Министерство образования и науки Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение**

**высшего профессионального образования**

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

**(ВлГУ)**

**Кафедра информационных систем и программной инженерии**

Лабораторная работа №5

по дисциплине «Основы алгоритмизации и

программирования»

Тема: «Обработка матриц»

Выполнил:

студент гр. ПРИ-120

Грачев Д.А.

Приняла:

Шамышева О.Н.

Владимир 2020 г.

ЦЕЛЬ РАБОТЫ

Разработка алгоритма обработки матриц, его программирование на языках С++ и Python.

ЗАДАЧА 6 (ВАРИАНТ №6) на Python

На языке Python выполнить задание:

Дана действительная матрица размера m×n, в которой не все

элементы равны нулю. Получить новую матрицу путем деления всех элементов данной матрицы на ее наибольший по модулю элемент.

Листинг программы:

from random import randint

M = 4

N = 3

matrix = [0] \* M

mm = 0

for i in range(M):

matrix[i] = [0] \* N

for j in range(N):

matrix[i][j] = randint(-50,50)

if abs(matrix[i][j]) > abs(mm):

mm = matrix[i][j]

print (matrix[i])

print ("Наибольший по модулю элемент: ", mm)

for i in range(M):

for j in range(N):

matrix[i][j] = round(matrix[i][j] / mm, 3)

for i in range(M):

print (matrix[i])

Скриншот работы программы представлен на Рис. 1.

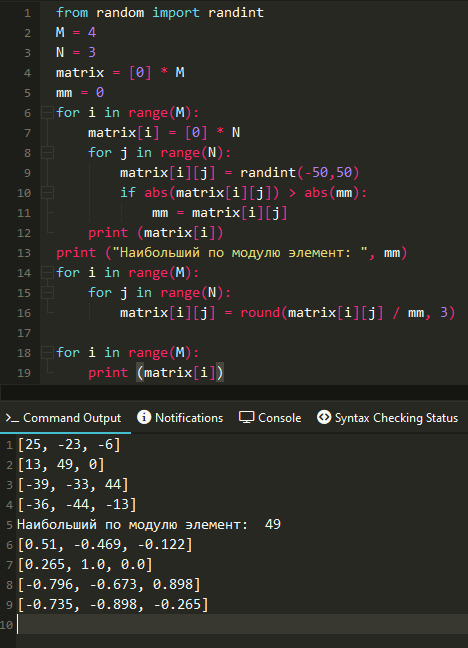


Рисунок 1. Скриншот работы программы.

ЗАДАЧА 1 (ВАРИАНТ №6) на С++

На языке С++ выполнить задание:

Дана действительная матрица размера m×n, в которой не все

элементы равны нулю. Получить новую матрицу путем деления всех элементов данной матрицы на ее наибольший по модулю элемент.

Листинг программы:

#include <iostream>

#include <cstdlib>

#include <ctime>

#include <cmath>

using namespace std;

double matrix[4][3], mm = matrix[0][0];

int main() {

srand(time(NULL));

for (int i = 0; i < 4; i++) {

for (int j = 0; j < 3; j++) {

matrix[i][j] = rand() % 101 - 50;

cout << matrix[i][j] << ' ';

if (abs(matrix[i][j]) > abs(mm)) {

mm = matrix[i][j];

}

}

cout << endl;

}

cout << "Наибольший по модулю элемент: " << mm << endl;

for (int i = 0; i < 4; i++) {

for (int j = 0; j < 3; j++) {

matrix[i][j] = round((matrix[i][j] / mm)\*1000)/1000;

cout << matrix[i][j] << ' ';

}

cout << endl;

}

return 0;

}

Скриншот работы программы представлен на Рис. 2.



Рисунок 2.Скриншот работы программы.

ЗАДАЧА 2 (Вариант №6)

На языке Python выполнить задание:

Дана действительная матрица размера M×N. Сформировать одномерный массив из средних арифметических значений каждого столбца матрицы.

ВЫПОЛНЕНИЕ РАБОТЫ

1. Системный анализ
   1. Цель разработки

Разработка выполняется в среде Python 3 c учебной целью для получения практических навыков программирования алгоритмов обработки матриц.

* 1. Назначение программы

Программа создает одномерный массив из средних арифметических значений каждого столбца матрицы.

* 1. Методы решения

При решении используются следующие функции:

* Рандомизация(randint)
* Сложение (+)
* Умножение (\*)
  1. Основные требования к выполняемым функциям

Вывести новый массив формате: «Новый массив: [новый\_массив]»

* 1. Набор тестов

Набор тестов к программе представлен на таблице 1.

Таблица 1. Набор тестов

|  |  |
| --- | --- |
| Номер теста | Вывод |
| 1 | [1, 40, 25]  [-26, 42, 44]  [44, -20, -23]  [24, 13, -21]  Новый массив: [10.75, 18.75, 6.25] |
| 2 | [47, 48, -24]  [-50, -49, -20]  [-21, 35, 18]  [8, -28, 43]  Новый массив: [-4.0, 1.5, 4.25] |
| 3 | [-31, -33, 3]  [25, -10, 8]  [-25, -10, 0]  [-21, 10, -5]  Новый массив: [-13.0, -10.75, 1.5] |

1. Структурное проектирование
   1. Структура данных

Данные:

* matrix, sr – выходные данные типа list
  1. Структура программы

Структура программы в виде блок-схемы представлена на Рис.3



Рисунок 3. Блок-схема структуры программы.

* 1. Используемые подпрограммы

Стандартные процедуры ввода-вывода.

Библиотека random.

1. Программирование

Листинг программы:

from random import randint

N = 3

M = 4

sr = [61] \* N

matrix = [[randint(-50,50) for j in range(N)] for i in range(M)]

for i in range(M):

print(matrix[i])

for i in range(N):

for j in range(M):

if sr[i] != 61:

sr[i] += matrix[j][i]

else:

sr[i] = matrix[j][i]

sr[i] /= M

print ('Новый массив:', sr)

1. Тестирование и отладка

В процессе отладки были устранены синтаксические ошибки. Тестирование по тестам п. 1.4. дало ожидаемые результаты.

Скриншот работы программы представлен на Рис.4.

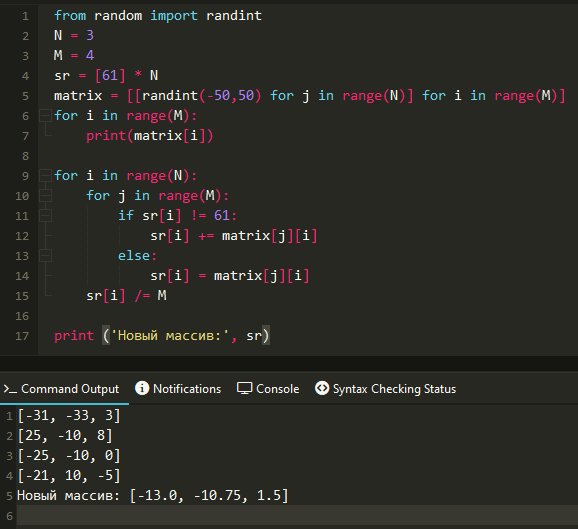


Рисунок 4.Скриншот работы программы.

ЗАДАЧА 2 (ВАРИАНТ №6) на С++

На языке С++ выполнить задание:

Дана действительная матрица размера M×N. Сформировать одномерный массив из средних арифметических значений каждого столбца матрицы.

Листинг программы:

#include <iostream>

#include <cstdlib>

#include <ctime>

#include <cmath>

using namespace std;

double matrix[4][3], sr[3] = {61,61,61};

int main() {

srand(time(NULL));

for (int i = 0; i < 4; i++) {

for (int j = 0; j < 3; j++) {

matrix[i][j] = rand() % 101 - 50;

cout << matrix[i][j] << ' ';

}

cout << endl;

}

cout << "Новый массив: ";

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 4; j++) {

if (sr[i] != 61) {

sr[i] = sr[i] + matrix[j][i];

}

else {

sr[i] = matrix[j][i];

}

}

cout << sr[i] << ' ';

}

return 0;

}

Скриншот работы программы представлен на Рис. 5.

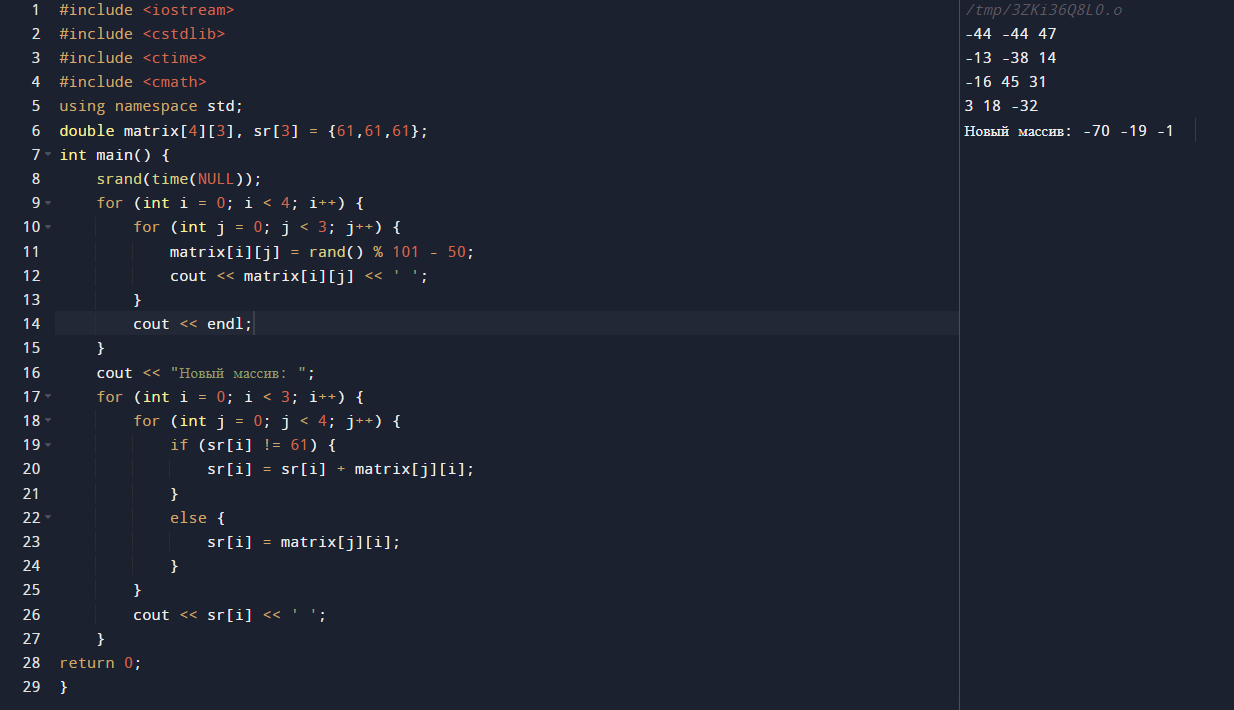


Рисунок 5. Скриншот работы программы.

ЗАДАЧА 3 (ВАРИАНТ №6) на Python

На языке Python выполнить задание:

Задана матрица A(N, N). Найти максимальный по модулю элемент матрицы. Переставить строки и столбцы матрицы таким образом,

чтобы максимальный по модулю элемент был расположен на пересечении

К-ой строки и К-го столбца (1 ≤ K ≤ N).

Листинг программы:

from random import randint

N = 3

matrix = [0] \* N

mm = 0

for i in range(N):

matrix[i] = [0] \* N

for j in range(N):

matrix[i][j] = randint(-50,50)

if abs(matrix[i][j]) > abs(mm):

mm = matrix[i][j]

strk = i

stlb = j

print (matrix[i])

k = randint(1,N-1)

print (k)

if strk != k:

matrix[k],matrix[strk] = matrix[strk],matrix[k]

for i in range(N):

if stlb != k:

matrix[i][k],matrix[i][stlb] = matrix[i][stlb],matrix[i][k]

print (matrix[i])

Скриншот работы программы представлен на Рис.6

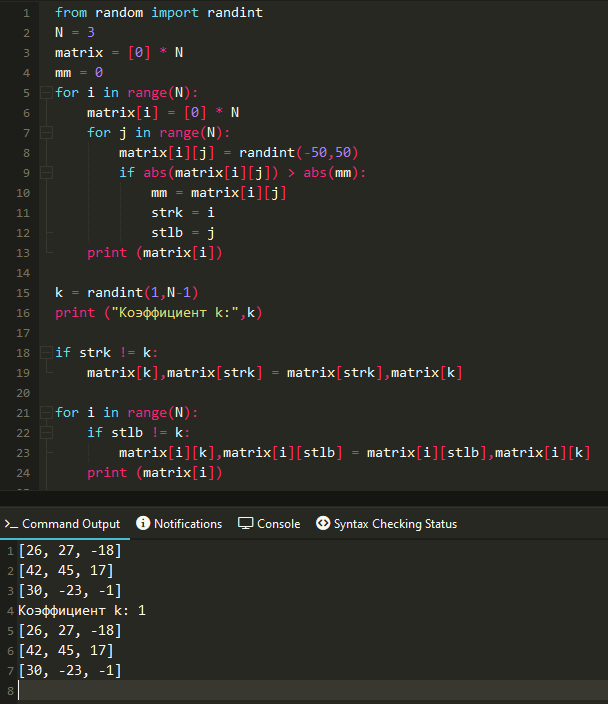


Рисунок 6. Скриншот работы программы.

ЗАДАЧА 3 (ВАРИАНТ №6) на С++

На языке С++ выполнить задание:

Задана матрица A(N, N). Найти максимальный по модулю элемент матрицы. Переставить строки и столбцы матрицы таким образом, чтобы максимальный по модулю элемент был расположен на пересечении К-ой строки и К-го столбца (1 ≤ K ≤ N).

Листинг программы:

#include <iostream>

#include <cstdlib>

#include <ctime>

#include <cmath>

using namespace std;

int a[3][3], t[3];

int sr,sl,k,mm;

int main() {

srand(time(NULL));

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) {

a[i][j] = rand() % 101 - 50;

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

if (abs(a[i][j]) > abs(mm)) {

mm =a[i][j];

sr = i;

sl = j;

}

}

cout << endl;

}

k = rand() % 3;

cout << "k = " << k << endl;

if (sr != k) {

for (int i = 0; i < 3; i++) {

t[i] = a[k][i];

a[k][i] = a[sr][i];

a[sr][i] = t[i];

}

}

for (int i = 0; i < 3; i++) {

if (sl != k) {

t[i] = a[i][k];

a[i][k] = a[i][sl];

a[i][sl] = t[i];

}

for (int j = 0; j < 3; j++) {

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

}

cout << endl;

}

return 0;

}

Скриншот работы программы представлен на Рис.7

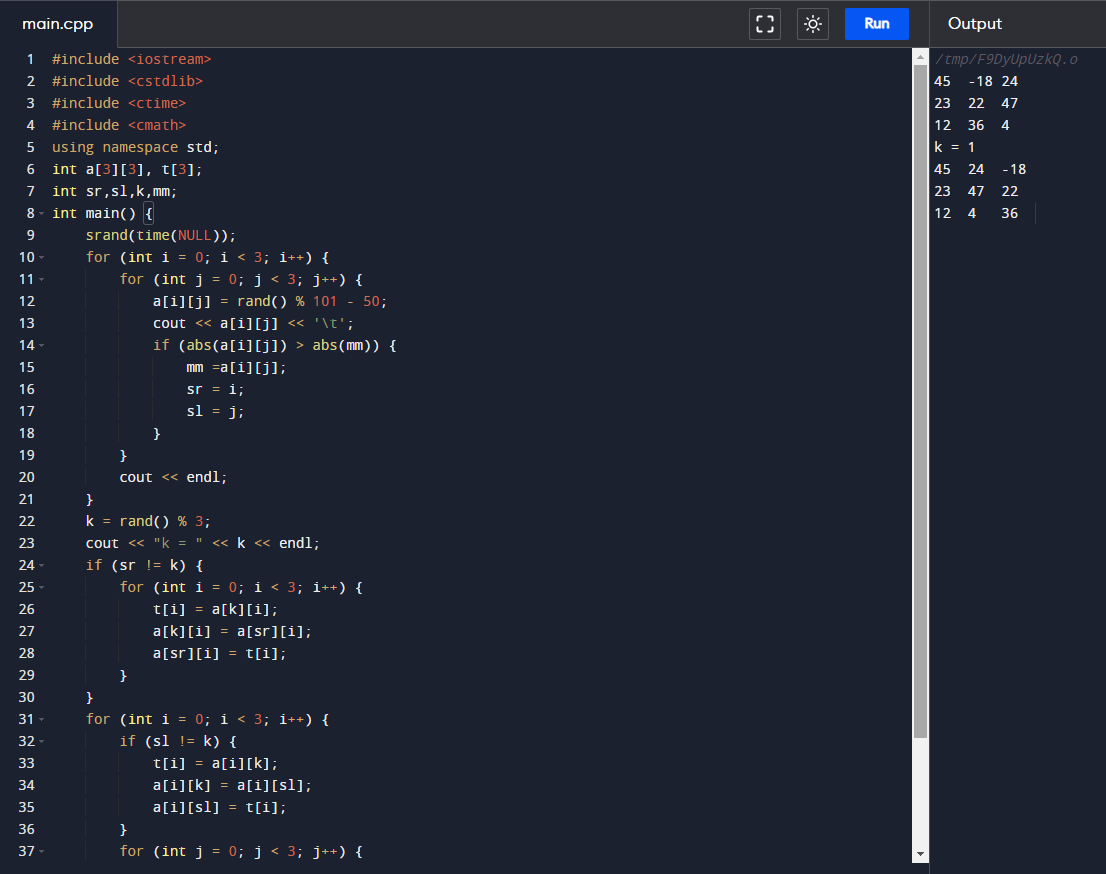


Рисунок 7. Скриншот работы программы.

ЗАДАЧА 4 (ВАРИАНТ №6) на Python

На языке Python выполнить задание:

Дан двумерный массив K(L, N). Определить номер строки этого

массива с максимальным по модулю произведением отрицательных элементов.

Листинг программы:

from random import randint

L = 4

N = 3

pr = 0

matrix = [0] \* L

for i in range(L):

matrix[i] = [0] \* N

mpr = 1

for j in range(N):

matrix[i][j] = randint(-50,50)

if matrix[i][j] < 0:

mpr \*= matrix[i][j]

if mpr != 1:

if abs(mpr) > abs(pr):

ind = i

pr = mpr

print (matrix[i])

print ("Индекс:", ind)

Скриншот работы программы представлен на Рис.8.

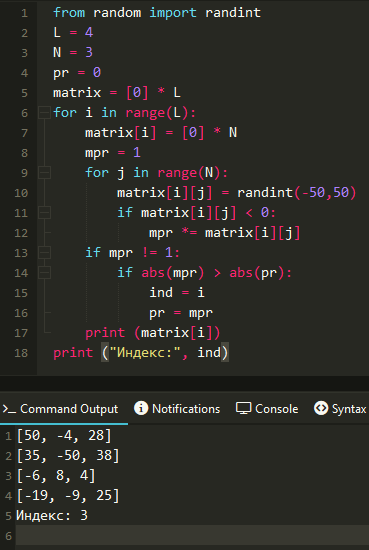


Рисунок 8. Скриншот работы программы.

ЗАДАЧА 4 (ВАРИАНТ №6) на С++

На языке С++ выполнить задание:

Дан двумерный массив K(L, N). Определить номер строки этого массива с максимальным по модулю произведением отрицательных элементов.

Листинг программы:

#include <iostream>

#include <cstdlib>

#include <ctime>

#include <cmath>

using namespace std;

int a[4][3];

int L = 4, N = 3, pr = 0, mpr, ind;

int main() {

srand(time(NULL));

for (int i = 0; i < 4; i++) {

mpr = 1;

for (int j = 0; j < 3; j++) {

a[i][j] = rand() % 101 - 50;

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

if (a[i][j] < 0) {

mpr \*= a[i][j];

}

}

if (mpr != 1) {

if (abs(mpr) > abs(pr)) {

ind = i;

pr = mpr;

}

}

cout << endl;

}

cout << "Index: " << ind;

return 0;

}

Скриншот работы программы представлен на Рис. 9.

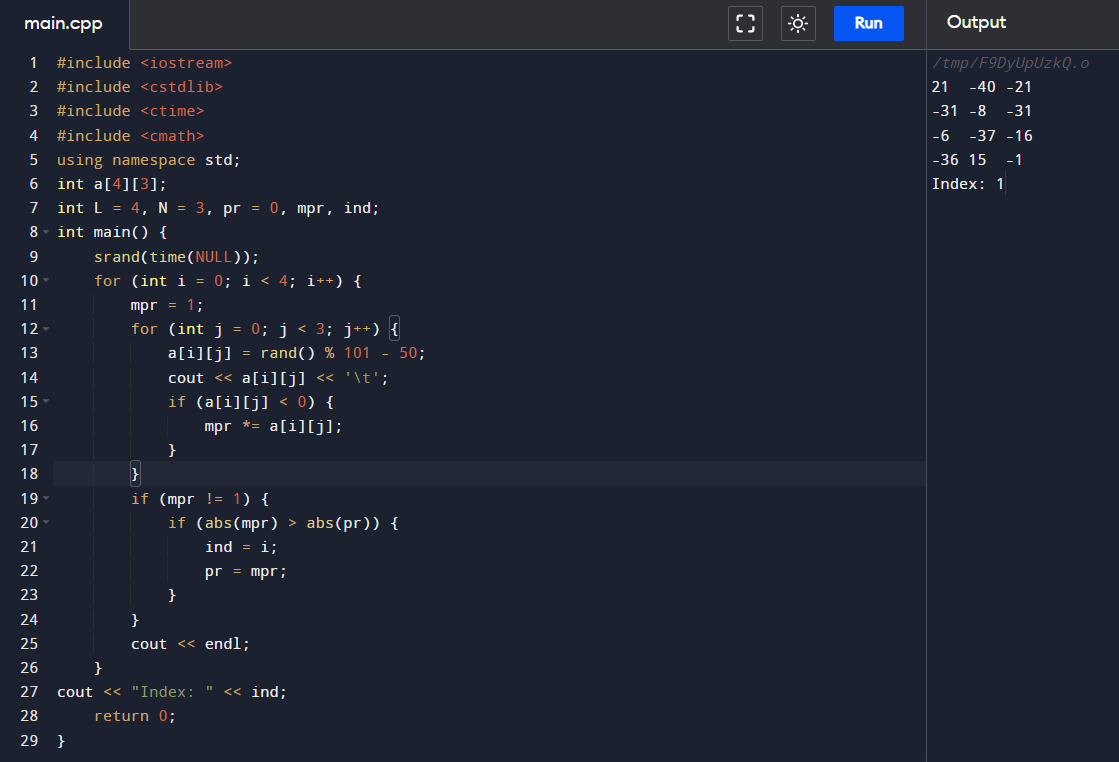


Рисунок 9. Скриншот работы программы.

ВЫВОД К РАБОТЕ

Я научился разрабатывать алгоритмы обработки матриц, программировать их на языках С++ и Python.