

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

ОТЧЕТ
по лабораторной работе №5
по дисциплине «Программирование»
Тема: Сортировка массива

Студент гр. 6304

Курков Д.В.

Преподаватель

Берленко Т.А.

Санкт-Петербург

2017

Цель работы:

Изучить различные методы сортировки массива, рассмотреть библиотечные функции из `time.h`.

Задание:

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

Программа должна совершать следующие действия:

- отсортировать массив с помощью алгоритма «сортировка пузырьком»
- посчитать время, за которое будет совершена сортировка пузырьком.
- отсортировать массив, используя алгоритм «быстрая сортировка»
- посчитать время, за которая будет совершена быстрая сортировка.
- вывести отсортированный массив.
- вывести время, за которое была совершена сортировка пузырьком
- вывести время, за которое была совершена быстрая сортировка.

Ход работы

1. Создание вспомогательных функций swap и compare.

Функция swap, необходимая для работы алгоритма сортировки пузырьком

```
void swap (int *a, int *b)
{
    int temp = *a;
    *a = *b;
    *b = temp;
}
```

Функция compare, для работы алгоритма быстрой сортировки.

```
int compare (const void * a, const void *b)
{
    return ( *(int*)a - *(int*)b);
}
```

2. Реализация алгоритма сортировки пузырьком.

```
void bubble_sort (int *arr)
{
    for (int i=1; i < N; i++)
    {
        for (int j=0; j < N-i; j++)
        {
            if (*(arr+j) > *(arr+j+1))
            {
                swap (arr+j, arr+j+1);
            }
        }
    }
}
```

3. Основная функция main с подключенными заголовочными файлами

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

const int N = 1000;

void bubble_sort (int *);

int main ()
{
    int arr1 [N], arr2 [N];
    for (int i = 0; i < N ;i++)
    {
        scanf ("%d", arr1+i);
    }
    clock_t t1, t2;
    time (&t1);
    t1 = clock();
    bubble_sort(arr1);
    t2 = clock();
    t2-=t1;
    clock_t t3, t4;
    t3 = clock ();
    qsort (arr2, N, sizeof(int), compare);
    t4 = clock ();
    t4-=t3;
    for (int i=0; i < N; i++)
        printf ("%d ", *(arr1+i));
    printf("\n%f\n%f", (double)t2/CLOCKS_PER_SEC,
        (double)t4/CLOCKS_PER_SEC);
    return 0;
}
```

4. Программа проверена на работоспособность.

5. Текст программы помещен в репозиторий группы.

Вывод: В данной лабораторной работе были усвоены особенности работы с функциями из библиотеки `time.h`. Рассмотрены различные типы сортировки массивом. Опытным путем установлено, что быстрая сортировка гораздо эффективнее, чем сортировка пузырьком.