**Міністерство освіти і науки України**

**Харківський національний університет радіоелектроніки**

**Кафедра Інформатики**

**Звіт**

з лабораторної роботи №5

з дисципліни *«Програмування»*

*Виконав:*

Самченко С. О.

*Перевірив:*

Бродецький Ф.А.

Харків

2020

21. Створити функцію, яка сортує елементи матриці за зростанням для кожного стовпця і знаходить мінімальний елемент.

#include <iostream>

#include <string.h>

using namespace std;

void F(int\* matr, int n) {

int min = matr[0 \* n + 0];

for (int i = 1; i < n; i++)

{

for (int j = 0; j < n; j++)

{

if (min > matr[i \* n + j]) min = matr[i \* n + j];

}

}

cout << "Сортировка всех столбцов матрицы по наростанию" << endl;

for (int k = 0; k < n; k++)

for (int i = 0; i < n - 1 - k; i++)

for (int j = 0; j < n; j++)

if (matr[(i + 1) \* n + j] < matr[i \* n + j]) {

int t = matr[(i + 1) \* n + j];

matr[(i + 1) \* n + j] = matr[i \* n + j];

matr[i \* n + j] = t;

}

for (int i = 0; i < n; ++i, cout << endl)

for (int j = 0; j < n; ++j)

cout << matr[i \* n + j] << "\t";

cout << endl;

cout << "Нахождение минимального элемента матрицы";

cout << "\nMins: " << min << endl;

cout << endl;

}

int main() {

setlocale(LC\_ALL, "ru");

const int n = 5;

int matr[n][n] = {3, 6, 9, 22, 12, 34, 1, 6, 6, 11, 87, 4, 12, 13, 15, 22, 65, 2, 9, 3, 6, 25, 76, 8, 21};

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

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

cout << matr[i][j] << "\t";

}

}

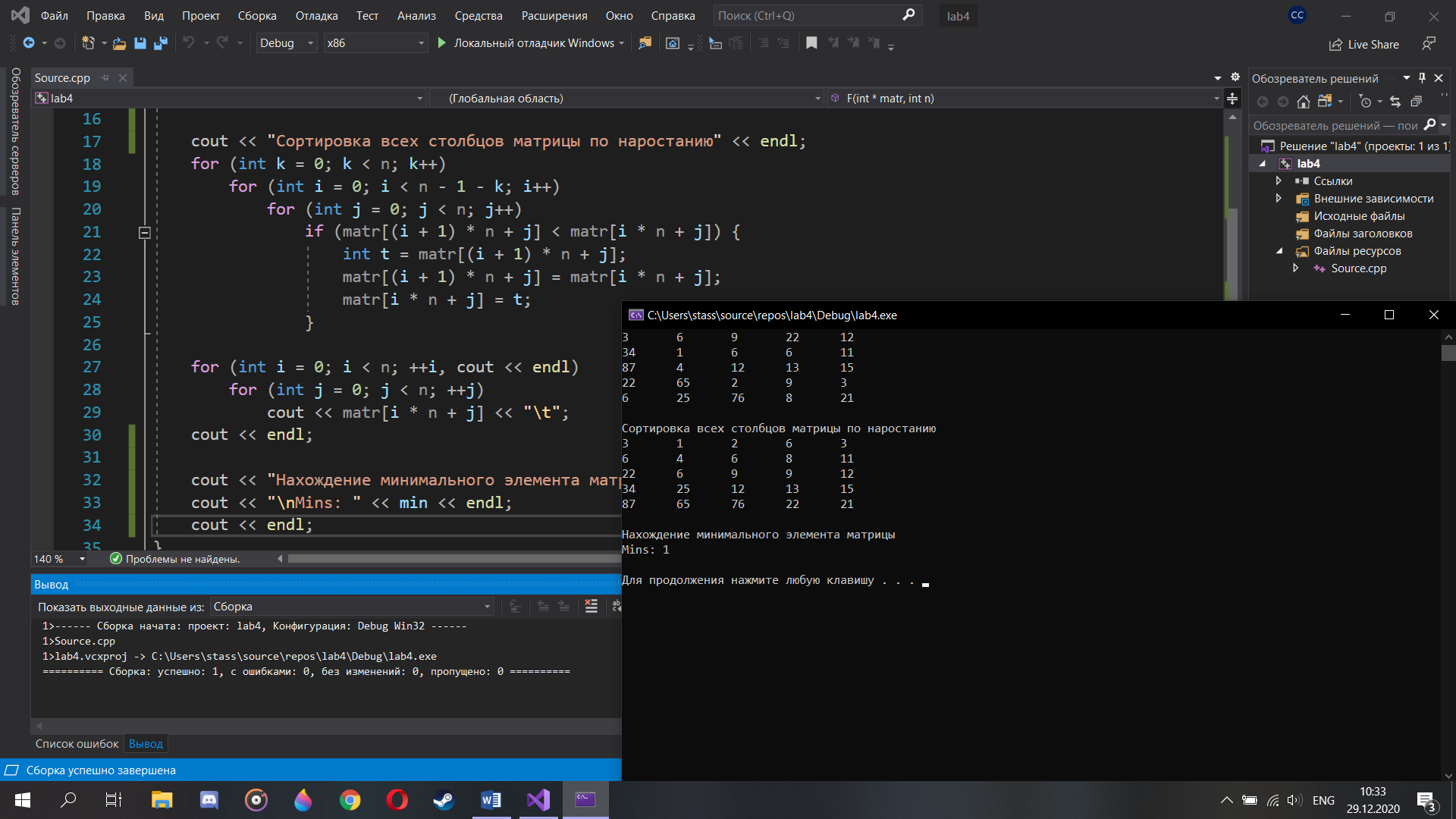
cout << endl;

F(&matr[0][0], n);

system("pause");

return 0;

}



#include <iostream>

#include <string.h>

using namespace std;

void Matr(int\* matr, int n) {

cout << "Заполнение матрицы случайными числами" << endl;

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

for (int j = 0; j < n; ++j)

matr[i \* n + j] = rand() % 21 - 10;

for (int i = 0; i < n; ++i, cout << endl)

for (int j = 0; j < n; ++j)

cout << matr[i \* n + j] << "\t";

cout << endl;

}

void Sort(int\* matr, int n) {

cout << "Сортировка всех столбцов матрицы по наростанию" << endl;

for (int k = 0; k < n; k++)

for (int i = 0; i < n - 1 - k; i++)

for (int j = 0; j < n; j++)

if (matr[(i + 1) \* n + j] < matr[i \* n + j]) {

int t = matr[(i + 1) \* n + j];

matr[(i + 1) \* n + j] = matr[i \* n + j];

matr[i \* n + j] = t;

}

for (int i = 0; i < n; ++i, cout << endl)

for (int j = 0; j < n; ++j)

cout << matr[i \* n + j] << "\t";

cout << endl;

}

void Min(int\* matr, int n) {

cout << "Нахождение минимального элемента матрицы";

int min = matr[0 \* n + 0];

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

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

if (min > matr[i \* n + j]) min = matr[i \* n + j];

}

}

cout << "\nMins: " << min << endl;

cout << endl;

}

int main() {

setlocale(LC\_ALL, "ru");

const int n = 5;

int matr[n][n];

Matr(&matr[0][0], n);

Min(&matr[0][0], n);

Sort(&matr[0][0], n);

system("pause");

return 0;

}

