# МИНОБРНАУКИ РОССИИ

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**ОТЧЕТ**

**по лабораторной работе №1**

**по дисциплине «Программирование» Тема: «Обзор стандартной библиотеки»**

|  |  |
| --- | --- |
| Студент гр. 7381 | Лауцюс М. |
| Преподаватель | Берленко Т.А. |

Санкт-Петербург 2018

# Цель работы.

Ознакомиться со стандартной библиотекой си, изучить ее функционал и возможности.

# Задание.

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.

# time.h

В заголовочном файле time.h можно найти объявления типов и функций для работы с датой и временем. В том числе:

* Функция, позволяющая получить текущее календарное время
* Функция, позволяющая получить время в тактах процессора с начала выполнения программы
* Функция для вычисления разности в секундах между двумя временными штампами
* Функции для вывода значения даты и времени на экран

А также структура tm, содержащая компоненты календарного времени и функция для преобразования значения времени в секундах в объект такого типа.

# ctype.h

В состав стандартной библиотеки входят функции для работы с символами, объявленые в заголовочном файле ctype.h

Эти функции позволяют переводить символы из верхнего регистра в нижний, определять является ли этот символ буквой, цифрой или каким-то служебным.

# assert.h

В стандартной библиотеке объявлен макрос препроцессора assert(). С его помощью можно выполнять проверку некоторых условий в процессе выполнения программы.

Если условие ложно, то в процессе выполнения будет выведена некоторая информация о том в каком месте это произошло (имя файла с исходным кодом, имя функции, номер строки) и само условие. Стоит заметить, что для отключения проверок достаточно всего лишь добавить макрос #define NDEBUG перед включением заголовочного файла assert.h

Использование assert может сильно упростить отладку программ, гарантируя, что все необходимые условия в процессе выполнения программы соблюдены.

# string.h﻿

﻿﻿ Стандартная библиотека Си содержит различные функции для работы с памятью и строками, объявленые в заголовочном файле string.h.  
 Среди них есть функции для:   
 - копирования памяти  
 - сравнения памяти  
 - сравнения строк  
 - разбиения строки на токены  
 - конкатенации строк  
 - поиска символов и построки в строке  
 - определения длины строки  
 и некоторые другие

# stdlib.h

В этом заголовочном файле собраны объявления различных функций, частью из которых мы уже пользовались ранее.

* 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

# Указатели на функции

Указатель на функцию позволяет нам работать с функцией как с обычной переменной, в том числе передавать функцию в качестве аргумента другой функции.

# Ход работы

Была написана программа, 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 выводит строку "exists", если **str** в тексте есть и "doesn't exist" в противном случае.

Программа реализует следующий алгоритм:

* разбивает текст на слова, используя**функции стандартной библиотеки**
* отсортировывает слова, используя алгоритм быстрой сортировки (см. **функции стандартной библиотеки**)
* определяет, присутствует ли в тексте **str**, используя алгоритм двоичного поиска.
* выводит строку "exists", если **str** в тексте есть и "doesn't exist" в противном случае.

# Вывод:

Было проведено ознакомление с основными функциями и типами данных стандартной библиотеки языка Си. Они были использованы при создании программы на языке Си, в которой используются некоторые возможности стандартной библиотеки.

# Исходный код проекта:

Файл Makefile

all:

gcc main.c

Файл main.c

# #include <stdio.h>

# #include <stdlib.h>

# #include <string.h>

# int compar1(const void \* a, const void \* b){return strcmp(a,b);}

# int main() {

# char text[1000];

# fgets(text, 1000, stdin);

# char word[30];//str

# fgets(word, 30, stdin);

# char words[500][30];

# char\* pch;

# pch = strtok (text,". ");

# int i=0;

# while (pch != NULL)

# {

# strcpy(words[i],pch);

# pch = strtok (NULL, " .");

# i++;

# }

# qsort (words, i, sizeof(char)\*30,compar1);

# if(bsearch(word,words,i,sizeof(char)\*30,compar1))printf("exists");

# else printf("doesn't exist");

# return 0;

# }