

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

ОТЧЕТ
по лабораторной работе №1
по дисциплине «Программирование»
Тема: Обзор стандартной библиотеки языка Си

Студент гр. 1304

Арчибасов Е.О.

Преподаватель

Чайка К.В.

Санкт-Петербург

2022

Цель работы.

Знакомство со стандартной библиотекой языка Си. Приобретение навыков применения функций, и прочего содержимого стандартной библиотеки языка.

Задание.

Вариант 1

Напишите программу, на вход которой подается текст на английском языке (длина текста не превышает 1000 символов) и слово `str` (длина слова не превышает 30 знаков). Слова в тексте разделены пробелами или точкой. Программа должна вывести строку "exists", если `str` в тексте есть и "doesn't exist" в противном случае.

Программа должна реализовать следующий алгоритм:

- разбить текст на слова, используя функции стандартной библиотеки
- отсортировать слова, используя алгоритм быстрой сортировки (см. функции стандартной библиотеки)
- определить, присутствует ли в тексте `str`, используя алгоритм двоичного поиска (для реализации алгоритма двоичного поиска используйте функцию стандартной библиотеки)
- вывести строку "exists", если `str` в тексте есть и "doesn't exist" в противном случае.

Основные теоретические положения.

Стандартной библиотекой языка Си называется часть стандарта ANSI C, посвященная заголовочным файлам и библиотечным подпрограммам. Является описанием реализации общих операций, таких как обработка ввода-вывода и строк, в языке программирования Си. Стандартная библиотека языка Си — это описание программного интерфейса, а не настоящая библиотека, пригодная для использования в процессе компиляции.

В работе использовались следующие заголовочные файлы:

1. *string.h* — заголовочный файл стандартной библиотеки языка Си, содержащий функции для работы с нуль-терминированными строками и различными функциями работы с памятью. Кроме этого, строковые функции работают только с набором символов ASCII или его совместимыми расширениями.
2. *stdlib.h* — заголовочный файл стандартной библиотеки языка Си, который содержит в себе функции, занимающиеся выделением памяти, контролем процесса выполнения программы, преобразованием типов и другие.
3. *stdio.h* — заголовочный файл стандартной библиотеки языка Си, содержащий определения макросов, константы и объявления функций и типов, используемых для различных операций стандартного ввода и вывода.

Выполнение работы.

В начале работы выделяется память для хранения введенного пользователем текста (*text*) и буфер для слова (*word*). Обе символьной строки считываются функцией стандартной библиотеки *fgets()* из стандартного потока ввода. Далее из строк удаляются полученные при вводе символы новой строки.

После этого выделяется память под отдельное хранение каждого из слов, содержащегося в полученном предложении. Слова выявляются при помощи функции *strtok()*, после чего последовательно направляются в выделенную память, где подвергаются сортировке с использованием *qsort()*. Для сравнения используется функция стандартной библиотеки языка *strcmp()*.

В конце работы программы, перед получением ответа о наличии слова в строке, отсортированные слова подвергаются проверке при помощи алгоритма двоичного поиска, реализованного в функции *bsearch()*. Очищается используемая память.

Тестирование

Результаты тестирования представлены в таблице 1.

Таблица 1-Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	sadasjhdbajsd Acxzcas src.ububu vid vid	exist	Программа работает корректно
2.	sadasjhdbajsd Acxzcas src.ububu vidik vid	doesn`t	Программа работает корректно

Вывод

В ходе выполнения программы, были изучены основные положения стандартной библиотеки языка Си. Была успешно разработана программа, которая использует функции стандартной библиотеки языка.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.c

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

int convert(const void *a, const void *b){
    char** x = (char**)a;
    char** y = (char**)b;
    return strcmp(*x, *y);
}

int main(){
    char *text = (char*)malloc(1001);
    char *word = (char*)malloc(31);

    fgets( text,1001,stdin);
    text[strcspn(text, "\n")] = 0;
    fgets(word,31,stdin);
    word[strcspn(word, "\n")] = 0;

    char **words = (char**)malloc(300*sizeof(char*));
    int i=0;
    char* cwrд = strtok(text, ". ");

    while (cwrд != NULL) {
        words[i++] = cwrд;
        cwrд = strtok(NULL, ". ");
    }
    qsort(words, i, sizeof(char**), convert);

    char ** res = bsearch(&word, words ,i, sizeof(char**), convert);

    if(res)
        printf("exists\n");
    else
        printf("doesn't exist\n");

    free(text);
    free(word);
    free(words);
    return 0;
}
```