Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Основы алгоритмизации и программирования (ОАиП)

ОТЧЕТ

по лабораторной работе №5

Тема работы: Работа с многомерными массивами

Выполнил

студент: гр. 251003 Дедов Н.Ю.

Проверил: Фадеева Е.П.

Минск 2022

Содержание

[1 Постановка задачи 3](#_Toc119950722)

[1.1 Условие задачи 3](#_Toc119950723)

[1.2 Ввод данных 3](#_Toc119950724)

[1.3 Вывод данных 3](#_Toc119950725)

[2 Структура Данных 4](#_Toc119950726)

[3 Текстовый алгоритм данных 5](#_Toc119950727)

[4 Методика решения 6](#_Toc119950728)

[4.1 Условия для ввода данных 6](#_Toc119950729)

[4.2 Проверка введенных данных 6](#_Toc119950730)

[4.3 Алгоритм решения 6](#_Toc119950731)

[5 Схема алгоритма решения задачи по ГОСТ 19.701-90 7](#_Toc119950732)

[6 Результаты расчетов 8](#_Toc119950733)

[Приложение А 9](#_Toc119950734)

[Приложение Б 11](#_Toc119950735)

# Постановка задачи

## Условие задачи

Дана матрица X размерности N\*M. Из матрицы X удалить все строки и столбцы, содержащие хотя бы один нулевой элемент. Матрицу уплотнить, сдвинув оставшиеся элементы к левому верхнему углу.

## Ввод данных

Пользователь вводит число строк и столбцов матрицы X. Нам необходимо удалить все строки и столбцы, содержащие нулевой элемент.

## Вывод данных

Если все условия ввода соблюдены, то вывод должен выглядеть следующим образом:

Преобразованная матрица:

…

Если нулевых элементов нет, то вывод должен выглядеть следующим образом:

В матрице нет нолей.

Если в результате преобразований не осталось ни единой строки или столбца выводится:

Все строки и столбцы удалены.

# Структура Данных

Таблица 1 – Данные

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| M | Integer | Число столбцов, вводится пользователем |
| N | Integer | Число строк, вводится пользователем |
| Mat | Array [1..nummax,  1..nummax] of real | Массив, содержащий матрицу |
| NumMax | Integer | Максимальное допустимое число строк и столбцов |
| I | Integer | Номер строки |
| J | Integer | Номер столбца |
| Zero | Boolean | Проверка есть ли хоть один столбец или строка с 0 |
| StrZero | Boolean | Проверка есть ли в данной строке 0 |
| StrNum | Integer | Временный номер строки |
| StolbNum | Integer | Временный номер столбца |

# Текстовый алгоритм данных

Таблица 2 – Алгоритм решения

|  |  |
| --- | --- |
| Номер  Шага | Назначение шага |
|  | NumMax:=20 |
|  | Ввод Mat[I,J] |
|  | I:=1; |
|  | Zero:=False; |
|  | Начало цикла А1. Проверка выполнение условия: ( I <= N). Если условие истинно, то идти к шагу 6, иначе – к шагу 32 |
|  | StrZero:=False; |
|  | Начало цикла А2. Проверка выполнение условия: (J:= 1 <= M). Если условие истинно, то идти к шагу 8, иначе – к шагу 20 |
|  | Проверка выполнение условия: (Mat[I,J] = 0). Если условие истинно, то идти к шагу 9, иначе – к шагу 20 |
|  | StrZero:=True; |
|  | Начало цикла А3. Проверка выполнение условия: (StrNum:= 1 <= N). Если условие истинно, то идти к шагу 11, иначе – к шагу 17 |
|  | Начало цикла А4. Проверка выполнение условия: (StolbNum:= J <= M – 1). Если условие истинно, то идти к шагу 12, иначе – к шагу 15 |
|  | Mat[StrNum, StolbNum] := Mat[StrNum, StolbNum+1] |
|  | StolbNum:=StolbNum +1 |
|  | Конец цикла А4. Вернуться к шагу 11 |
|  | StrNum:=StrNum +1 |
|  | Конец цикла А3. Вернуться к шагу 10 |
|  | M:=M-1 |
|  | J:=J+1 |
|  | Конец цикла А2. Вернуться к шагу 7 |
|  | Проверка выполнения условия: (StrZero = True). Если условие истинно, то идти к шагу 21, иначе – к шагу 30 |
|  | Zero:=True |
|  | Начало цикла А5. Проверка выполнение условия: (StolbNum := 1 <= M). Если условие истинно, то идти к шагу 23, иначе – к шагу 29 |
|  | Начало цикла А6. Проверка выполнение условия: (StrNum:= I <= N – 1). Если условие истинно, то идти к шагу 24, иначе – к шагу 27 |
|  | Mat[StrNum, StolbNum] := Mat[StrNum+1, StolbNum] |
|  | StrNum:=StrNum +1 |
|  | Конец цикла А6. Вернуться к шагу 23 |
|  | StolbNum:=StolbNum + 1 |
|  | Конец цикла А5. Вернуться к шагу 22 |
|  | N:=N-1. Идти к шагу 31 |
|  | I:=I+1 |
|  | Конец цикла А1. Вернуться к шагу 5 |
|  | Проверка выполнение условия: (Zero = False). Если условие истинно, то идти к шагу 33, иначе – к шагу 34 |
|  | Вывод сообщения, что нулевые элементы отсутствуют. Идти к шагу 37 |
|  | Проверка выполнение условия: (N = 0). Если условие истинно, то идти к шагу 35, иначе – к шагу 36 |
|  | Вывод сообщения о том, что все строки и столбцы были удалены. Идти к шагу 37 |
|  | Вывод Mat [I , J] |
|  | Останов. |

# Методика решения

## Условия для ввода данных

1. M должно быть числом.
2. Число M должно быть действительным.
3. Число строк и столбцов не должно быть больше 20.

## Проверка введенных данных

Так как пользователь может ввести неверные данные, то программа должна это отслеживать и оповестить пользователя об этом. Мы будем использовать проверку соблюдения правил ввода ([см. условия для ввода данных](#_Условия_для_ввода)). В программе это выглядит так:

Repeat

Write('Количество строк до ',NumMax,' n=');

Readln(n);

Until n in [1..NumMax];

Repeat

Write('Количество столбцов до ',NumMax,' m=');

Readln(m);

Until m in [1..NumMax];

## Алгоритм решения

Чтобы найти все нулевые элементы мы будем идти по строкам с первой до последней, и по столбцам с первого до последнего. Если мы встречаем нулевой элемент, то сначала мы удалим весь столбец, а затем и строку со сдвигом в левый верхний угол. Если мы не встретим ни одного нуля, то мы сообщим пользователю. Также если в результате удалений ни одной строки или столбца не останется, мы также сообщим пользователю.

# Схема алгоритма решения задачи по ГОСТ 19.701-90



Рисунок 1 – Схема алгоритма решения задачи по ГОСТ 19.701-90



Рисунок 2 – Схема алгоритма решения задачи по ГОСТ 19.701-90



Рисунок 3 – Схема алгоритма решения задачи по ГОСТ 19.701-90

# Результаты расчетов

Как результат выполнения программы, на экран выводятся следующие результаты расчетов:

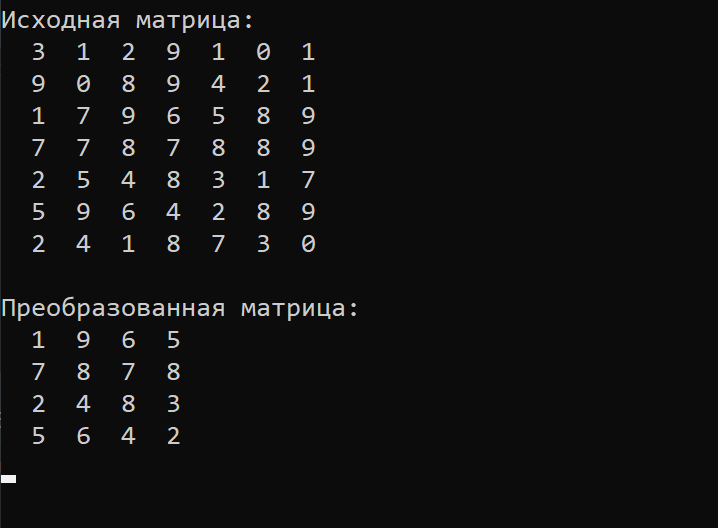


Рисунок 4 – Результаты расчетов

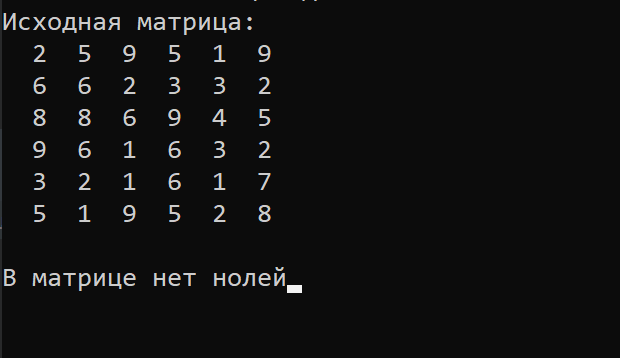


Рисунок 5 – Результаты расчетов при отсутствии нулей

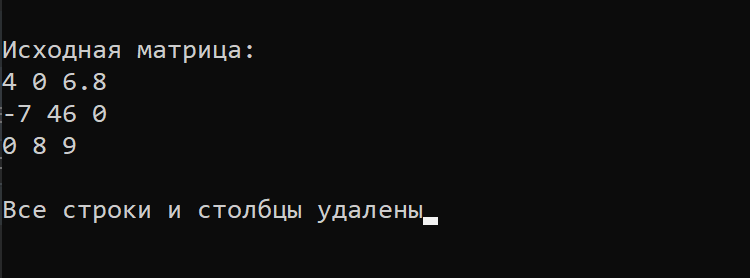


Рисунок 6 – Результаты расчетов при наличии нулей во всех строках и столбцах

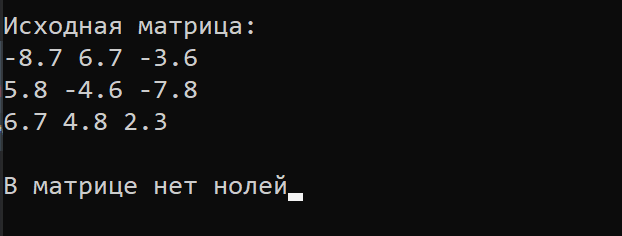


Рисунок 7 – Результаты расчетов при наличии дробных и отрицательных значений

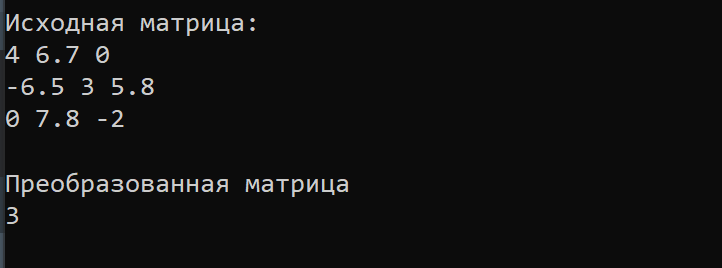


Рисунок 8 – Результаты расчетов при наличии дробных и отрицательных значений

Приложение А

(обязательное)

Исходный код программы

program Lab5f1;

{$APPTYPE CONSOLE}

{$R \*.res}

uses

System.SysUtils;

const NumMax=20;

var

Mat: array [1..NumMax,1..NumMax] of real;

N,M, StrNum, StolbNum, I, J: integer;

Zero, StrZero : Boolean;

//Mat – array where we have our matrix

//N – Number of strings

//M – Number of columns

//StrNum – Temporary variable for strings

//StolbNum – Temporary variable for columns

//I – String variable now

//J – Column variable now

Begin

writeln('Исходная матрица:');

repeat

write('Количество строк до ',NumMax,' n=');

readln(n);

until n in [1..NumMax];

repeat

write('Количество столбцов до ',NumMax,' m=');

readln(m);

until m in [1..NumMax];

writeln('Исходная матрица:');

for i:=1 to n do

begin

for j:=1 to m do

begin

writeln('Mat[',i,' , ',j,']= ');

readln(Mat[i,j]);

end;

writeln;

end;

writeln;

writeln('Исходная матрица:');

for i:=1 to n do

begin

for j:=1 to m do

begin

if Frac(Mat[i,j])=0 then

write(Trunc(Mat[i,j]),' ')

else

write(Mat[i,j]:3:1,' ');

end;

writeln;

end;

writeln;

// Setting that we haven’t delete any strings yet

i:=1;

Zero:=False;

// Setting our cycle for deleting stings and columns

while I <= N do

begin

//Setting that there is no 0 on that string

StrZero:=False;

for j:=1 to m do

if Mat[i,j]=0 then

begin

//Deleting a column from what we are now

StrZero:=True;

for StrNum:=1 to n do

for StolbNum:=j to m-1 do

Mat[StrNum,StolbNum]:=Mat[StrNum,StolbNum+1];

m:=m-1;

end;

// If there was a 0 on a string we get rid of string

if StrZero=True then

begin

Zero:=True;

for StolbNum:=1 to m do

for StrNum:=i to n-1 do

Mat[StrNum, StolbNum]:=Mat[StrNum +1, StolbNum];

n:=n-1;

end

else

i:=i+1;

end;

// If there was no 0 informing a user

if Zero=False then

write('В матрице нет нолей')

else

if (n=0) then

write('Все строки и столбцы удалены')

else

begin

writeln('Преобразованная матрица');

for i:=1 to n do

begin

for j:=1 to m do

begin

if Frac(Mat[i,j])=0 then

write(Trunc(a[i,j]),' ')

else

write(Mat[i,j]:3:1,' ');

end;

writeln;

end;

writeln;

end;

readln

End.