**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МОЭВМ**

отчет

**по лабораторной работе №1**

**по дисциплине «Программирование»**

Тема: Обзор стандартной библиотеки

|  |  |  |
| --- | --- | --- |
| Студентка гр. 6304 |  | Иванкова В.М. |
| Преподаватель |  | Берленко Т.А. |

Санкт-Петербург

2017

# **Цель:**

Написать программу, на вход которой подается массив целых чисел длины 1000. Реализовать сортировку массива с помощью алгоритмов "сортировка пузырьком" и «быстрая сортировка». Подсчитать время работы обоих алгоритмов. Вывести отсортированный массив, элементы которого будут разделены пробелом. Вывести время работы обоих алгоритмов.

# **Задание:**

Напишите 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, используя при этом **функцию стандартной библиотеки**
* отсортировать массив 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 сортировка пузырьком
* вывести время, за которое была совершена быстрая сортировка

# **Ход работы:**

Код программы:

**#include <stdio.h>**

**#include <stdlib.h>**

**#include <string.h>**

**#include <time.h>**

**void bubble\_sort(int \*array, int number) //**Алгоритм «сортировка пузырьком»

**{**

**int element;**

**for (int i=0;i<number;i++)**

**{**

**for(int j=number-2;j>=i;j--)**

**if(array[j]>array[j+1])**

**{**

**element=array[j];**

**array[j]=array[j+1];**

**array[j+1]=element;**

**}**

**}**

**}**

**int cmp(const void \*first, const void \*second) //**Алгоритм «быстрая сортировка»

**{**

**return(\*(int\*)first-\*(int\*)second);**

**}**

**int main()**

**{**

**int \*array;**

**int number=1000;**

**array=(int\*)malloc(sizeof(int)\*number);**

**for(int i=0;i<number;i++)**

**{**

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

**}**

**int \*save=(int\*)malloc(sizeof(int)\*number); //**Создаем копию заданного массива

**for(int i=0;i<number;i++)**

**{**

**save[i]=array[i];**

**}**

**time\_t a = clock(); //**Функция стандартной библиотеки, которая посчитывает время, за которое будет совершена сортировка

**bubble\_sort(array, number);**

**a=clock()-a;**

**time\_t b = clock();**

**qsort(save, number, sizeof(int), cmp);**

**b=clock()-b;**

**for(int i=0;i<number;i++)**

**{**

**printf("%d", array[i]); //**Выводим отсортированный массив

**}**

**printf("\nTime of bebble=%d", a); //**Выводим время работы алгоритма «сортировка пузырьком»

**printf("\nTime of Qsort=%d", b); //**Выводим время работы алгоритма «быстрая сортировка»

**return 0;**

**}**

# **Вывод:**

В ходе данной работы успешно отсортирован и выведен массив, также выведены с новой строки время работы алгоритма «сортировка пузырьком» и время работы алгоритма «быстрая сортировка».