

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

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

Студент гр. 3344		Тукалкин.В.А
Преподаватель		Глазунов.С.А

Санкт-Петербург
2023

Цель работы

Освоение работы с указателями и динамической памятью.

Задание.

Вариант 2.

Напишите программу, которая форматирует некоторый текст и выводит результат на консоль. На вход программе подается текст, который заканчивается предложением "Dragon flew away!". 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, которые заканчиваются на '?' 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).

Выполнение работы

Выполнение работы будет расписано по шагам:

- 1) Включить библиотеки `stdio.h`, `stdlib.h`, `string.h`
- 2) Написать функцию `main`, выделить память для переменной `string`, определить два счётчика.
- 3) Написать цикл `while`, который будет считывать и форматировать строку.
- 4) Выделить память для `extra_string`, определить счётчик и переменную типа `char` для помощи в считывании строки.
- 5) Написать функцию `end`, которая будет определять является символ концом строки или нет.
- 6) Считывать строку в переменную `string` и перевыделять память для этой переменной в цикле.
- 7) После завершения цикла добавить в конец строки символ `\0` и увеличить счётчик `start_len`.
- 8) Проверить строку на наличие '?', если его нет, то проверить нулевой символ строки, если первый символ знак пробела или `\t` и если первый символ `\n`, то сместить строку на один символ, иначе нулевой символ определить `\n`. Далее `extra_string` записывается в `string` и перевыделять память для `string`.
- 9) Написать условие, если в строке есть "Dragon flew away!", то закончить всё.
- 10) Циклом `for` освободить всю память от `string`.
- 11) Вывести строку "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), и указать длины на 1 меньше, чтобы не учитывать строку "Dragon flew away!". Очистить память.

Тестирование.

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	<p>555 posuere velit efficitur, egestas nunc quis, dictum purus? Phasellus nunc augue, dApibus quis molestie iD, gravida a velit? Donec congue mauris sed lacus pulvinar, quis semper orci sol4licitudin? 40 Nu555lla rutrum feugiat felis a pharetra. Integer at quam et erat iaculis iaculis hendrerit a te4llus? Morbi co7ndimentum 555 ex justo, nec pharetra mauris vestibulum a. Suspendisse quis mi neque7. Sed finibus magna et mauris elementum tempus? Sed</p>	<p>40 Nu555lla rutrum feugiat felis a pharetra. Morbi co7ndimentum 555 ex justo, nec pharetra mauris vestibulum a. Suspendisse quis mi neque7. Class aptenT taciti sociosqu ad litora torquent per cOnubia nostra, per inceptos himenaeos. Donec at nunc ac mauris suscipit venenatis. Nam 7elementum id enim eu congue; Donec accumsan convallis ipsum vitae lacinia. Dragon flew away! Количество</p>	Верный ответ

<p>finibus magna et mauris elementum tempus? Class aptenT taciti sociosqu ad litora torquent per cOnubia nostra, per inceptos himenaeos. Donec at nunc ac mauris suscipit venenatis. Nam 7elementum id enim eu congue; Donec accumsan convallis ipsum vitae lacinia. Dragon flew away!</p>	<p>предложений до 13 и количество предложений после 7</p>	
--	---	--

Выводы

Была изучена работа с указателями и динамической памятью.

Разработана программа, выполняющая операции с поступающим текстом. На вход подаётся текст, который программа считывает и форматирует по следующим правилам: каждое предложение должно начинаться с новой строки, табуляция в начале предложения должна быть удалена, все предложения, которые заканчиваются на '?' должны быть удалены. После нового текста добавляется строка "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).

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

```
Название файла: main.c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int end(int str){
    if(strchr(".;?!",str)) return 1;
    return 0;
}

int main(){
    char **string=malloc(sizeof(char*));
    int start_len=0, end_len=0;
    while(1){
        char *extra_string=malloc(sizeof(char));
        char symbol=1;
        int len=0;
        while(!end(symbol)){
            symbol=getchar();
            extra_string[len++]=symbol;
            extra_string=realloc(extra_string,
sizeof(char)*(len+1));
        }
        extra_string[len]='\0';
        start_len++;
        if(strchr(extra_string,'?')==NULL){
            if(strchr(" \t", extra_string[0])){
                if(extra_string[1]=='\n'){
                    memmove(extra_string, extra_string+1,
strlen(extra_string));
                }else{
                    extra_string[0]='\n';
                }
            }
            string[end_len++]=extra_string;
            string=realloc(string,
sizeof(char*)*(end_len+1));
        }
        if (strstr(extra_string, "Dragon flew away!")) break;
    }
    for(int i=0;i<end_len;i++){
        printf("%s",string[i]);
        free(string[i]);
    }
    printf("\nКоличество предложений до %d и количество
предложений после %d", start_len - 1, end_len - 1);
    free(string);
}
```