



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

## Структурно програмирање

Аудиториски вежби 7

Верзија 1.0, 14 Ноември, 2016

# Содржина

1. Матрици .....	1
1.1. Задача 1 .....	1
1.2. Задача 2 .....	1
1.3. Задача 3 .....	2
2. Функции .....	3
2.1. Потсетување од предавања .....	3
2.2. Функции од математичката библиотека <code>math.h</code> .....	4
2.3. Најчесто користени математички функции .....	4
2.4. Задача 1 .....	5
3. Изворен код од примери и задачи .....	6

# 1. Матрици

## 1.1. Задача 1

Да се напише програма која за матрица внесена од тастатура ќе ја пресмета разликата на збирот на елементите на непарните колони и збирот на елементите на парните редици. Матрицата не мора да биде квадратна.

*Решение p7\_1.c*

```
#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;
}
```

## 1.2. Задача 2

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.

```
#include <stdio.h>
#define MAX 100

int main() {
    int a[MAX][MAX];
    int n;
    scanf("%d", &n);
    int i, j;
    for (i = 0; i < n; ++i) {
        for (j = 0; j < n; ++j) {
            scanf("%d", &a[i][j]);
            if (i == 0 && j == 0) {
                min = max = a[i][j];
            } else if (a[i][j] > max) {
                max = a[i][j];
            } else if (a[i][j] < min) {
                min = a[i][j];
            }
        }
    }

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

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

### 1.3. Задача 3

Да се напише програма која за квадратна матрица внесена од тастатура ќе испечати на екран дали таа е симетрична во однос на главната дијагонала.

```
#include <stdio.h>
#define MAX 100
int main () {
    int a[MAX][MAX], n, i, j, simetrichna = 1;
    printf("Vnesete dimenzija na kvadratna matrica: \n");
    scanf("%d", &n);
    printf("Vnesete gi elementite: \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]) {
                simetrichna = 0;
                break;
            }
        if (!simetrichna) break;
    }
    if (simetrichna)
        printf("Matricata e SIMETRICHNA vo odnos na glavnata dijagonala\n");
    else
        printf("Matricata ne e SIMETRICHNA vo odnos na glavnata dijagonala\n");
    return 0;
}
```

## 2. Функции

### 2.1. Потсетување од предавања

#### 2.1.1. Дефиниција на функција во C

```
tip ime(lista_na_formalni_argumenti) {
    telo_na_funkcijata
}
```

- `tip` - типот на вредноста која ја враќа функцијата
- `ime` - името на функцијата
- `lista_na_formalni_argumenti` - листата со формални аргументи ги содржи аргументите заедно со нивните типови, разделени со запирка
- `telo_na_funkcijata` - телото на функцијата ги содржи истите елементи како и самата `main()` функција

#### 2.1.2. Повик на функција

```
ime(lista_na_argumenti);
```

- `ime` - името на веќе дефинираната функција
- `lista_na_argumenti` - листата на аргументи е со вистински аргументи кои што ако се повеќе ако се одделуваат со запирка

## 2.1.3. Пример на кориснички дефинирана функција

Да се напише програма во која со посебна функција се пресметува куб  $n^3$  за вчитан природен број  $n$ .

*Пример ex7\_1.c*

```
#include <stdio.h>

double kub(int x) {
    return x * x * x;
}

int main() {
    int n;
    printf("Vnesete eden prirodan broj: ");
    scanf("%d", &n);
    double rezultat = kub(n);

    printf("Kubot na brojot %d e %.2f\n", n, rezultat);
    return 0;
}
```

## 2.2. Функции од математичката библиотека `math.h`

- Во C постои стандардна математичка библиотека `math.h` која што содржи многу готови математички функции.
- За да се употребува, треба претходно да се вклучи со: `#include <math.h>`
- Сите функции од стандардната библиотека `math.h` примаат аргументи од тип `double` и враќаат вредности од истиот тип.

## 2.3. Најчесто користени математички функции

Функција	Објаснување
<code>sqrt(x)</code>	квадратен корен од $x$
<code>exp(x)</code>	експоненцијална функција $e^x$
<code>log(x)</code>	природен логаритам од $x$ (со основа $e$ )
<code>log10(x)</code>	логаритам од $x$ со основа 10
<code>fabs(x)</code>	апсолутна вредност од $x$
<code>ceil(x)</code>	заокружува $x$ на најмалиот цел број не помал од $x$

Функција	Објаснување
<code>floor(x)</code>	заокружува $x$ на најголемиот цел број не поголем од $x$
<code>pow(x, y)</code>	$x$ на степен $y$
<code>fmod(x, y)</code>	остаток од $x/y$ како реален број
<code>sin(x)</code>	синус од $x$ (во радијани)
<code>cos(x)</code>	косинус од $x$ (во радијани)
<code>tan(x)</code>	тангенс од $x$ (во радијани)

### 2.3.1. Пример на користење на функција од `math.h`

Да се напише програма во која со посебна функција се пресметува куб  $n^3$  за вчитан природен број  $n$ .

*Пример ex7\_2.c*

```
#include <stdio.h>
#include <math.h>

int main() {
    int n;
    printf("Vnesete eden prirodan broj: ");
    scanf("%d", &n);
    double rezultat = pow(n, 3);

    printf("Kubot na brojot %d e %.2f\n", n, rezultat);
    return 0;
}
```

## 2.4. Задача 1

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

*Решение p6\_4.c*

```
Unresolved directive in sp_av7.adoc - include::../../src/av7/p6_4.c[]
```

## 3. Изворен код од примери и задачи

<https://github.com/finki-mk/SP/>

Source code ZIP