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 №4

Тема работы: Работа с двумерными массивами

Выполнил

студент: гр. 151003 Матошко И.В

Проверил: Фадеева Е.П.

Минск 2021

содержание

[1 Постановка задачи 3](#_Toc88497718)

[2 Текстовый алгоритм решения задачи 4](#_Toc88497719)

[3 Структура данных 5](#_Toc88497720)

[4 Схема алгоритма решения задачи по ГОСТ 19.701-90 6](#_Toc88497721)

[5 Результаты расчетов 8](#_Toc88497722)

[ПРИЛОЖЕНИЕ А 9](#_Toc88497723)

[ПРИЛОЖЕНИЕ Б 12](#_Toc88497724)

[ПРИЛОЖЕНИЕ В 14](#_Toc88497725)

[ПРИЛОЖЕНИЕ Г 16](#_Toc88497726)

[ПРИЛОЖЕНИЕ Д 18](#_Toc88497727)

# Постановка задачи

## Первоначальная постановка

Дан целочисленный массив  размерности . Каждая строка массива упорядочена по неубыванию. Найти числа, встречающиеся во всех строках.

## Окончательная постановка

Дан целочисленный массив  размерности . Каждая строка массива упорядочена по неубыванию. Найти числа, встречающиеся во всех строках.

Вывести на печать: числа, встречающиеся во всех строках.

# Текстовый алгоритм решения задачи

Таблица 1 ­­– Алгоритм решения

|  |  |
| --- | --- |
| Номер  шага | Назначение шага |
|  | N:=4 |
|  | Ввод A[1..N,1..N] |
|  | Вывод A[1..N,1..N] |
|  | m:=1 |
|  | Начало цикла А1. Проверка выполнения условия (m<=N). Если условие истинно, идти к шагу 6, иначе – к шагу 25 |
|  | check: = true |
|  | i:=2 |
|  | Начало цикла А2. Проверка выполнения условия (i<=N). Если условие истинно, идти к шагу 9, иначе – к шагу 21 |
|  | Проверка выполнения условия (check = true). Если условие истинно, идти к шагу 10, иначе – к шагу 19 |
|  | k:=0 |
|  | j:=1 |
|  | Начало цикла А3. Проверка выполнения условия  (A[1, m] >= A[i, j]) and (j <= N). Если условие истинно, идти к шагу 13, иначе – к шагу 17 |
|  | Проверка выполнения условия (A[1, m] = r[i, j]). Если условие истинно, идти к шагу 14, иначе – к шагу 15 |
|  | k := k + 1 |
|  | j := j + 1 |
|  | Конец цикла А3. Идти к шагу 12 |
|  | Проверка выполнения условия (k = 0). Если условие истинно, идти к шагу 18, иначе – к шагу 19 |
|  | check := false |
|  | i:=i+1 |
|  | Конец цикла А2. Идти к шагу 8 |
|  | Проверка выполнения условия (check = true). Если условие истинно, идти к шагу 22, иначе – к шагу 23 |
|  | Вывод A[1,m] |
|  | m:=m+1 |
|  | Конец цикла А1. Идти к шагу 5 |
|  | Останов. |

# Структура данных

Таблица 2 – Данные

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| i | Integer | Счетчик цикла/  Индекс массива |
| j | Integer | Счетчик цикла/  Индекс массива |
| m | Integer | Счетчик цикла/  Индекс массива |
| k | Integer | Количество повторений элемента в строках |
| A | Integer | Вводимый массив |
| check | Boolean | Переменная для проверки условия |
| N | Integer | Константа размера массива |

# 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

# Результаты расчетов

Вследствие выполнения программы на экран выводятся следующие результаты:

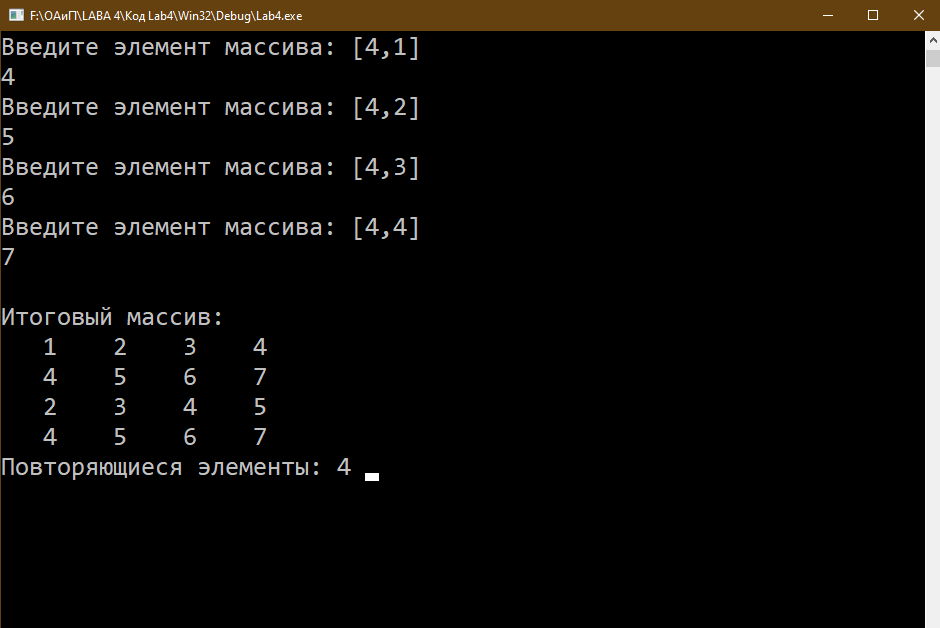


Рисунок 3 – Результаты расчётов

ПРИЛОЖЕНИЕ А

(обязательное)

Исходный код программы

Реализация 1

«Ввод массива с клавиатуры»

Program Lab4;

{The ptogram output the numbers that

are in each row of a two-dimensional array }

{$APPTYPE CONSOLE} // Console application

// Modules declaration

Uses

System.SysUtils;

// Cost declaration

Const

N = 4;

// Variables declaration

Var

i, j, m, k,error: integer;

A: array [1 .. N, 1 .. N] of integer;

check: boolean;

vvod:string;

{ i,j,m - Array indexes, cycle counters;

k - check for coincidence in a number of numbers;

Arr - Initial array;

check - check for coincidence in a number of numbers. }

Begin

// Filling the array A by lines

for i := 1 to N do

begin

repeat

Writeln('Введите элемент массива: ', '[', i, ',', 1, ']');

Readln(vvod);

// Search for wrong input

val(vvod,A[i, 1],error);

if error>0 then

begin

Writeln('Вы ввели неверный член массива');

end;

until error=0;

// Filling the array A by column

for j := 2 to N do

begin

repeat

repeat

Writeln('Введите элемент массива: ', '[', i, ',', j, ']');

Readln(vvod);

// Search for wrong input

val(vvod,A[i, j],error);

if error>0 then

begin

Writeln('Вы ввели неверный член массива');

end;

until error=0;

// Checking for array sorting

if A[i, j] <= A[i, j - 1] then

writeln('Строки в массиве отсортированы по возрастанию, введите число больше предыдущего');

until A[i, j] >= A[i, j - 1];

end;

end;

// Printing array A

writeln;

writeln('Итоговый массив: ');

for i := 1 to N do

begin

for j := 1 to N do

begin

write(A[i, j]:4, ' ');

end;

Writeln;

end;

// Searching for coincidence in a number of numbers

write('Повторяющиеся элементы: ');

for m := 1 to N do

begin

check := true;

for i := 2 to N do

begin

if check = true then

begin

k := 0;

j := 1;

// comparing the elements of line 1 with the rest of

//the lines

while (A[1, m] >= A[i, j]) and (j <= N) do

begin

if A[1, m] = A[i, j] then

k := k + 1;

j := j + 1;

end;

// Checking for matches in a row

if k = 0 then

check := false;

end;

end;

// Output if there are all matches

if check = true then

write(A[1,m],' ');

end;

Readln;

End.

ПРИЛОЖЕНИЕ Б

(обязательное)

Тестовые наборы

Тестовый набор 1

«Проверка ввода»

Тест 1

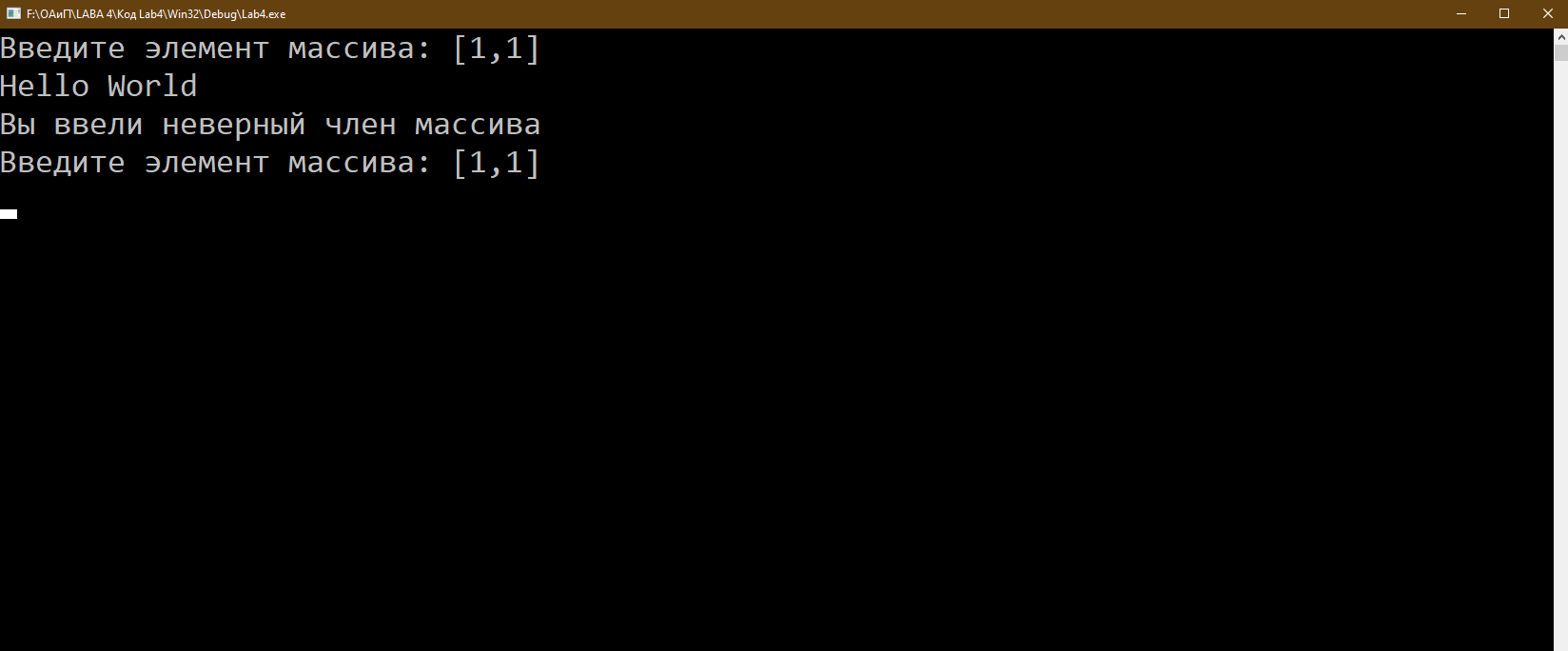
Тестовая ситуация: проверка ввода букв в поле для чисел

Исходные данные: A[i, 1] = «Hello World»

Ожидаемый результат:

Вы ввели неверный член массива

Полученный результат:



Тест 2

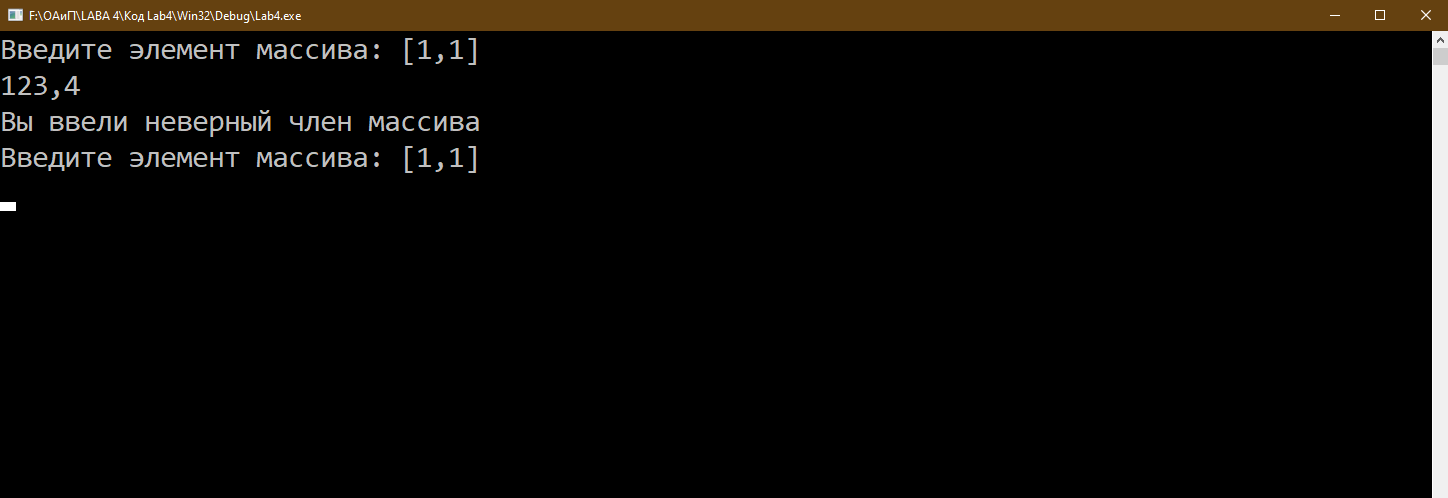
Тестовая ситуация: проверка ввода нецелочисленных значений

Исходные данные: A[i, 1] = «123,4»

Ожидаемый результат:

Вы ввели неверный член массива

Полученный результат:



Тестовый набор 2

«Проверка решений»

Тест 1

Тестовая ситуация: проверка решений

Исходные данные: A[1,1] = «1», A[1,2] = «2», A[1,3] = «3», A[1,4] = «4», A[2,1] = «2», A[2,2] = «3», A[2,3] = «4», A[2,4] = «5», A[3,1] = «2», A[3,2] = «3», A[3,3] = «4», A[3,4] = «5», A[4,1] = «4», A[4,2] = «5», A[4,3] = «6», A[4,4] = «7»,

Ожидаемый результат:

Повторяющиеся элементы: 4

Полученный результат:



ПРИЛОЖЕНИЕ В

(обязательное)

Дополнительный код программы

Реализация 2

«Ввод массива с помощью генератора чисел»

Program Lab4R2;

{The ptogram output the numbers that

are in each row of a two-dimensional array }

{$APPTYPE CONSOLE} // Console application

// Modules declaration

Uses

System.SysUtils;

// Cost declaration

Const

N = 4;

// Variables declaration

Var

i, j, m, k: integer;

Arr: array [1 .. N, 1 .. N] of integer;

check: boolean;

{ i,j,m - Array indexes, cycle counters;

k - check for coincidence in a number of numbers;

Arr - Initial array;

check - check for coincidence in a number of numbers. }

Begin

// Filling the array Arr

for i := 1 to N do

begin

Arr[i,1]:=Random(100)-50;

for j := 2 to N do

begin

repeat

Arr[i,j]:=Random(100)-50;

until Arr[i, j] >= Arr[i, j - 1];

end;

end;

// Printing array Arr

writeln;

writeln('Итоговый массив: ');

for i := 1 to N do

begin

for j := 1 to N do

begin

write(Arr[i, j]:4, ' ');

end;

Writeln;

end;

// Searching for coincidence in a number of numbers

write('Повторяющиеся элементы: ');

for m := 1 to N do

begin

check := true;

for i := 2 to N do

begin

if check = true then

begin

k := 0;

j := 1;

while (Arr[1, m] >= Arr[i, j]) and (j <= N) do

begin

if Arr[1, m] = Arr[i, j] then

k := k + 1;

j := j + 1;

end;

// Checking for matches in a row

if k = 0 then

check := false;

end;

end;

if check = true then

write(Arr[1,m],' ');

end;

Readln;

End.

ПРИЛОЖЕНИЕ Г

(обязательное)

Дополнительный код программы

Реализация 3

«Ввод массива с помощью типизированной константы-массив»

Program Lab4R3;

{The ptogram output the numbers that

are in each row of a two-dimensional array }

{$APPTYPE CONSOLE} // Console application

// Modules declaration

Uses

System.SysUtils;

// Cost declaration

Const

N = 3;

Arr: array [1 .. N, 1 .. N] of integer = ((3,4,5),(4,6,43),(2,4,77));

// Variables declaration

Var

i, j, m, k: integer;

check: boolean;

{ i,j,m - Array indexes, cycle counters;

k - check for coincidence in a number of numbers;

Arr - Initial array;

check - check for coincidence in a number of numbers. }

Begin

// Printing array Arr

writeln;

writeln('Итоговый массив: ');

for i := 1 to N do

begin

for j := 1 to N do

begin

write(Arr[i, j]:4, ' ');

end;

Writeln;

end;

// Searching for coincidence in a number of numbers

write('Повторяющиеся элементы: ');

for m := 1 to N do

begin

check := true;

for i := 2 to N do

begin

if check = true then

begin

k := 0;

j := 1;

while (Arr[1, m] >= Arr[i, j]) and (j <= N) do

begin

if Arr[1, m] = Arr[i, j] then

k := k + 1;

j := j + 1;

end;

// Checking for matches in a row

if k = 0 then

check := false;

end;

end;

if check = true then

write(Arr[1,m],' ');

end;

Readln;

End.

ПРИЛОЖЕНИЕ Д

(обязательное)

Дополнительный код программы

Реализация 4

«Ввод массива с помощью типизированной переменной-массив»

Program Lab4R4;

{The ptogram output the numbers that

are in each row of a two-dimensional array }

{$APPTYPE CONSOLE} // Console application

// Modules declaration

Uses

System.SysUtils;

// Cost declaration

Const

N = 3;

// Variables declaration

Var

i, j, m, k: integer;

check: boolean;

Arr: array [1 .. N, 1 .. N] of integer = ((3,4,5),(3,6,43),(3,4,77));

{ i,j,m - Array indexes, cycle counters;

k - check for coincidence in a number of numbers;

Arr - Initial array;

check - check for coincidence in a number of numbers. }

Begin

// Printing array Arr

writeln;

writeln('Итоговый массив: ');

for i := 1 to N do

begin

for j := 1 to N do

begin

write(Arr[i, j]:4, ' ');

end;

Writeln;

end;

// Searching for coincidence in a number of numbers

write('Повторяющиеся элементы: ');

for m := 1 to N do

begin

check := true;

for i := 2 to N do

begin

if check = true then

begin

k := 0;

j := 1;

while (Arr[1, m] >= Arr[i, j]) and (j <= N) do

begin

if Arr[1, m] = Arr[i, j] then

k := k + 1;

j := j + 1;

end;

// Checking for matches in a row

if k = 0 then

check := false;

end;

end;

if check = true then

write(Arr[1,m],' ');

end;

Readln;

End.