Министерство образования и науки Российской Федерации

Севастопольский государственный университет

Кафедра ИС

Отчет

по лабораторной работе № 2

«Математический анализ алгоритмов»

Выполнил

ст. гр. И12д

Серегин А.В.

Проверил:

ст.пр. Шишкевич В.Е.

Севастополь

2015

1. Цель работы

Лабораторная работа посвящена экспериментальной проверке теоретической оценки трудоемкости алгоритма поиска минимума и включает ознакомление с принципами использования генератора случайных чисел для создания наборов исходных данных.

2. Варианты заданий

Вариант 6: промежуток чисел 0-800, длины массива: 100, 1000, 2000.

3. Структурная схема

На рисунках ниже представлена структурная схема алгоритма программы.

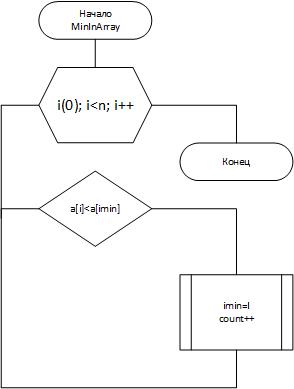


Рисунок 1 – функция MinInArray.

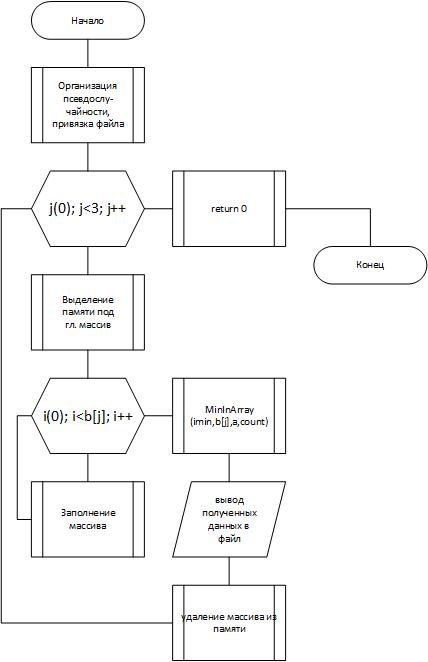


Рисунок 2 – функция main.

4. Код программы

#include <iomanip> //для setpresition()

#include <fstream> //для файлового ввода вывдода

#include <ctime> //time(NULL), для организации рандома

using namespace std;

//функция нахождения индекса минимального значения в массиве

void MinInArray(int& imin, const int& n,const int \*a, int& count)

{

for (int i(0); i < n; i++) // цикл перебора эл-тов массива

if (a[i] < a[imin]) // сравнение эл-тов

{

imin = i;

count++; //щетчик оп-ций переприсваивания

}

}

// основная функция программы

int main()

{

setlocale(LC\_ALL, "Russian"); //отбр. кириллицы

srand(time(NULL)); //вкл. псевдорандома

ofstream fout("TA\_lab2.txt", ios\_base::app); //привязка файла

int \*a;

int b[3] = { 100, 1000, 2000 }; //массив хранящий длинну основного массива

for (int j(0); j < 3; j++) //цикл смены массивов и их длинн

{

a = new int[b[j]]; //выделение памяти под массив

fout << "Массив состоит из " << b[j] << " элементов." << endl;

for (int i(0); i < b[j]; i++) //цикл заполнения массива

{

a[i] = rand() % 801;

fout << a[i] << ' ';

}

int count = 0; //cчетчик переприсваиваний

int imin = 0; //индекс мин. эл-та массива

MinInArray(imin, b[j], a, count); //поиск imin

fout << endl;

fout << "Мин. элемент массива: " << a[imin] << endl;

fout << "Кол-во переприсваиваний при поиске:" << count << endl;

fout << setprecision(2) << "n-е гармоничное число: Hn=" << log(b[j]) + 0.57 << endl;

fout << endl;

delete [] a; //удаление массива из памяти

}

system("pause");

return 0;

}

5. Результаты работы программы.

Тест 1 – массив из 100 элементов :

Мин. элемент массива: 2

Кол-во переприсваиваний при поиске:5

n-е гармоничное число: Hn=5.2

Тест 2 – массив из 1000 элементов :

Мин. элемент массива: 0

Кол-во переприсваиваний при поиске:6

n-е гармоничное число: Hn=7.5

Тест 3 – массив из 2000 элементов:

Мин. элемент массива: 0

Кол-во переприсваиваний при поиске:2

n-е гармоничное число: Hn=8.2

Расчеты n-го гармоничного числа совпали с результатами программы.

Выводы

В ходе лабораторной работы мы заполнили массив случайными числами, нашли минимальный элемент массива и научились считать количество выполняемых операций программой, написанной на высокоуровневом алгоритмическом языке программирования.