

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.

Студентка гр. 0382

Кривенцова Л.С.

Преподаватель

Берленко Т.А.

Санкт-Петербург

2021

Цель работы.

Изучить и освоить функционал стандартной библиотеки языка программирования Си.

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.

stdlib.h :

- Функции для работы с динамической памятью;
- Функции для преобразования строки в число;
- Генерации псевдослучайных чисел;
- Функции для управления процессом выполнения программы;
- Функции для вычисления абсолютного значения и деления целых чисел;
- Функции для сортировки и поиска.

qsort и bsearch :

```
void qsort (void* base, size_t num, size_t size,  
            int (*compar)(const void*,const void*));
```

Функция принимает указатель на начальный элемент массива, количество элементов и размер одного элемента, а также указатель на функцию для сравнения двух элементов.

Так как тип элементов может быть любым, то и указатель на первый элемент массива имеет тип `void`. Это позволяет, зная адрес первого элемента и размер каждого элемента вычислить адрес любого элемента массива в памяти и обратиться к нему. Остается только сравнить 2 элемента имея 2 указателя на них. Это выполняет функция `strcmp`, указатель на которую передается функции `qsort` в качестве одного из параметров.

Функция `strcmp` принимает 2 указателя типа `void`, но в своей реализации может привести их к конкретному типу (так как её реализация остается за программистом, он точно знает элементы какого типа он сортирует) и сравнивает их. Результат сравнения определяется знаков возвращаемого функций `qsort` числа.

Выполнение работы.

Ход решения:

Используется стандартная библиотека языка `C` и её заголовочный файлы `stdlib.h` и `string.h`. После выделения динамической памяти для хранения строк (с помощью `malloc`), программа принимает через стандартный поток ввода `stdin` на вход английский текст, затем слово (поиск которого будет производиться далее). Текст разбивается на слова (создается динамический массив слов). Выполняется быстрая сортировка слов с помощью функции стандартной библиотеки `qsort`. Используя функцию стандартной библиотеки `bsearch`, алгоритмом двоичного поиска, определяется, присутствует ли в тексте введенное ранее слово. Через стандартный поток вывода `stdout` на экран печатается строка-результат ("exists", если слово в тексте есть и "doesn't exist" в противном случае). Освобождается динамически выделенная память.

Переменные:

`int i;` -целочисленная переменная, хранящая количество слов в тексте.

char sym; - переменная, в которую считывается символ, прежде чем записать его в символьный массив (текста или слова).

*char*phrase;* - символьный динамический массив, хранящий текст.

*char *str;* - символьный динамический массив, хранящий слово.

*char**words;* - двумерный динамический массив слов текста.

*char *ps;* - вспомогательная переменная-указатель для корректной работы функции *strtok*.

*char *pointer;* - переменная, в которую записывается результат функции *bsearch*.

Функции:

1. int main().

Функция выполняет выделение памяти под динамические массивы (*malloc*), считывает текст и слово, разбивает текст на слова (*wcstok*), выполняет быструю сортировку массива со словами по алфавиту (*qsort*) и поиск слова в массиве (*bsearch*). После проверки условия оператором (*if; else*) печатает результат поиска на экран. Освобождает выделенную память (*free*). Не принимает аргументов, возвращает значение 0 при корректной работе.

2. int cmp(const void *p1, const void *p2).

Функция, указатель на которую передается функциям *qsort* и *bsearch* в качестве одного из параметров. Функция принимает 2 указателя типа *void*, но в своей реализации приводит их к типу (*char***) и сравнивает их. Результат сравнения определяется знаком возвращаемого функций (*qsort* или *bsearch*) числа.

Разработанный программный код см. в приложении А.

Тестирование.

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	Java is a general-purpose computer programming language that is concurrent class-based object-oriented and specifically	exists	Программа выводит верный ответ.

	<p>designed to have as few implementation dependencies as possible. It is intended to let application developers "write once, run anywhere" (WORA) meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of computer architecture. As of 2016 Java is one of the most popular programming languages in use particularly for client-server web applications, with a reported 9 million developers. Java was originally developed by James Gosling at Sun Microsystems (which has since been acquired by Oracle Corporation) and released in 1995 as a core component of Sun Microsystems Java platform.</p> <p>is</p>		
2.	<p>The function takes a pointer to the starting element of the array, the number of elements and the size of one element, and a pointer to the function for comparing two elements.</p>	doesn't exist	Программа выводит верный ответ.
3.	<p>The compar function accepts 2 pointers of the void type, but in its implementation it can cast them to a specific type (since its implementation is left to the programmer, he knows exactly what type of elements he sorts) and compares them. The result of the comparison is determined by</p>	exists	Программа выводит верный ответ.

	the characters of the number returned by the qsort functions.		
--	--	--	--

Выводы.

Были изучен и освоен функционал стандартной библиотеки языка программирования Си.

Разработана программа, разбивающая текст на слова, сортирующая их, выполняющая поиск слова в тексте, используя функции стандартной библиотеки (strtok, qsort и bsearch).

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: lb1.c

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

int cmp(const void *p1, const void *p2)
{
    return strcmp(*(char**)p1, *(char**)p2);
}

int main(){
    int i = 0;
    char sym;
    char *phrase = malloc(1001*sizeof(char));
    char *str = malloc(31*sizeof(char));
    char **words = malloc(1000*sizeof(char*));
    char *ps;
    fgets( phrase,1001,stdin);
    fgets( str,31,stdin);
    phrase[strlen(phrase)]='\0';
    str[strlen(str)]='\0';
    ps = strtok(phrase, " .");
    while (ps != NULL) {
        words[i++] = ps;
        ps = strtok(NULL, " .");
    }
    qsort( words, i, sizeof(char**), cmp);
    char ** pointer;
    pointer = bsearch(&str, words,i, sizeof(char**), cmp);
    if(pointer)
    {
        printf("exists");
    }
    else
    {
        printf("doesn't exist");
    }
    free(phrase);
    free(words);
    free(str);
    return 0;
}
```