Министерство транспорта РФ

Федеральное государственное автономное

образовательное учреждение высшего образования

«Российский университет транспорта» (МИИТ)

Лабораторная работа №2

«Алгоритм сортировки вставкой»

по дисциплине

«Методы программирования»

Выполнил:

студент группы ТКИ-311,

Петросян А. А.

Проверил:

к.т.н., доц. Логинова Л. Н,

к.т.н., доц. Сафронов А. И.

Москва – 2022 г.

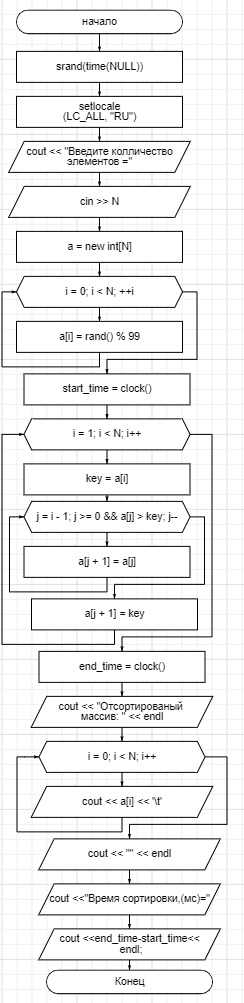
**Цель работы:**

Изучение алгоритма сортировки вставкой.

**Задача:**

Реализовать программу, выполняющую алгоритм сортировки вставкой.

**Блок схема кода:**



**Тестовые примеры:**

Вводим массив длинной в 7 элементов.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Ввод | 5 | 6 | 8 | 2 | 12 | 45 | 3 |
| Вывод | 2 | 3 | 5 | 6 | 8 | 12 | 45 |

**Код программы:**

#include <iostream>

#include <ctime>

using namespace std;

int main()

{

int \*a;

int key, N, i, j;

srand(time(NULL));

setlocale(LC\_ALL, "RU");

cout << "Введите колличество элементов =" ;

cin >> N;

a = new int[N];

for (i = 0; i < N; ++i)

a[i] = rand() % 99;

float start\_time, end\_time;

start\_time = clock();

for (i = 1; i < N; i++)

{

key = a[i];

for (j = i - 1; j >= 0 && a[j] > key; j--)

a[j + 1] = a[j];

a[j + 1] = key;

}

end\_time = clock();

cout << "Отсортированый массив: " << endl;

for (int i = 0; i < N; i++)

cout << a[i] << '\t';

cout << "" << endl;

cout <<"Время сортировки,(мс)=";

cout <<end\_time-start\_time<< endl;

}

**Результаты работы:**

Изображение выглядит как текст, устройство, счетчик, панель управления

Автоматически созданное описание

Результат совпадает с тестовым примером.

Проверим скорость выполнения сортировки для 5.000 и 50.000 элементов.

Изображение выглядит как текст

Автоматически созданное описание

Изображение выглядит как текст, оранжевый

Автоматически созданное описание

**Вывод:**

Я изучил и создал программу для реализации алгоритма сортировки вставкой. Программа работает правильно, выдаёт ожидаемые результаты. Алгоритм прост в реализации. Данный метод применим случае необходимости сортировки малого объема данных. В случае большого количества данных данный метод требует достаточно много времени.