Варіант 21 (1)

Створити програму, яка вводить c клавіатури (з обробкою помилок) двовимірний квадратний масив цілих чисел, замінює всі негативні елементи головної діагоналі на суму всіх елементів відповідного рядка і відображає даний масив на екрані в транспоновану вигляді.

a [1,1] a [2,1] a [3,1]

a [1,2] a [2,2] ...

#include <iostream>

#include <vector>

#include <ctime>

using namespace std;

void P(vector<vector<int>> matrix) {

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

for (int j = 0; j < matrix[i].size(); j++) {

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

}

cout << endl << endl;

}

}

int main() {

setlocale(LC\_ALL, "ru");

srand(time(NULL));

int n;

try {

double temp;

cout << "Введите размер матрицы: ";

cin >> temp;

if (temp - (int)temp > 0) { throw "размер матрицы не может быть дробным"; }

else if (cin.fail()) { throw "не возможно создать матрицу такого типа "; }

else n = temp;

if (n <= 0.0) { throw "невозможно создать матрицу такого размера"; }

}

catch (const char\* exeption) { cout << "\*ERROR\* " << exeption << endl; return 0; }

vector<vector<int>> matrix;

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

vector<int> row;

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

row.push\_back(rand() % 10 - 1);

}

matrix.push\_back(row);

}

cout << endl << "-\*- МАТРИЦА -\*-" << endl;

P(matrix);

cout << endl;

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

for (int j = 0; j < matrix[i].size(); j++) {

if (matrix[i][i] < 0) {

int sum = 0;

for (int k = 0; k < matrix[i].size(); k++) {

sum += matrix[i][k];

}

matrix[i][i] = sum;

}

}

}

cout << endl << "-\*- ЗАМЕНА ОТРИЦАТЕЛЬНЫХ ЭЛЕМЕНТОВ ГЛАВНОЙ ДИАГОНАЛИ -\*-" << endl;

P(matrix);

vector<vector<int>> temp\_matrix;

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

vector<int> row;

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

row.push\_back(0);

}

temp\_matrix.push\_back(row);

}

cout << endl;

int k = 0, p = 0;

for (int i = 0; i < temp\_matrix.size(); i++) {

k = 0;

for (int j = 0; j < temp\_matrix[i].size(); j++) {

temp\_matrix[i][j] = matrix[k++][p];

}

p++;

}

cout << endl << "-\*- ИТОГОВЫЙ РЕЗУЛЬТАТ -\*-" << endl;

P(temp\_matrix);

system("pause");

return 0;

}

