Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет»**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

Направление подготовки: Разработка информационных систем (РИС)

**Отчёт сортировки**

Выполнил студент гр. РИС-24-3б

Носков Егор

Проверил:

Доц. каф. ИТАС

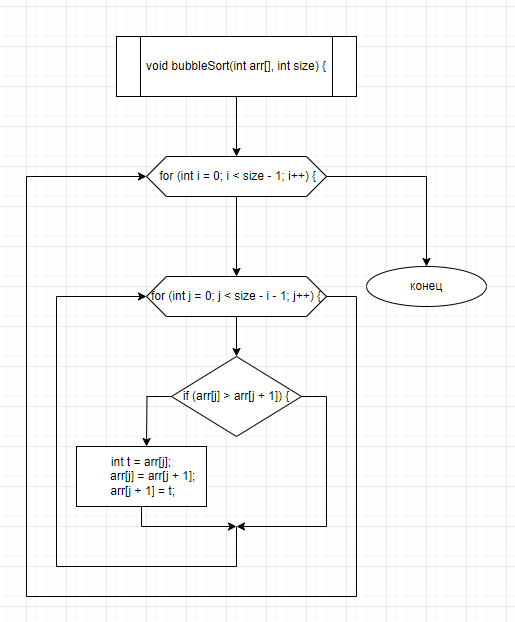
Ольга Андреевна Полякова

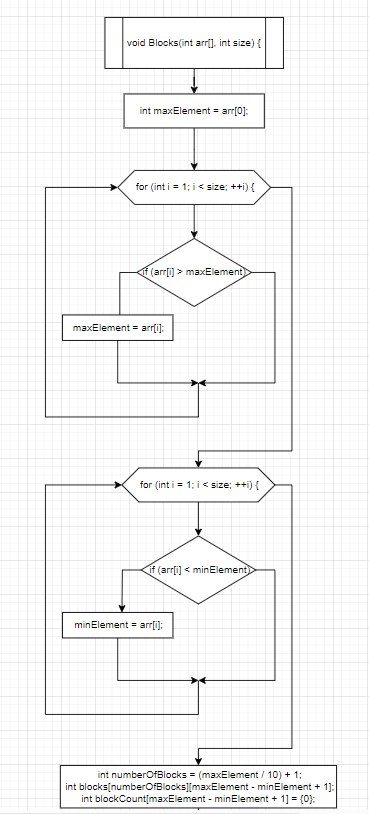
г. Пермь, 2024

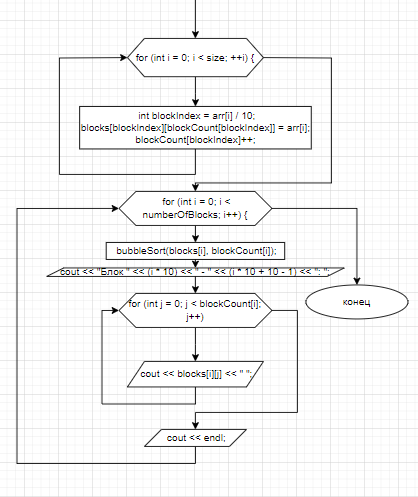
Постановка задачи

Сортировка подсчётом, блочная и слиянием

Блочная сортировка







#include <iostream>

using namespace std;

void bubbleSort(int arr[], int size) {

for (int i = 0; i < size - 1; i++) {

for (int j = 0; j < size - i - 1; j++) {

if (arr[j] > arr[j + 1]) {

int t = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = t;

}

}

}

}

void Blocks(int arr[], int size) {

int maxElement = arr[0];

for (int i = 1; i < size; i++) {

if (arr[i] > maxElement) {

maxElement = arr[i];

}

}

int minElement = arr[0];

for (int i = 1; i < size; i++) {

if (arr[i] < maxElement) {

minElement = arr[i];

}

}

int numberOfBlocks = (maxElement / 10) + 1;

int blocks[numberOfBlocks][maxElement - minElement + 1];

int blockCount[maxElement - minElement + 1] = {0};

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

int blockIndex = arr[i] / 10;

blocks[blockIndex][blockCount[blockIndex]] = arr[i];

blockCount[blockIndex]++;

}

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

bubbleSort(blocks[i], blockCount[i]);

cout << "Блок " << (i \* 10) << " - " << (i \* 10 + 10 - 1) << ": ";

for (int j = 0; j < blockCount[i]; j++) {

cout << blocks[i][j] << " ";

}

cout << endl;

}

}

int main() {

system("chcp 1251 > NULL");

int arr[] = {5, 12, 3, 17, 8, 10, 1, 15, 20, 25, 31, 45, 9, 4, 0};

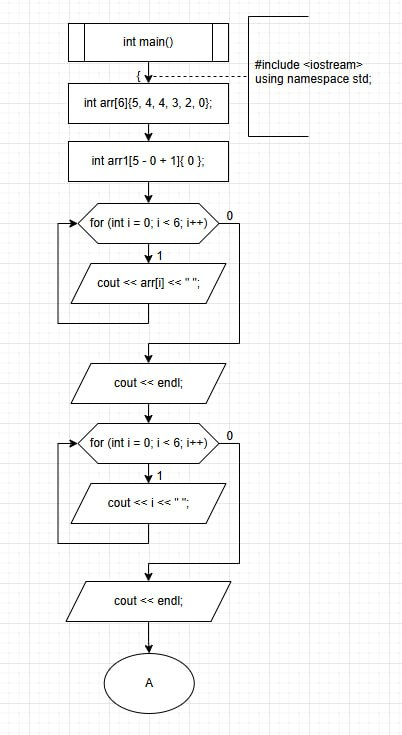
int size = sizeof(arr) / sizeof(arr[0]);

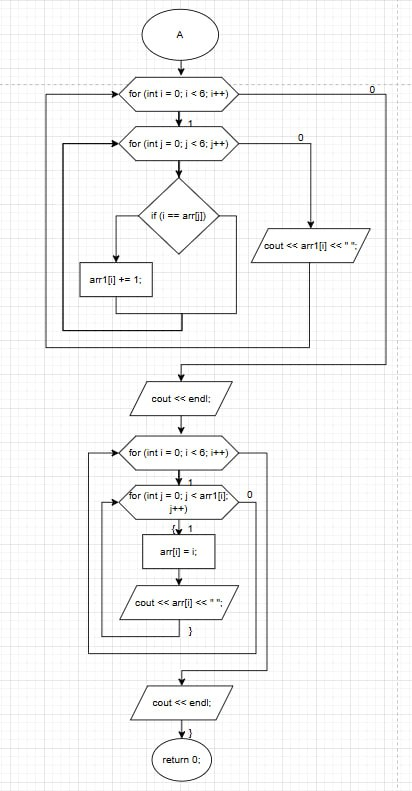
Blocks(arr, size);

return 0;

}

Сортировка подчетом





#include <iostream>

using namespace std;

int main()

{

int arr[6]{5, 4, 4, 3, 2, 0};

int arr1[5 - 0 + 1]{ 0 };

for (int i = 0; i < 6; i++) {

cout << arr[i] << " ";

}

cout << endl;

for (int i = 0; i < 6; i++) {

cout << i << " ";

}

cout << endl;

for (int i = 0; i < 6; i++) {

for (int j = 0; j < 6; j++) {

if (i == arr[j]) arr1[i] += 1;

}

cout << arr1[i] << " ";

}

cout << endl;

for (int i = 0; i < 6; i++) {

for (int j = 0; j < arr1[i]; j++) {

arr[i] = i;

cout << arr[i] << " ";

}

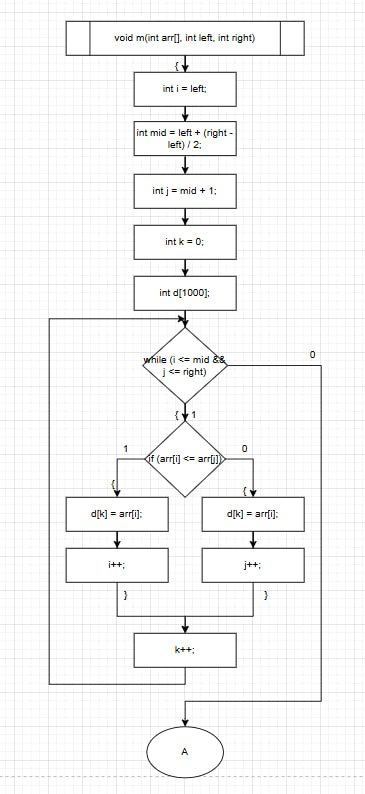
}

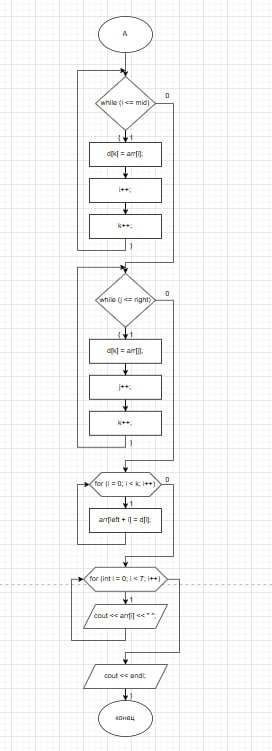
cout << endl;

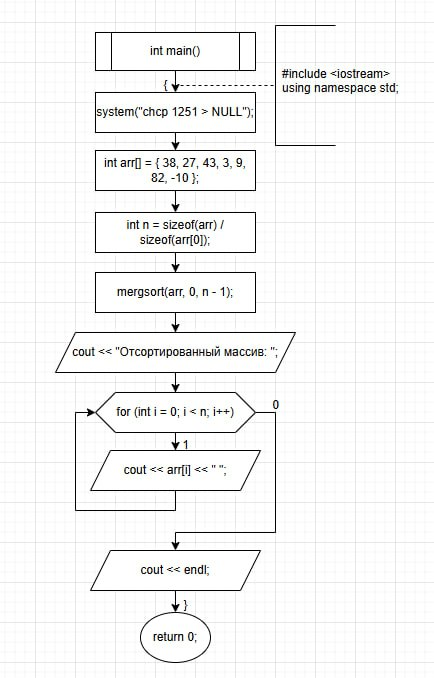
return 0;

}

Сортировка слиянием







#include <iostream>

using namespace std;

void m(int arr[], int left, int right) {

int i = left;

int mid = left + (right - left) / 2;

int j = mid + 1;

int k = 0;

int d[1000];

while (i <= mid && j <= right) {

if (arr[i] <= arr[j]) {

d[k] = arr[i];

i++;

}

else {

d[k] = arr[j];

j++;

}

k++;

}

while (i <= mid) {

d[k] = arr[i];

i++;

k++;

}

while (j <= right) {

d[k] = arr[j];

j++;

k++;

}

for (i = 0; i < k; i++) {

arr[left + i] = d[i];

}

for (int i = 0; i < 7; i++) {

cout << arr[i] << " ";

}

cout << endl;

}

void mergsort(int arr[], int left, int right) {

if (left < right) {

int mid = left + (right - left) / 2;

mergsort(arr, left, mid);

mergsort(arr, mid + 1, right);

m(arr, left, right);

}

}

int main()

{

system("chcp 1251 > NULL");

int arr[] = { 38, 27, 43, 3, 9, 82, -10 };

int n = sizeof(arr) / sizeof(arr[0]);

mergsort(arr, 0, n - 1);

cout << "Отсортированный массив: ";

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

cout << arr[i] << " ";

}

cout << endl;

return 0;

}

****