**Домашняя работа №6**

**Задание:**

1:

Написать программу, которая позволяет пользователю с помощью встроенной функции sort отсортировать вектор, состоящий из 10 элементов, по возрастанию.

Вывести отсортированный вектор в консоль.

2:

Создать и заполнить массив элементами с клавиатуры.

Выполнить сортировку массива с помощью "сортировки пузырьком".

Вывести на экран массив после сортировки.

**Код:**

#include <algorithm>

#include <iostream>

#include <vector>

#include <cstdlib>

#include <ctime>

using namespace std;

void deafult\_sort() {

srand(time(0));

vector<int> default\_vec;

for (int i = 0; i <= 10; i++) {

default\_vec.push\_back(rand() % 100);

}

cout << "Неотсортированный вектор: ";

for (const auto& elem : default\_vec) {

cout << elem << " ";

}

cout << endl;

sort(default\_vec.begin(), default\_vec.end());

cout << "Отсортированный вектор: ";

for (const auto& elem : default\_vec) {

cout << elem << " ";

}

cout << endl;

}

void bubble\_sort() {

vector<int> bubble\_vec;

int count = 0;

int size;

cout << "Вводите числа, тем самым заполняя массив. "

<< "Если захотите прекратить, то просто введите другой тип данных" << endl;

cout << "1 элемент: ";

while (cin >> size) {

count += 1;

cout << count + 1 << " элемент: ";

bubble\_vec.push\_back(size);

}

if (count == 0) { cout << "Вы не ввели ни 1 элемента массива!"; }

else {

cout << "Неотсортированный вектор: ";

for (const auto& elem : bubble\_vec) {

cout << elem << " ";

}

cout << endl;

cout << "Отсортированный вектор: ";

for (int i = 0; i < bubble\_vec.size(); i++) {

for (int j = 1; j < bubble\_vec.size() - i; j++) {

if (bubble\_vec[j - 1] > bubble\_vec[j]) { swap(bubble\_vec[j], bubble\_vec[j - 1]); }

}

}

for (const auto& elem : bubble\_vec) {

cout << elem << " ";

}

cout << endl;

}

}

ы

int main() {

setlocale(LC\_ALL, "RU");

cout << "С помощью sort: \n";

deafult\_sort();

cout << "\nС помощью пузырьковой молитвы богу пузырьков: \n";

bubble\_sort();

}

**Вывод:**

