**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра математического обеспечения и применения ЭВМ**

отчет

**по лабораторной работе №1**

**по дисциплине «Программирование»**

Тема: Обзор стандартной библиотеки

|  |  |  |
| --- | --- | --- |
| Студент гр. 6303 |  | Филиппенко Д.Р. |
| Преподаватель |  | Берленко Т.А. |

Санкт-Петербург

2017

**Цель работы:** провести обзор стандартной библиотеки,научиться использовать алгоритм быстрой сортировки элементов массива, определить время работы данного алгоритма.

**Задание:**

Написать программу, на вход которой подается массив целых чисел длины 1000.

Программа должна совершать следующие действия:

* отсортировать массив по невозрастанию модулей элементов с помощью алгоритма "быстрая сортировка" (quick sort), используя при этом функцию стандартной библиотеки
* посчитать время, за которое будет совершена сортировка, используя при этом функцию стандартной библиотеки
* вывести отсортированный массив (элементы массива должны быть разделены пробелом)
* вывести время, за которое была совершена быстрая сортировка

**Ход работы:**

1. Подключены необходимые заголовочные файлы:

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

1. Добавлена константа len, задающая количество элементов в массиве (в данном случае 1000 элементов):

**#**define len 1000

1. Создана функция сравнения compare, необходимая в дальнейшей работе с функцией qsort:

int compare(const void \* x1, const void \* x2)

{

return ( \*(int\*)x2 - \*(int\*)x1 );

}

1. В функции main выполняется:

- объявление переменных:

int array[len];

int i;

time\_t sort\_start, sort\_end;

- считывание массива:

for(i=0; i<len; i++)

scanf("%d",&array[i]);

- перед запуском функции быстрого поиска сохранено текущее время в секундах:

sort\_start = time(NULL);

- вызов функции qsort, аргументами которой являются: массив чисел, количество элементов в нем, размер элементов массива в байтах и compare:

qsort(array, 1000, sizeof(int), compare);

- записано время окончания поиска в секундах:

sort\_end = time(NULL);

- вывод элементов отсортированного массива на экран:

for(i=0; i<len; i++)

printf("%d ",array[i]);

- вывод времени сортировки на экран:

printf("\n%f", difftime(sort\_end, sort\_start));

1. При помощи консоли на github были выполнены:

- создание новой ветки:

git checkout -b Filippenko\_Daria\_lr1

- создание новой директории:

mkdir Filippenko\_Daria\_lr1

- добавление файла в директорию:

git add Filippenko\_Daria\_lr1/lr1.c

git commit -m “Filippenko lr1 done”

git push origin Filippenko\_Daria\_lr

- таким же образом загружен отчет по лабораторной работе.

**Вывод:** в ходе выполнения работы были получены навыки работы с функциями стандартной библиотеки: быстрого поиска qsort и подсчета времени выполнения программы.