РУСЕНСКИ УНИВЕРСИТЕТ КАТЕДРА

„АНГЕЛ КЪНЧЕВ“ „ИИТ“

**КУРСОВА ЗАДАЧА 1**

по дисциплината „Програмиране“

на Диан Лъчезаров Лазаров

студент I курс, специалност КСТ, група 26А, фак. № 233078

**Русе** Разработил: ...................................

**20.11.2023 г.** **/**Д. Лазаров**/**

Проверил: ...................................

**/**гл. ас. д-р инж. С. Антонов**/**

**ЗАДАНИЕ**

Даден е двумерен масив **A** с **m** реда и **n** стълба. Да се състави блок-схема на алгоритъм, чрез който се създава нов масив ***b1***, ***b2***,..., ***bn***,като стойността на ***bj*** е равна на индекса на най-голямата отрицателна стойност в ***j***-ия стълб. Да се състави програма по този алгоритъм.

**ИЗИСКВАНИЯ КЪМ ПРОГРАМАТА**

Програмата се състои от два масива – двумерен (за въвеждане от клавиатурата на броя редове, броя стълбове и съответно елементите на масива) и едномерен (за извеждане на индекса на най-голямата отрицателна стойност във всеки стълб).

Предполага се, че ще се въвеждат само цели числа и затова всички декларирани променливи и масиви са от тип **integer (int)**. За обработка на данните са използвани условен оператор **if** и цикъл **for**.

Преди извеждане на резултата програмата проверява дали в съответния стълб са въведени отрицателни стойности. Ако това не е така, се извежда подходящо съобщение.

**ПРОЕКТИРАНЕ НА ПРОГРАМАТА**

Край

За

j

от

0

до

n

Bj:=max\_otr\_index

max\_otr\_index:=i

Начало

max\_otr:=-1

max\_otr\_index:=-1

Въведи

n, B[n]

(Aij<0)

(Aij>max\_otr) или

(max\_otr\_index:==-1)

‘Няма отрицателни стойности в стълб j’

Изведи

B[j]

Да

Да

Не

Bj != -1

За

j

от

0

до

n

За

i

от

0

до

m

max\_otr:=Aij

Не

Въведи

m, n, A[m, n]

**ТЕКСТ НА ПРОГРАМАТА**

#include <stdio.h>

#include <stdlib.h>

void main() {

system("chcp 1251");

int A[10][20], B[10];

int m, n, i, j;

printf\_s("\nВъведете брой редове: "); scanf\_s("%d", &m);

printf\_s("Въведете брой стълбове: "); scanf\_s("%d", &n);

printf\_s("\nВъведете елементите на двумерния масив A[%d][%d]:\n", m, n);

for (i = 0; i < m; i++) {

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

scanf\_s("%d", &A[i][j]);

}

}

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

int max\_otr = -1; //най-голяма отрицателна стойност

int max\_otr\_index = -1; //индекс на най-голямата отрицателна стойност

for (i = 0; i < m; i++) {

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

if (A[i][j] > max\_otr || max\_otr\_index == -1) {

max\_otr = A[i][j];

max\_otr\_index = i;

}

}

}

B[j] = max\_otr\_index;

}

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

if (B[j] != -1) { //проверка за отрицателни стойности

printf\_s("Индексът на най-голямата отрицателна стойност в стълб %d е: %d\n", j, B[j]);

}

else {

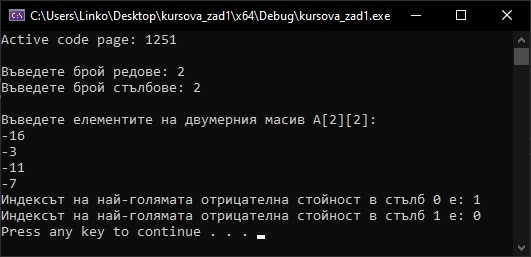
printf\_s("Няма отрицателни стойности в стълб %d\n", j);

}

}

system("pause");

}

**ТЕСТВАНЕ**