

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего
профессионального образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА №14

ОТЧЕТ ЗАЩИЩЕН

ОЦЕНКОЙ ПРЕПОДАВАТЕЛЬ

Доцент, кандидат техн. наук

Е.П. Овсянников

должность, уч. степень, звание

подпись, дата

инициалы, фамилия

Отчет по лабораторной работе № 10

Динамические массивы

по дисциплине: “Основы программирования”

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. 1942

К.Д.Марков

подпись, дата

инициалы, фамилия

Санкт-Петербург

2020г.

1) Цель:

Изучение методов организации массивов переменной размерности и работы с ними.

2) Постановка задачи:

По содержимому текстового файла составить словарь (в алфавитном порядке).

3) Формализация:

- Существует данный файл, из которого происходит преобразование
- Вывод производится на экран в командной строке
- Текстовый файл должен содержать только строчные буквы английского алфавита.

4) Листинг программы:

```
#define CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <malloc.h>
#include <string.h>
char **dictionary, **swop;
char* word, * ptr;
char line[1024];
FILE* file;
int i = 0, j = 0, k = 0;
int nw = 1, lw = 0, plw = 0, flag = 0, s = 1, si = 0;
void write() {
    dictionary = (char**)realloc(dictionary, nw * sizeof(char*));
    dictionary[nw - 1] = (char*)calloc(plw+1, sizeof(char));
    for (i = nw-1; i < nw; i++) {
        for (j = 0; j < plw; j++) {
            dictionary[i][j] = word[j];
        }
        dictionary[i][j] = '\0';
    }
    nw++;
}
void sort() {
    i = 0;
    while (si != nw - 1) {
        while (i + s < nw - 1) {
            if (strcmp(dictionary[i], dictionary[i + s]) == 0) {
                free(dictionary[i+s]);
                lw++;
            }
            s++;
        }
        i++;
        s = 1;
        si++;
    }
}
```

```

    }
    si = 0;
    i = 0;
    s = 1;
    swop = (char**)calloc(nw-1-lw, sizeof(char*));
    while (si != nw-1-lw) {
        while (i + s < nw-1) {
            if (strcmp(dictionary[i], dictionary[i + s]) < 0) {
                flag = i;
                s++;
            }
            else {
                flag = i + s;
                i = flag;
                s = 1;
            }
        }
        swop[si] = (char*)calloc(dictionary[flag], sizeof(char));
        strcpy(swop[si], dictionary[flag]);
        free(dictionary[flag]);
        si++;
        i = 0;
        s = 1;
    }
    free(dictionary);
}

void read() {
    while (!feof(file)) {
        ptr = fgets(line, 1024, file);
        while (*ptr != '\0') {
            if (*ptr != ' ' && *ptr != ',' && *ptr != '.' && *ptr != '?' && *ptr != '!' && *ptr
!= '(' && *ptr != ')' && *ptr != ':' && *ptr != ';' && *ptr != '"' && *ptr != '\n') {
                plw++;
                word = (char*)realloc(word, plw * sizeof(char));
                word[k++] = *(ptr++);
            }
            else {
                if (*(ptr - 1) == ' ' || *(ptr - 1) == ',' || *(ptr - 1) == '.' || *(ptr - 1)
== '?' || *(ptr - 1) == '!' || *(ptr - 1) == '(' || *(ptr - 1) == ')' || *(ptr - 1) == ':' || *(ptr - 1) == ';' || *(ptr
- 1) == '"' || *(ptr - 1) == '\n') {
                    ptr++;
                }
                else {
                    ptr++;
                    write();
                    plw = 0;
                    k = 0;
                    free(word);
                    word = (char*)calloc(plw, sizeof(char));
                }
            }
        }
    }
}

int main(void) {
    fopen_s(&file, "help.txt", "r+t");
    word = (char*)calloc(plw, sizeof(char));
    dictionary = (char**)calloc(nw, sizeof(char*));
    read();
    fclose(file);
}

```

```

    sort();
    for (i = 0; i < nw-1-lw; i++) {
        puts(swap[i]);
    }
    for (i = 0; i < nw-1-lw; i++) {
        free(swap[i]);
    }
    free(swap);
    return 0;
}

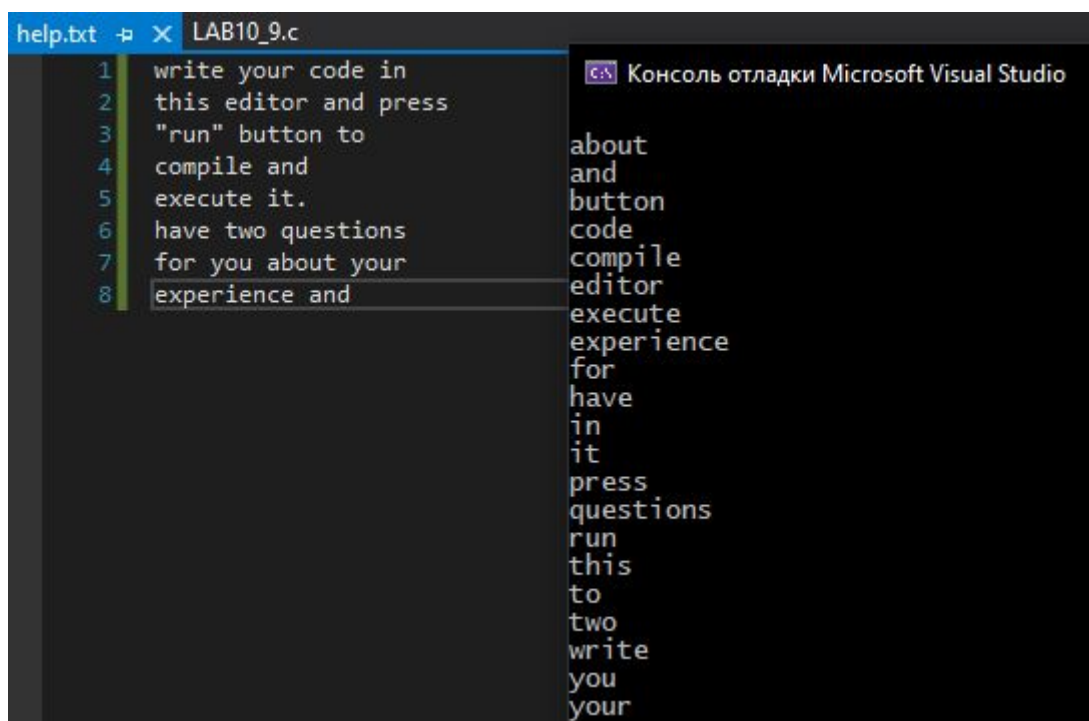
```

5) Тестовые примеры:

Таблица 1 - тестовые примеры

№	исходные данные	ВЫВОД
1	<p>write your code in this editor and press "run" button to compile and execute it.</p> <p>have two questions for you about your experience and</p>	<p>about and button code compile editor execute experience for have in it press questions run this to two write you your</p>

6) Результаты выполнения:



The screenshot shows the Microsoft Visual Studio IDE. On the left, the 'LAB10_9.c' file is open, containing the following code:

```
1 write your code in  
2 this editor and press  
3 "run" button to  
4 compile and  
5 execute it.  
6 have two questions  
7 for you about your  
8 experience and
```

On the right, the 'Консоль отладки Microsoft Visual Studio' (Microsoft Visual Studio Debug Console) displays the output of the program, which is a list of words extracted from the code and sorted alphabetically:

```
about  
and  
button  
code  
compile  
editor  
execute  
experience  
for  
have  
in  
it  
press  
questions  
run  
this  
to  
two  
write  
you  
your
```

Рисунок 1 - Результат выполнения

7) Выводы:

В результате выполнения лабораторной работы была создана программа, составляющая словарь из исходного текстового файла в алфавитном порядке.

Результаты работы программы в п.6 совпадают с данными рассчитанными вручную в п.5, следовательно, программа работает верно.

Выполнив лабораторную работу, я научился работать с динамическими массивами.