



Универзитет „Св. Кирил и Методиј“ - Скопје  
**ФАКУЛТЕТ ЗА ИНФОРМАТИЧКИ НАУКИ  
И КОМПЈУТЕРСКО ИНЖЕНЕРСТВО**

## Аудиторски вежби 5

### Полиња и матрици

Структурно програмирање

# Содржина

- 1 Вектори (еднодимензионални полиња)
- 2 Матрици (дводимензионални полиња)

# Задача 1

Да се напише програма која за две низи кои се внесуваат од тастатура ќе провери дали се еднакви или не. На екран да се испачати резултатот од споредбата.

Максимална големина на низите е 100.

## Решение 1 дел

```
#include <stdio.h>
#define MAX 100
int main() {
    int n1, n2, element, i;
    int a[MAX], b[MAX];
    printf("First array size: ");
    scanf("%d", &n1);
    printf("Second array size: ");
    scanf("%d", &n2);
    if (n1 != n2)
        printf("Arrays are equal\n");
}
```

# Задача 1

## Решение 2 дел

### Решение 2 дел

```
else {
    printf("Elements of the first array: \n");
    for (i = 0; i < n1; ++i) {
        printf("a[%d] = ", i);
        scanf("%d", &a[i]);
    }
    printf("Elements of the second array: \n");
    for (i = 0; i < n2; ++i) {
        printf("b[%d] = ", i);
        scanf("%d", &b[i]);
    }
    // check if arrays are equal:
    for (i = 0; i < n1; ++i)
        if (a[i] != b[i])
            break;
    if (i == n1)
        printf("Arrays are equal\n");
    else
        printf("Arrays are not equal\n");
}
return 0;
}
```

## Задача 2

Да се напише програма која за низа, чии што елементи се внесуваат од стандарден влез, ќе го пресмета збирот на парните елементи, збирот на непарните елементи, како и односот помеѓу бројот на парни и непарни елементи. Резултатот да се испечати на екран.

### Пример

За низата:

3 2 7 6 2 5 1

На екран ќе се испечати:

`suma_parni = 8`

`suma_neparni = 16`

`odnos = 0.75`

## Задача 2

## Решение

## Решение

```
#include <stdio.h>
#define MAX 100
int main() {
    int i, n, a[MAX], brNep = 0, brPar = 0, sumNep = 0, sumPar = 0;
    scanf("%d", &n);
    for (i = 0; i < n; ++i)
        scanf("%d", &a[i]);
    for (i = 0; i < n; ++i) {
        if (a[i] % 2) {
            brNep++;
            sumNep += a[i];
        } else {
            brPar++;
            sumPar += a[i];
        }
    }
    printf("Sum of odd elements: %d\nSum of even elements: %d\n", sumPar, sumNep);
    printf("The ratio is %.2f\n", (float)brPar / brNep);
    return 0;
}
```

## Задача 3

Да се напише програма која ќе го пресмета скаларниот производ на два вектори со по  $n$  координати. Бројот на координати  $n$ , како и координатите на векторите се внесуваат од стандарден влез. Резултатот да се испечати на екран.

# Задача 3

## Решение

### Решение

```
#include <stdio.h>
#define MAX 100
int main() {
    int a[MAX], b[MAX], n, i, scalar = 0;
    scanf("%d", &n);
    for (i = 0; i < n; ++i)
        scanf("%d", &a[i]);
    for (i = 0; i < n; ++i)
        scanf("%d", &b[i]);
    for (i = 0; i < n; ++i)
        scalar += a[i] * b[i];
    printf("The scalar product is: %d\n", scalar);
    return 0;
}
```



## Задача 4

Да се напише програма која ќе провери дали дадена низа од  $n$  елементи која се внесува од тастатура е строго растечка, строго опаѓачка или ниту строго растечка ниту строго опаѓачка. Резултатот да се испечати на екран.

# Задача 4

## Решение

### Решение

```
#include <stdio.h>
#define MAX 100
int main() {
    int n, element, a[MAX], i;
    short rastecka = 1, opagacka = 1;
    scanf("%d", &n);
    for (i = 0; i < n; ++i)
        scanf("%d", &a[i]);
    for (i = 0; i < n - 1; ++i) {
        if (a[i] >= a[i + 1]) {
            rastecka = 0;
            break;
        }
    }
    for (i = 0; i < n - 1; ++i) {
        if (a[i] <= a[i + 1]) {
            opagacka = 0;
            break;
        }
    }
    if (!opagacka && !rastecka)
        printf("Array is not increasing and not decreasing\n");
    else if (opagacka)
        printf("Array is decreasing\n");
    else if (rastecka)
        printf("Array is increasing\n");
    return 0;
}
```

## Задача 5

Да се напише програма која што ќе ги избрише дупликатите од една низа. На крај, да се испечати на екран новодобиената низа. Елементите од низата се внесуваат од стандарден влез.

## Задача 5

## Решение

## Решение

```
#include <stdio.h>
#define MAX 100
int main() {
    int a[MAX], n, i, j, k, izbrisani = 0;
    scanf("%d", &n);
    for (i = 0; i < n; ++i)
        scanf("%d", &a[i]);
    for (i = 0; i < n - izbrisani; ++i)
        for (j = i + 1; j < n - izbrisani; ++j)
            if (a[i] == a[j]) {
                for (k = j; k < n - 1 - izbrisani; ++k)
                    a[k] = a[k + 1];
                izbrisani++;
            }
    n -= izbrisani;
    for (i = 0; i < n; ++i)
        printf("%d\t", a[i]);
    return 0;
}
```

# Содржина

- 1 Вектори (еднодимензионални полиња)
- 2 Матрици (дводимензионални полиња)

## Задача 6

Да се напише програма која ќе испечати на екран дали дадена матрица е симетрична во однос на главната дијагонала. Димензиите и елементите на матрицата се внесуваат од стандарден влез.

## Задача 6

## Решение

## Решение

```
# include <stdio.h>
# define MAX 100
int main () {
    int a[MAX][MAX], n, i, j, zname = 0;
    printf ("Vnesi dimenzii na matricata: \n");
    scanf ("%d", &n);
    printf ("Vnesi gi elementite na matricata: \n");
    for (i = 0; i < n; ++i)
        for (j = 0; j < n; ++j)
            scanf ("%d", &a[i][j]);
    for (i = 0; i < n - 1; ++i) {
        for (j = i + 1; j < n; ++j)
            if (a[i][j] != a[j][i]) {
                zname = 1;
                break;
            }
        if (zname) break;
    }
    if (!zname)
        printf ("Matricata e simetricna vo odnos na glavnata dijagonala \n");
    else
        printf ("Matricata ne e simetricna vo odnos na glavnata dijagonala \n");
    return 0;
}
```

## Задача 7

Да се напише програма која за матрица внесена од тастатура ќе ги замени елементите од главната дијагонала со разликата од максималниот и минималниот елемент во матрицата. Резултантната матрица да се испечати на екран.



## Задача 7

## Решение

## Решение

```
#include <stdio.h>
#define MAX 100
int main() {
    int a[MAX][MAX], n, i, j, max, min;
    scanf("%d", &n);
    for (i = 0; i < n; ++i)
        for (j = 0; j < n; ++j) {
            scanf("%d", &a[i][j]);
            if (i == 0 && j == 0)
                max = min = a[i][j];
            else if (max < a[i][j])
                max = a[i][j];
            else if (min > a[i][j])
                min = a[i][j];
        }

    for (i = 0; i < n; ++i)
        a[i][i] = max - min;

    for (i = 0; i < n; ++i) {
        printf("\n");
        for (j = 0; j < n; ++j)
            printf("%d\t", a[i][j]);
    }
    return 0;
}
```

## Задача 8

Да се пресмета разликата на збирот на елементите во непарните колони и збирот на елементите во парните редици. Резултатот да се испечати на екран. Податоците за матрицата се внесуваат од тастатура. Матрицата не мора да биде квадратна.

## Задача 8

## Решение

## Решение

```
#include <stdio.h>
#define MAX 100
int main() {
    int a[MAX][MAX], n, m, i, j, sumKol = 0, sumRed = 0;
    scanf("%d %d", &n, &m);
    for (i = 0; i < n; ++i)
        for (j = 0; j < m; ++j)
            scanf("%d", &a[i][j]);

    for (i = 0; i < n; ++i)
        for (j = 0; j < m; ++j) {
            if ((j + 1) % 2)
                sumKol += a[i][j];
            if (!((i + 1) % 2))
                sumRed += a[i][j];
        }
    printf("%d", sumKol - sumRed);
    return 0;
}
```

# Материјали и прашања

Предавања, аудиториски вежби, соопштенија  
[courses.finki.ukim.mk](https://courses.finki.ukim.mk)

Изворен код на сите примери и задачи  
<https://github.com/tdelev/SP/tree/master/latex/src>

Прашања и дискусија  
[forum.finki.ukim.mk](https://forum.finki.ukim.mk)