Министерство образования и науки Российской Федерации

Севастопольский государственный университет

Кафедра ИС

Отчет

по лабораторной работе № 3

«Математический анализ алгоритмов»

Выполнил

ст. гр. И12д

Серегин А.В.

Проверил:

ст.пр. Шишкевич В.Е.

Севастополь

2015

1. Цель работы

Экспериментальная проверка теоретически полученной функции трудоемкости для алгоритма точного решения задачи о сумме методом полного перебора. Получение практических навыков расстановки счетчика операций в программе на языке высокого уровня.

2. Варианты заданий

Вариант 6.

Размер вектора: 8.

Максимальное случайное число 35.

Значение суммы : 134.

3. Структурная схема

Ниже представлена структурная схема программы.

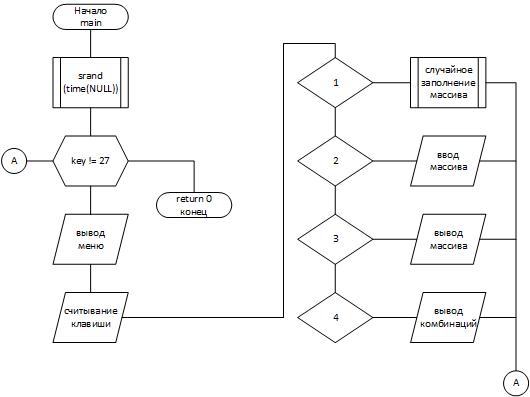


Рисунок 1 – Функция main.

4. Код программы

main.cpp

#include <iostream>

#include <time.h>

#include <conio.h>

#include "Array.h"

using namespace std;

int main()

{

setlocale(LC\_ALL, "Rus");

srand(time(NULL));

Combo combo;

char key;

int n, sum , max;

do

{

system("cls");

cout << "1.Заполнить массив случайными числами." << endl;

cout << "2.Считать массив." << endl;

cout << "3.Отобразить массив." << endl;

cout << "4.Вывести комбинации." << endl;

cout << "Esc. Exit." << endl;

key = \_getch();

system("cls");

switch (key)

{

case '1':

cout << "Введите размер массива:" << endl;

cin >> n;

cout << "Введите максимальное значение эл-та:" << endl;

cin >> max;

combo.RandArray(n, max);

break;

case '2':

cout << "Введите размер массива:" << endl;

cin >> n;

combo.ScanArray(n);

break;

case '3':

combo.PrintArray();

system("pause");

break;

case '4':

cout << "Введите сумму эл-тов:" << endl;

cin >> sum;

combo.Print\_Combo(sum);

system("pause");

break;

}

} while (key != 27);

system("pause");

}

Array.h

#pragma once

#include <vector>

class Array

{

protected:

std::vector<int> arr;

public:

void RandArray(const int, const int);

void ScanArray(const int);

void PrintArray();

};

class Combo : public Array

{

private:

std::vector<std::vector<int>> combo;

public:

void Print\_Combo(int);

};

Array.cpp

#include "Array.h"

#include <iostream>

void Array::RandArray(const int n, const int rand\_max)

{

arr.clear();

for (int i(0); i < n; i++)

{

arr.push\_back(rand() % (rand\_max+1));

}

}

void Array::ScanArray(const int n)

{

arr.clear();

int temp;

std::cout << "Введите массив:" << std::endl;

for (int i(0); i < n; i++)

{

std::cin >> temp;

arr.push\_back(temp);

}

}

void Array::PrintArray()

{

if (arr.empty())

{

std::cout << "Массив пуст." << std::endl;

return;

}

std::cout << "Массив:" << std::endl;

for (auto a : arr)

std::cout << a << ' ';

std::cout << std::endl;

}

void Combo::Print\_Combo(int sum)

{

int s, k = 0;

combo.clear();

for (int i(0); i < (1 << arr.size()); i++)

{

s = 0;

combo.resize(combo.size() + 1);

for (int j(0); j < arr.size(); j++)

if (i + 1 & (1 << j))

{

combo.back().push\_back(arr[j]);

s = s + arr[j];

}

if (s != sum)

combo.pop\_back();

}

if (combo.empty())

{

std::cout << "Комбинаций не существует." << std::endl;

return;

}

std::cout << "Список комбинаций:" << std::endl;

for (int i(0); i < combo.size(); i++)

{

for (auto a : combo.at(i))

std::cout << a << ' ';

std::cout << std::endl;

}

}

5. Тестовые примеры

На рисунках ниже представлены тестовые примеры работы программы.

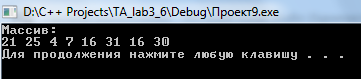


Рисунок 2 – Случайный массив.

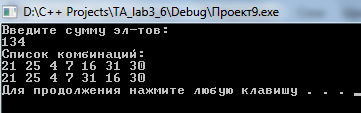


Рисунок 3 – Комбинации суммы.

Выводы

В ходе лабораторной работы мы научились находить в массиве числа, сумма которых будет равна задаваемому числу, с помощью метода полного перебора.