**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. Введена переменная определяющая длину массива:

#define len 1000

1. Создана функция сравнения, которая в дальнейшем потребуется для сортировки массива. Она сравнивает модули двух элементов по невозрастанию:

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

{

return ( abs(\*(int\*)x2) – abs(\*(int\*)x1 ));

}

1. Объявлены переменные:

int array[len];

int i;

clock\_t sort\_start, sort\_end;

1. Считывание массива:

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

{

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

}

1. В переменную sort\_start записывание время в тактах до запуска сортировки:

sort\_start = clock();

1. Запускаем функцию стандартной библиотеки qsort. Функции передаются значения: указатель на начало массива, количество элементов в нем, размер элементов массива и функция сравнение:

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

1. В переменную end \_start записывание время в тактах после запуска сортировки:

sort\_end = clock();

1. Вывод на консоль отсортированного массива:

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

{

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

}

1. Вывод на консоль времени, за которое была произведена сортировка массива (в секундах):

printf("\n%f", (float)(sort\_end - sort\_start)/CLOCKS\_PER\_SEC);

CLOCKS\_PER\_SEC – константа, отвечающая за перевод значения времени в тактах в секунды.

1. При помощи консоли на github были выполнены:

- создание новой ветки:

git checkout -b Filippenko\_sem2\_lr1

- создание новой директории:

mkdir Filippenko\_sem2\_lr1

- добавление файла в директорию:

git add lr1.c

git commit -m “lr1 done”

git push origin Filippenko\_sem2\_lr1

- 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 в тактах выполнения программы clock.