**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

Лабораторная работа №1

по дисциплине «Программирование»

Указатели

Группа: АВТ-041

Студенты: Антонов А.

Преподаватель: Балакин В.

НОВОСИБИРСК 2021

**Задание.**

Функция находит в строке симметричный фрагмент вида "abcdcba" длиной 7 и более символов (не содержащий пробелов) и возвращает указатель на его начало и длину. С использованием функции «вычеркнуть» все симметричные фрагменты из строки.

**Проектирование программы**

Идея: Создание функции поиска и удаления симметричных строк.

Составные части программы:

1. Размерность массива:

#define STRUCT\_SIZE 5

#define STRING\_SIZE 255

1. Проверка на наличие отсутствия пробелов:

for (int i = 0; i < len; i++) {

if (str[i] == ' ') return 1;

}

1. Цикл поиска симметричных строк

for (int i = 0; i < len; i++) { //i and j the range in which the string is checked for symmetry

int j = len - 1; //ii and jj variables that perform validation within a range i and j

int ii = i;

int jj = j;

while (1) {

...

}

}

1. Функция удаления симметричных строк:

void razrez(char\* str) {

for (int i = 0; i < 5; i++) {

int d = start[i] - str; // Start of a symmetric string

int q = lenght[i]; // Size

while (q-- > 0) {

str[d++] = '\0'; // Fill in the lines with the null character

}

}

}

1. Вывод полученной строки на консоль:

for (int i = 0; i < len; i++) {

if (str[i] == '\0') continue;

printf("%c", str[i]);

}

Переменные:

start, lenght - массив

str\_simm\_num - индекс символа в строке

str - строка

len - длина строки

i, j, jj, ii - переменные-счетчики

d - начало симметричной строки

q - размер симметричной строки

**Текст программы с комментариями**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#define STRUCT\_SIZE 5

#define STRING\_SIZE 255

void razrez(char\*);

char\* start[STRUCT\_SIZE];

int lenght[STRUCT\_SIZE];

int str\_simm\_num = 0;

int main() {

char str[STRING\_SIZE];

scanf\_s("%s", str);

int len = strlen(str);

//////////////// checking for the presence absence of spaces /////////////////////

for (int i = 0; i < len; i++) {

if (str[i] == ' ') return 1;

}

///////////////////////////////////////////////////////////////////////////

/////////////////////////////// search for symmetric strings /////////////////////

for (int i = 0; i < len; i++) { //i and j the range in which the string is checked for symmetry

int j = len - 1; //ii and jj variables that perform validation within a range i and j

int ii = i;

int jj = j;

while (1) {

if (ii > jj) { // If it turns out that the lower search range is larger than the higher search range then the string is symmetric

start[str\_simm\_num] = &str[i]; // Writing a pointer to the beginning in the structure

lenght[str\_simm\_num] = j - i + 1; // Subtraction from the larger range of the lower one to find its length and write it to the structure

str\_simm\_num++; // We increase the index so that when we find the next symmetric rows they are not overwritten

i = j + 1; // If a repeat is found then we write it to the structure and shift the right border of the search by the entire size of the repeated element

}

if (j - i + 1 < 7) break; // If the search range is less than 7, then the problem condition that the symmetric string must be greater than 6 is violated, we interrupt the loop

if (str[ii] == str[jj]) { // If the two characters are the same, then you need to move the lower range by +1, and the upper range by -1 to continue checking

ii++; // Shifts the search scope -> <-

jj--;

continue;

}

else { // If the check failed, we restore the search ranges and shift the upper one to -1

ii = i; // Setting the left border to the beginning and shifting 1 position to the left <- <-

j--;

jj = j;

}

}

}

///////////////////////////////////////////////////////////////////////////

/\*

printf("%c",\*str\_simm[0].start);

printf(" %i",str\_simm[0].lenght);

\*/

razrez(str); // Function for deleting symmetric strings

// Displaying a string on the screen

for (int i = 0; i < len; i++) {

if (str[i] == '\0') continue;

printf("%c", str[i]);

}

return 0;

}

void razrez(char\* str) {

for (int i = 0; i < 5; i++) {

int d = start[i] - str; // Start of a symmetric string

int q = lenght[i]; // Size

while (q-- > 0) {

str[d++] = '\0'; // Fill in the lines with the null character

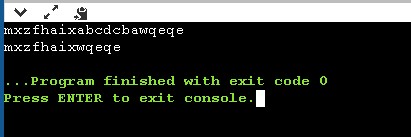
}

}

}

**Пример работы программы**

**Входные данные: mxzfhaixabcdcbawqeqe**



**Вывод:**  
Функция была реализована, программа работает корректно.