимально возможной длины MAX.

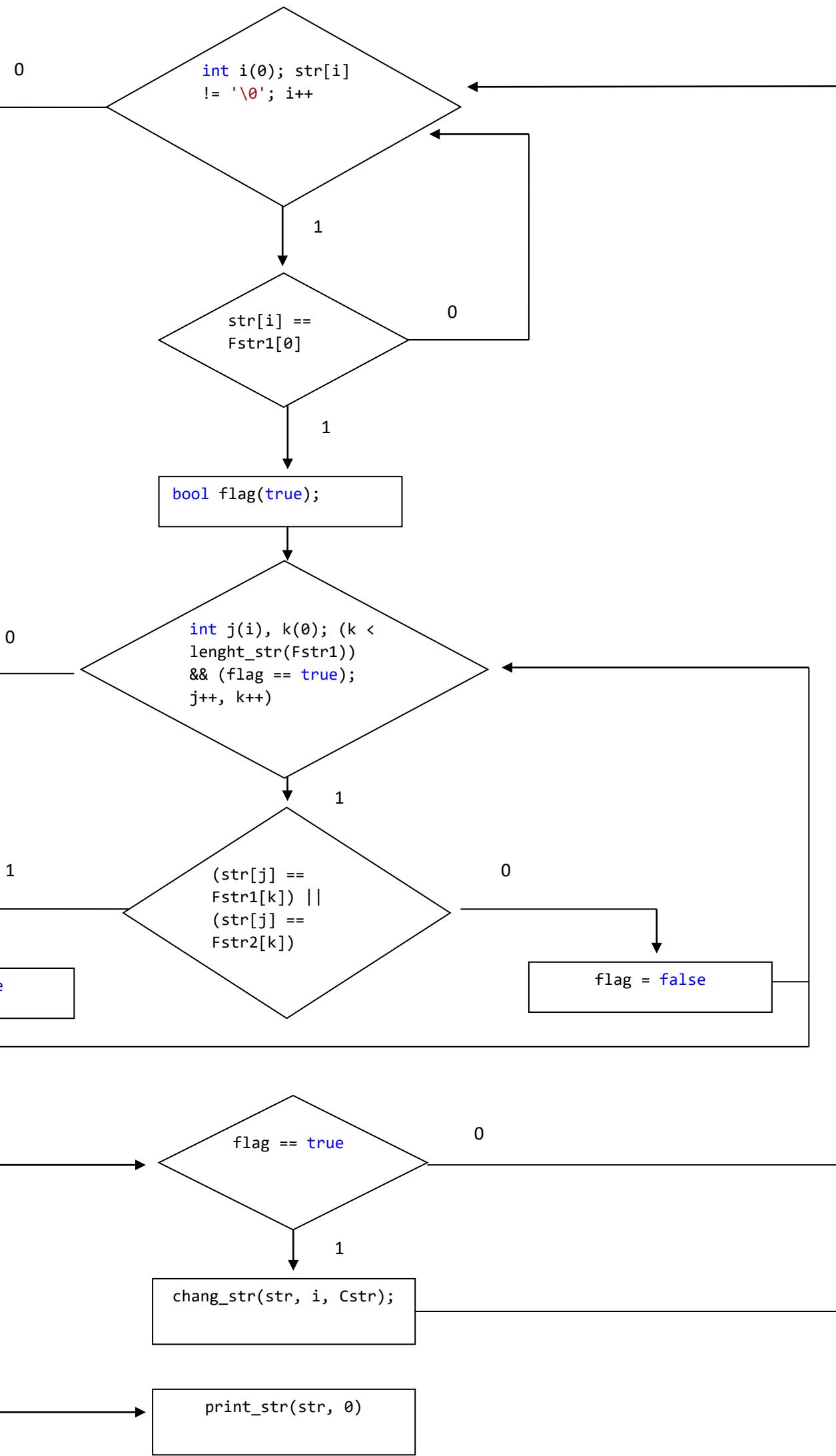
print_str(str, Num) – для вывода строки str с Num элемента.

length_str(str) – для расчёта длины строки str.

chang_str(str, Num, ch_str) – для замены с Num элемента на ch_str в строке str.

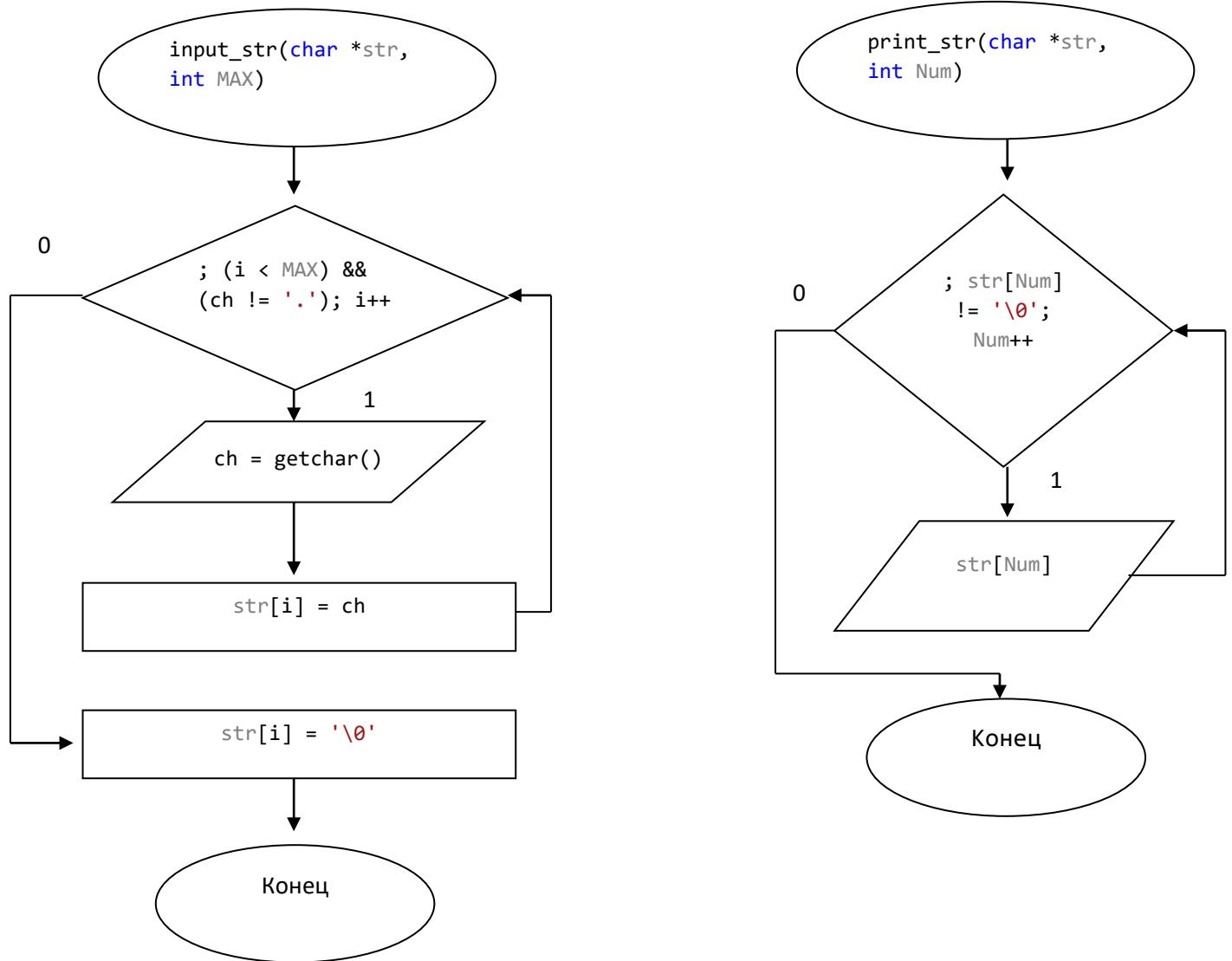
3) Алгоритм:

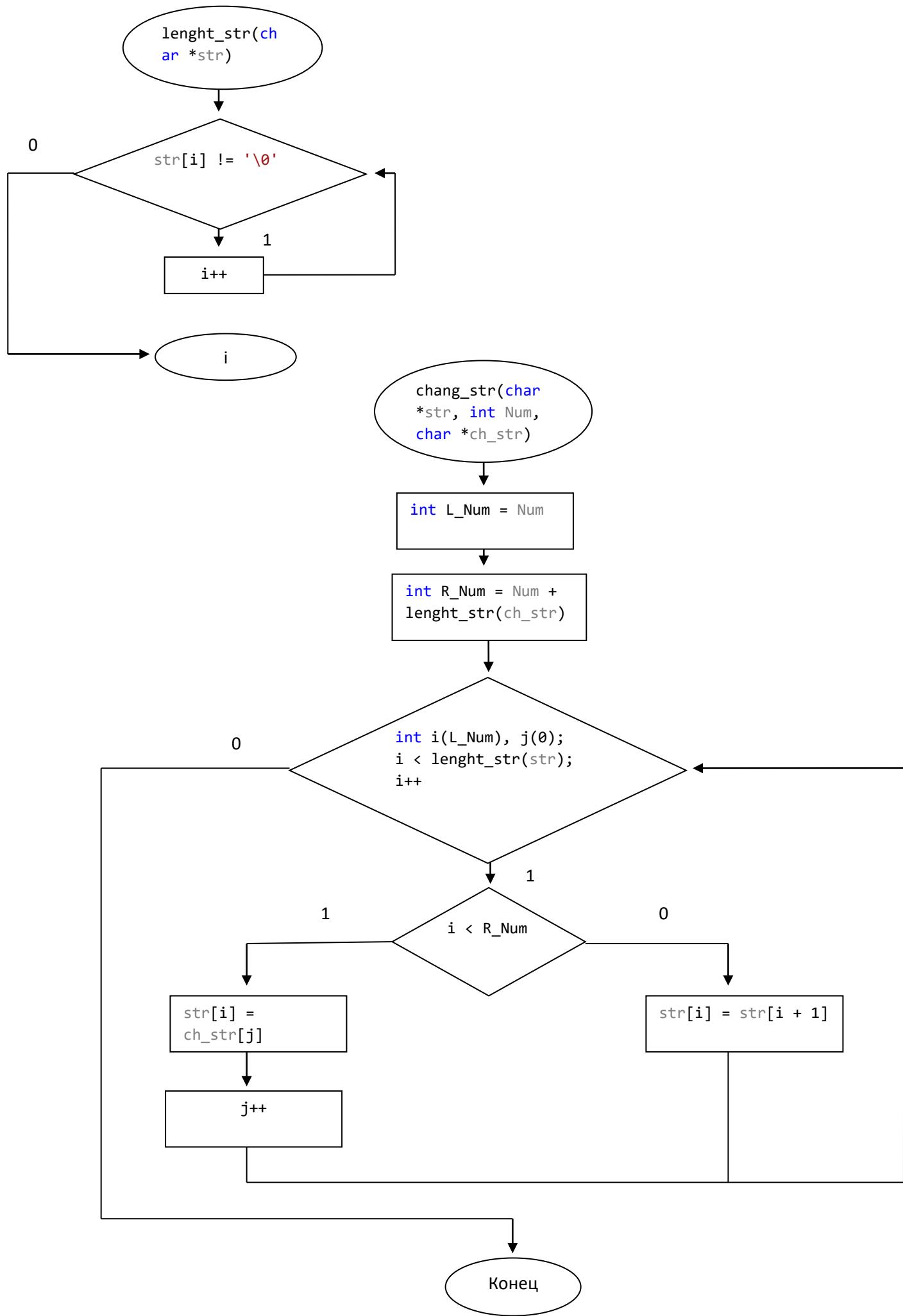




Конец

4) Структура программы





5) Текст программы:

```
#include<iostream>

using namespace std;

void input_str(char *str, int MAX)
{
    char ch(NULL);
    int i(0);
    for (; (i < MAX) && (ch != '.') ; i++)
    {
        ch = getchar();
        str[i] = ch;
    }
    str[i] = '\0';
}

void print_str(char *str, int Num)
{
    for (; str[Num] != '\0'; Num++) cout << str[Num];
}

int lenght_str(char *str)
{
    int i(0);
    while (str[i] != '\0') i++;
    return i;
}

void chang_str(char *str, int Num, char *ch_str)
{
    int L_Num = Num;
    int R_Num = Num + lenght_str(ch_str);

    for (int i(L_Num), j(0); i < lenght_str(str); i++)
    {
        if (i < R_Num)
        {
            str[i] = ch_str[j];
            j++;
        }
        else
        {
            str[i] = str[i + 1];
        }
    }
}

int main()
{
    const int N(1001);
    char str[N];

    input_str(str, N);

    char Fstr1[5] = { 'i', 'n', 'g', ' ', '\0' };
    char Fstr2[5] = { 'i', 'n', 'g', '.', '\0' };
    char Cstr[3] = { 'e', 'd', '\0' };

    for (int i(0); str[i] != '\0'; i++)
    {
        if (str[i] == Fstr1[0])
        {
            bool flag(true);

            for (int j(i), k(0); (k < lenght_str(Fstr1)) && (flag == true); j++, k++)
            {
                if ((str[j] == Fstr1[k]) || (str[j] == Fstr2[k]))
                {

```

```

        flag = true;
    }
    else
    {
        flag = false;
    }
}
if (flag == true)
{
    chang_str(str, i, Cstr);
}
}

print_str(str, 0);

return 0;
}

```

6) Тесты:

№	Дано	Результат	Примечание
	Строка str	Изменённая Странка str	
1	watching listening.	watched listened.	Слова оканчиваются на ing, перед пробелом и перед точкой.
2	ingredient.	ingredient.	Слово начинается с ing.
3	ing.	ed.	Слово состоит только из ing.

7) Результат работы программы:

Программа работает правильно, что подтверждают тесты.