

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МОЭВМ

ОТЧЕТ
по лабораторной работе №3
по дисциплине «Программирование»
Тема: «Использование указателей»

Студент гр. 7381

Тарасенко Е.А.

Преподаватель

Берленко Т.А.

Санкт-Петербург

2017

Цель работы

Написать программу, которая форматирует некоторый текст и выводит результат на консоль. На вход программе подается текст, который заканчивается предложением "Dragon flew away!".

Дополнительная информация для выполнения лабораторной работы

Предложение (кроме последнего) может заканчиваться на:

- . (точка)
- ; (точка с запятой)
- ? (вопросительный знак)

Программа должна изменить и вывести текст следующим образом:

- Каждое предложение должно начинаться с новой строки.
- Табуляция в начале предложения должна быть удалена.
- Все предложения, в которых больше одной заглавной буквы, должны быть удалены.
- Текст должен заканчиваться фразой "Количество предложений до n и количество предложений после m", где n - количество предложений в изначальном тексте (**без учета** терминального предложения "Dragon flew away!") и m - количество предложений в отформатированном тексте (**без учета** предложения про количество из данного пункта).

* Порядок предложений не должен меняться;

* Статически выделять память под текст нельзя;

* Пробел между предложениями является разделителем, а не частью какого-то предложения.

Вывод

В ходе данной работы были отработаны некоторые навыки работы со строками. В частности, в этом проекте была изучена стандартная библиотека языка СИ <ctype.h>, содержащая функции, позволяющие определить, принадлежит ли введенная буква к символам верхнего или нижнего регистров.

В данной работе память под вводимый текст выделялась динамически функцией «malloc», содержащейся в библиотеке <stdlib.h>. После окончания всех операций память зачищается (ф-ция «free»). Стоит отметить, что количество вызовов функции «malloc» совпало с количеством вызовов функции «free».

Приложение

Код программы:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <ctype.h>
#define N 1000
char** text;
int* index_del;
int count_str = 0;
int deleted_str = 0;
int i, j, n = 0, flag = 0;
char a;
int main()
{
    text = (char**)malloc(N*sizeof(char*));
    for(i=0; i<N; i++) text[i] = (char*)malloc(N*sizeof(char));
    index_del = (int*)malloc(N*sizeof(int));
    i = 0;
    for(int k; k<(N*N); k++)
    {
        a = getchar();
        if ((a == ' ' && i == 0) || a == '\t' || a == '\n')
            continue;
        text[count_str][i] = a;
        if(text[count_str][i] == '.'){
            text[count_str][i+1] = '\n';
            count_str++;
            i = 0;
            continue;
        }
    }
}
```

```

        if(text[count_str][i] == ';'){
            text[count_str][i+1] = '\n';
            count_str++;
            i = 0;
            continue;
        }
        if(text[count_str][i] == '?'){
            text[count_str][i+1] = '\n';
            count_str++;
            i = 0;
            continue;
        }
        if(text[count_str][i] == '!'){
            text[count_str][i+1] = '\n';
            count_str++;
            i = 0;
            break;
        }
        i++;
    }
    for(i=0; i<count_str; i++)
    {
        n = 0;
        for(j=0; j<strlen(text[i]); j++){
            if(isupper(text[i][j])) n++;
        }
        if(n > 1){
            index_del[deleted_str] = i;
            deleted_str++;
        }
    }
    for(i=0; i<count_str; i++)
    {
        flag = 0;
        for(n=0; n<deleted_str; n++) if(i == index_del[n])
flag = 1;
        if(flag == 0) for(j=0; j<strlen(text[i]); j++)
printf("%c", text[i][j]);
    }
    printf("Количество предложений до %d и количество предложений
после %d\n", count_str-1, count_str-deleted_str-1);
    for(i=0; i<N; i++) free(text[i]);
    free(text);
    free(index_del);
    return 0;
}

```