

**Міністерство освіти та науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет прикладної математики
Кафедра системного програмування і спеціалізованих комп’ютерних
систем**

**ЛАБОРАТОРНА РОБОТА №4
з дисципліни
“Програмування”
Тема: «Блочна структура програми. Функції»**

**Виконав: Фесенко Д.О.
Студент групи КВ-34
Варіант №23**

Київ 2023

Постановка задачі

Написати програму розв'язку завдання за варіантом з обов'язковим використанням функцій для опису дій, які повинні повторюватися, а саме: - введення елементів масиву;

- виведення елементів масиву;

- виконання над масивом дій, заданих за варіантом.

Завдання за варіантом №23

23. Дано дві матриці цілих чисел $A[4,4]$, $B[6,5]$. Переписати кожену матрицю в одномірний масив по рядках.

Код програми

```
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#define nmA 4
#define nB 6
#define mB 5
#define mult nmA * nmA + nB * mB

void scanM(const int* M, int row, int col);
void printM(int* M, int row, int col);
int copy(int* M1, int* M2, int index);

int main() {
    int A[nmA][nmA], B[nB][mB];

    printf("Fill first matrix A: \n");
    scanM(&A[0][0], nmA, nmA);
    printf("Fill second matrix B: \n");
    scanM(&B[0][0], nB, mB);

    printf("Matrix A: \n");
    printM(&A[0][0], nmA, nmA);
    printf("Matrix A: \n");
    printM(&B[0][0], nB, mB);

    printf("Array from matrix A & B: \n");
    for (int i = 0; i < mult; i++)
    {
        printf("[%6d] ", copy(&A[0][0], &B[0][0], i));
    }

    return 0;
}

void scanM(int* M, int row, int col) {
    for (int i = 0; i < row; i++) {
        for (int j = 0; j < col; j++) {
            scanf_s("%d", &M[i * col + j]);
        }
    }
}

void printM(int* M, int row, int col) {
    for (int i = 0; i < row; i++) {
        for (int j = 0; j < col; j++) {
            printf("[%6d] ", M[i * col + j]);
        }
        printf("\n");
    }
}

int copy(int* M1, int* M2, int index) {
    int temp[mult], b=0;

    for (int i = 0; i < 1; i++) {
        for (int j = 0; j < nmA * nmA; j++, b++) {
            temp[b] = M1[i * nmA * nmA + j];
        }
        for (int j = 0; j < nB * mB; j++, b++) {
```

```

        temp[b] = M2[i * nB * mB + j];
    }
    return temp[index];
}

```

Тестування

```

Fill first matrix A:
1
2
3
4
5
6
7
8
9
10
11
12
13
14
1
16
Fill second matrix B:
30
29
28
27
26
25
24
23
22
21
20
19
18
17
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1

```

```

Matrix A:
[ 1] [ 2] [ 3] [ 4]
[ 5] [ 6] [ 7] [ 8]
[ 9] [10] [11] [12]
[13] [14] [ 1] [16]
Matrix B:
[30] [29] [28] [27] [26]
[25] [24] [23] [22] [21]
[20] [19] [18] [17] [16]
[15] [14] [13] [12] [11]
[10] [ 9] [ 8] [ 7] [ 6]
[ 5] [ 4] [ 3] [ 2] [ 1]
Array from matrix A & B:
[ 1] [ 2] [ 3] [ 4] [ 5] [ 6] [ 7] [ 8] [ 9] [10] [11] [12] [13] [14] [ 1] [16] [30] [29] [28] [27]
[26] [25] [24] [23] [22] [21] [20] [19] [18] [17] [16] [15] [14] [13] [12] [11] [10] [ 9] [ 8] [ 7]
[ 6] [ 5] [ 4] [ 3] [ 2] [ 1]

```