

**UNIVERSITATEA DE STAT DIN MOLDOVA**

**FACULTATEA MATEMATICĂ ȘI INFORMATICĂ**  
**DEPARTAMENTUL INFORMATICĂ**

**CEMÎRTAN CRISTIAN**

**Lucrarea individuală nr. 10**  
la disciplina *Arhitectura Calculatoarelor și Limbaje de Asamblare*

Coordonator: Sturza Greta, lector universitar

**Chișinău, 2021**

## Cuprins

<b>Sarcină.....</b>	<b>3</b>
<b>Cod sursă.....</b>	<b>3</b>
<b>Rezultat.....</b>	<b>6</b>
<b>Concluzie.....</b>	<b>7</b>

## Sarcină

Fie A o matrice numerică de dimensiunea  $m \times n$ . Să se găsească elementele minimale ale fiecărei linii și să se amplaseze în prima coloană, iar elementele primei coloane – în locul elementelor minimale.

## Cod sursă

```
COMMENT *
    Lucrare individuala nr. 10, varianta complexa nr. 3
    Copyright Cemirtan Cristian 2021
    Grupa I 2101
*

INCLUDE stdlibc.inc

.MODEL small
.STACK

crlf EQU 0Dh, 0Ah

.DATA
txt1 DB 'Introduceti m si n:', 0
txt2 DB crlf, 'Rezultat:', 0
fmt1 DB '%d%d', 0
fmt2 DB crlf, 'Introduceti %d elemente:', crlf, 0

m DW 0
n DW 0

.CODE
; initializare
    .STARTUP

; afiseaza text #1
    lea si, txt1
    call puts

; citeste m si n
    push OFFSET n
    push OFFSET m
    lea si, fmt1
    call scanf
    add sp, 4

; testare daca e pozitiv si mai mare decat 0
    mov ax, m
    mov bx, n

    test ax, ax
    jle iesire_err
```

```

; testare
    test bx, bx
    jg inceput

iesire_err:
    .EXIT 1

inceput:
; rezervez spatiu pentru matrice
    mul bx

; iesire in caz de revarsare
    jo iesire_err

; afisarea
    push ax
    lea si, fmt2
    call printf
    pop ax

; cadru nou
    push bp
    mov bp, sp

    add ax, ax ; n*m cuvinte
    add ax, 2 ; cuvant temporar ce contine valoarea initiala a reg. di
    sub sp, ax ; rezervez spatiu

    mov [fmt1 + 2], 0 ; fmt1 <- '%d', 0

; algoritm
    mov di, sp

; se initializeaza spatiul rezervat cu 0
    xor dx, dx
    call memset

    mov cx, m

al_buc1a_1:
; se stocheaza numarul citit in ss:[di]
    push di
    lea si, fmt1
    call scanf
    add sp, 2

    mov [bp - 2], di ; inceput
    mov bx, di ; pozitia minimului

    mov dx, n
    add di, 2
    dec dx
    jz al_buc1a_1_dec

```

```

    al_bucla_2:
        push di
        call scanf
        add sp, 2

; se compare elementul curent cu cel minim
        mov ax, ss:[di]
        cmp ss:[bx], ax
        jle al_bucla_2_dec

; daca ss:[bx] > ss:[di]
        mov bx, di

    al_bucla_2_dec:
        add di, 2
        dec dx
        jnz al_bucla_2

; se schimba cu pozitiile
        mov si, [bp - 2]
        mov dx, ss:[si]
        mov ax, ss:[bx]

        mov ss:[bx], dx
        mov ss:[si], ax

    al_bucla_1_dec:
        dec cx
        jnz al_bucla_1

; afisare
        lea si, txt2
        call puts

        lea si, fmt1
        mov WORD PTR [si + 2], ' ' ; fmt1 <- '%d ', 0

        mov di, sp
        mov cx, m

    af_bucla_1:
        mov dx, n

        af_bucla_2:
            push ss:[di]
            call printf
            add sp, 2

            add di, 2
            dec dx
            jnz af_bucla_2

; linie noua
        putnl

```

```

        dec cx
        jnz af_bucla_1

; sfarsit cadru pentru algoritm
        mov sp, bp
        pop bp

; iesire cu succes
        .EXIT 0
END

```

## Rezultat

```

E:\>tasm i10
Turbo Assembler Version 3.0 Copyright (c) 1988, 1991 Borland International

Assembling file:   i10.ASM
Error messages:    None
Warning messages:  None
Passes:            1
Remaining memory:  448k

E:\>tlink i10 stdlibc
Turbo Link Version 2.0 Copyright (c) 1987, 1988 Borland International

E:\>i10
Introduceti m si n:
5 4

Introduceti 20 elemente:
6 7 0 -3
-9 6 -10 1
5 4 9 5
1 2 3 4
6 5 7 6

Rezultat:
-3 7 0 6
-10 6 -9 1
4 5 9 5
1 2 3 4
5 6 7 6

E:\>_

```

Figura 1. Rezultatul afișat la ecran.

## Concluzie

Pe parcursul realizării a programului, am utilizat operații asupra stivei pentru a crea dinamic, tablouri bidimensionale de dimensiuni diferite. În privința implementării, matricea este reprezentată ca un vector congruent, cu lungimea care este produsul între mărimile  $m$  și  $n$ .

Pentru a asigura stabilitatea programului, inițial se reduc la zero elementele a unei matrice, pentru a nu conține valori fictive, în cazul introducerii de la tastatură, nu șir de caractere ce nu reprezintă un număr.

De asemenea, înainte de a rezerva spațiu pentru un matrice, se verifică dacă valorile  $m$  și  $n$  sunt numere naturale pozitive, și dacă produsul lor nu depășește un cuvânt. În cazul nesatisfacerii a acestor condiții, programul se termină.