

01001111 01110011 01101110 01101111  
01110110 01101001 00100000 01110000  
01110010 01101111 01100111 01110010  
01100001 01101101 01101001 01110010  
01100001 01101110 01101010 01100001



# OSNOVI PROGRAMIRANJA

## VEŽBE 9

Marina Svičević, Đorđe Nedić, Mladen Marić,  
Danica Prodanović, Jovan Janićijević

- **Matrica** dimenzije  $m \times n$  je šema brojeva koja se zapisuje kao:

$$A = \begin{bmatrix} a_{00} & a_{01} & \dots & a_{0n-1} \\ a_{10} & a_{11} & \dots & a_{1n-1} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m-10} & a_{m-11} & \dots & a_{m-1n-1} \end{bmatrix}$$

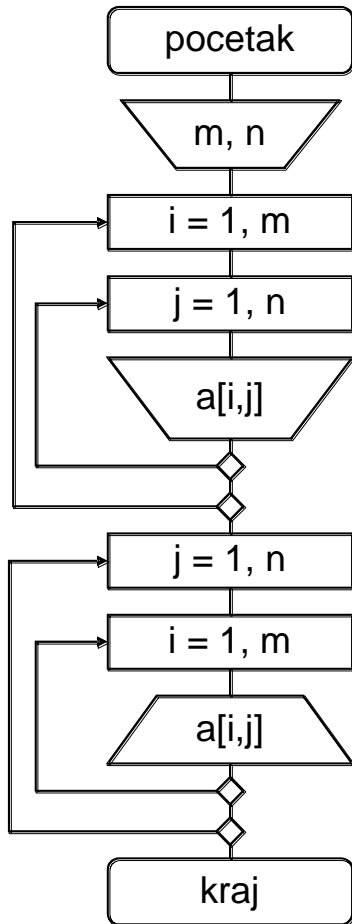
vrste

kolone

- Matrica kod koje je  $m = n$  zove se **kvadratna matrica**.
- Definisanje matrice:

**tip\_vrednosti** matrica[broj\_vrsta][broj\_kolona]

- Napisati algoritam i program kojim se za unete prirodne brojeve  $m$  i  $n$  ( $1 \leq m, n \leq 20$ ) koji predstavljaju dimenzije matrice, unose realni brojevi koji predstavljaju elemente matrice po vrstama, a zatim se dobijena matrica ispisuje po kolonama.



```

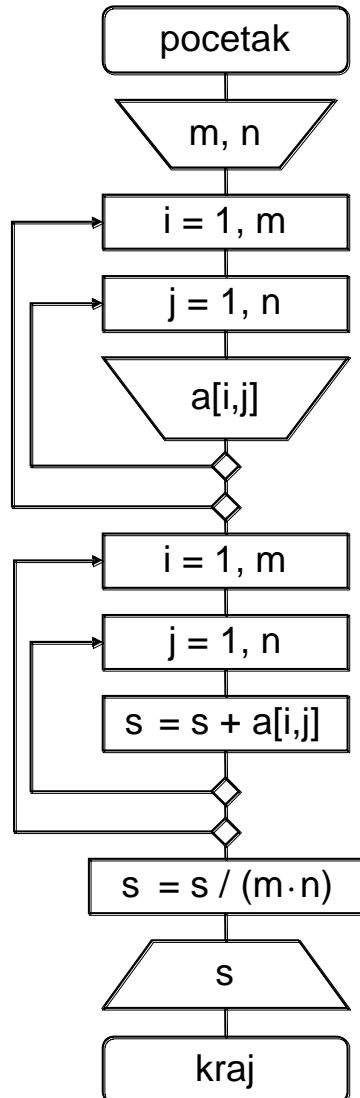
#include <stdio.h>

main()
{
    float a[20][20];
    int i,j,m,n;

    scanf("%d%d", &m, &n);
    for (i=0; i<m; i++)
        for (j=0; j<n; j++)
            scanf("%f", &a[i][j]);

    for (j=0; j<n; j++)
    {
        for (i=0; i<m; i++)
            printf("%8.2f", a[i][j]);
        printf("\n");
    }
}
  
```

- Napisati algoritam i program kojim se za unete prirodne brojeve  $m$  i  $n$  ( $1 \leq m, n \leq 20$ ) koji predstavljaju dimenzije matrice, unose celi brojevi koji predstavljaju elemente matrice po vrstama, a zatim za unetu matricu računa prosečna vrednost svih elemenata matrice.



```

#include <stdio.h>

main()
{
    int a[20][20];
    int i,j,m,n;
    float s;

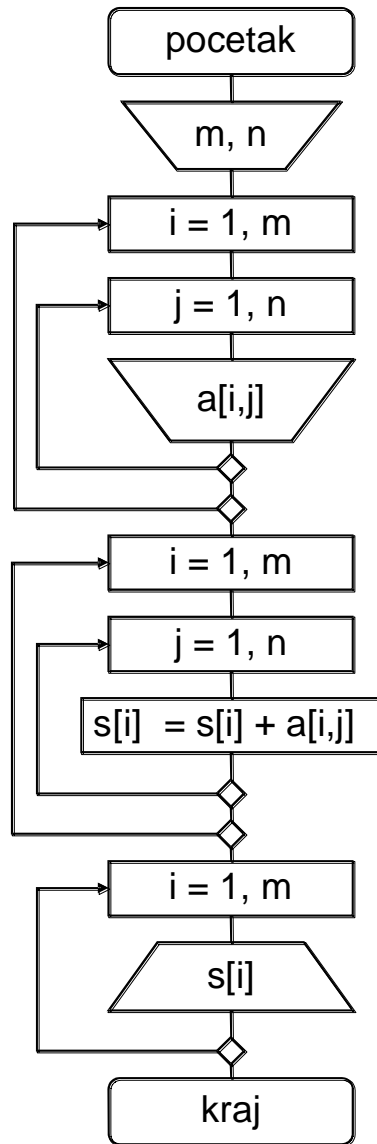
    scanf("%d%d",&m,&n);
    for (i=0;i<m;i++)
        for (j=0;j<n;j++)
            scanf("%d",&a[i][j]);

    s=0.0;
    for (i=0;i<m;i++)
        for (j=0;j<n;j++) s+=a[i][j];
    s=s/(m*n);

    printf("%8.2f\n",s);

}
  
```

- Napisati algoritam i program kojim se za unete prirodne brojeve  $m$  i  $n$  ( $1 \leq m, n \leq 20$ ) koji predstavljaju dimenzije matrice, unose celi brojevi koji predstavljaju elemente matrice po vrstama, a zatim formira niz koji sadrži zbir elemenata po vrstama.



```

#include <stdio.h>

main()
{
    int a[20][20], s[20];
    int i,j,m,n;

    scanf("%d%d", &m, &n);
    for (i=0; i<m; i++)
        for (j=0; j<n; j++)
            scanf("%d", &a[i][j]);

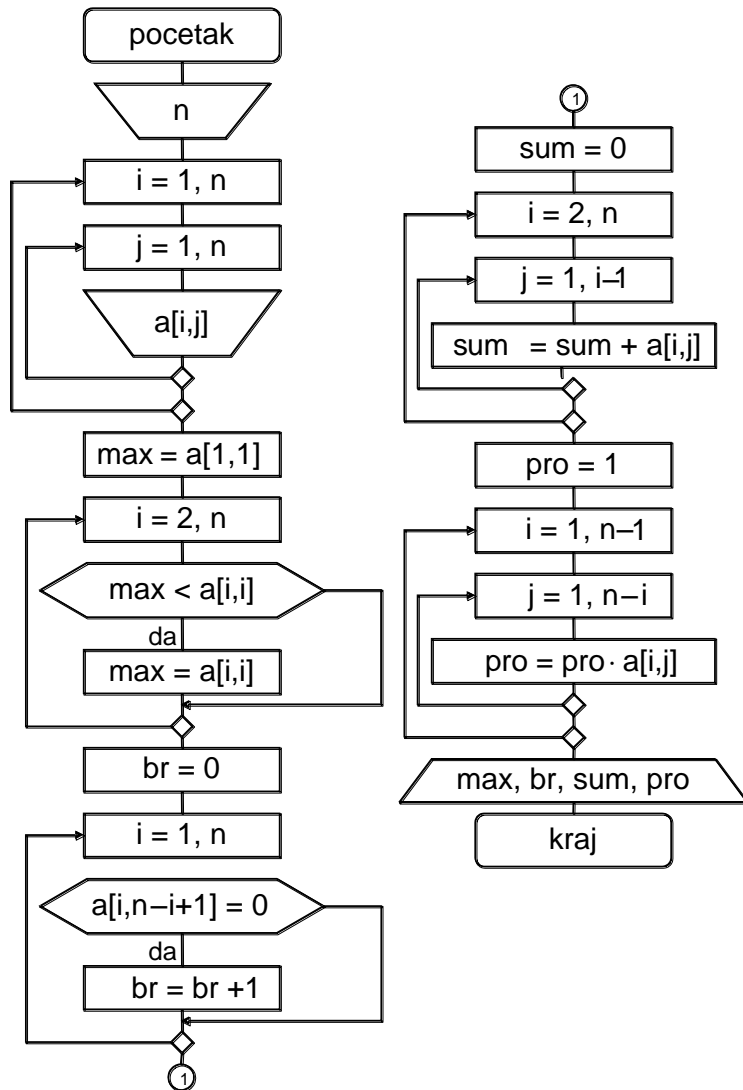
    for(i=0; i<m; i++)
        s[i]=0;

    for (i=0; i<m; i++)
        for (j=0; j<n; j++) s[i] = s[i] + a[i][j];

    for (i=0; i<m; i++)
        printf("%d\n", s[i]);
}
  
```



- Napisati algoritam i program kojim se za uneti prirodan broj  $n$  ( $1 \leq n \leq 20$ ) koji predstavlja dimenzije kvadratne matrice, unose celi brojevi koji predstavljaju elemente matrice po vrstama, a zatim izračunava:
  1. maksimum elemenata na glavnoj dijagonali,
  2. broj 0 na sporednoj dijagonali,
  3. zbir elemenata ispod glavne dijagonale
  4. proizvod elemenata iznad sporedne dijagonale



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

```
main()
```

```
{
```

```
    int a[20][20];
```

```
    int i,j,n;
```

```
    int max,br,sum,pro;
```

```
    scanf("%d",&n);
```

```
    for (i=0;i<n;i++)
```

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

```
            scanf("%d",&a[i][j]);
```

```
    max = a[0][0];
```

```
    for(i=1;i<n;i++)
```

```
        if (max < a[i][i]) max = a[i][i];
```

```
    printf("max = %d\n",max);
```

```
    br = 0;
```

```
    for(i=0;i<n;i++)
```

```
        if (a[i][n-i-1]==0) br++;
```

```
    printf("br = %d\n",br);
```

```
    sum=0;
```

```
    for(i=1;i<n;i++)
```

```
        for(j=0;j<i;j++)
```

```
            sum=sum+a[i][j];
```

```
    printf("sum = %d\n",sum);
```

```
    pro = 1;
```

```
    for (i=0;i<n-1;i++)
```

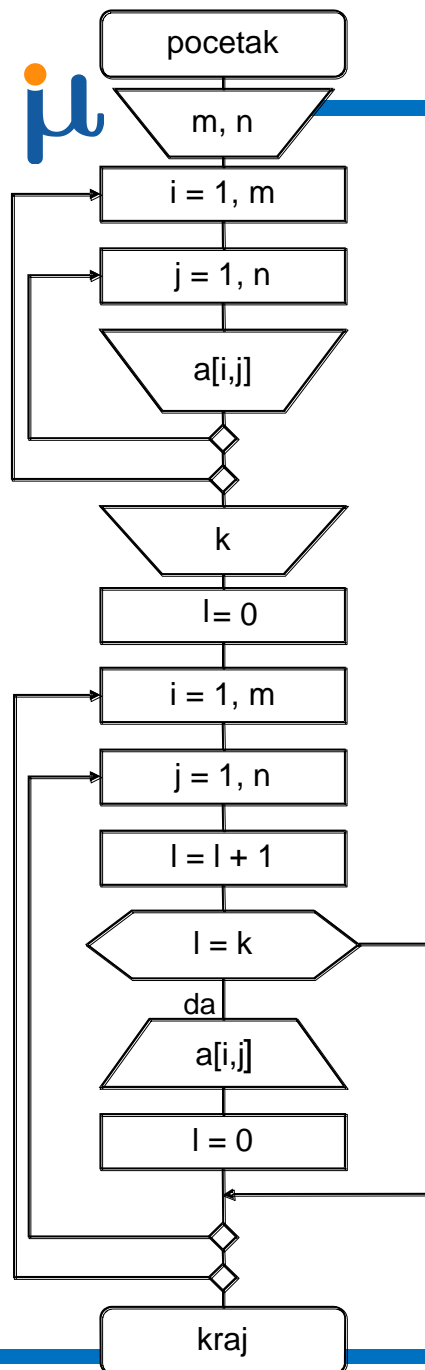
```
        for(j=0;j<n-i-1;j++)
```

```
            pro*=a[i][j];
```

```
    printf("pro = %d\n",pro);
```

```
}
```

- Napisati algoritam i program kojim se za unete prirodne brojeve  $m$  i  $n$  ( $1 \leq m, n \leq 20$ ) koji predstavljaju dimenzije matrice, unose celi brojevi koji predstavljaju elemente matrice po vrstama i prirodan broj  $k$ , a zatim se počev od elementa  $a_{00}$ , ispisuje svaki  $k$ -ti element matrice čitano po vrstama.



```
#include <stdio.h>

main()
{
    int a[20][20];
    int i,j,m,n,k;

    scanf("%d%d",&m,&n);
    for (i=0;i<m;i++)
        for (j=0;j<n;j++)
            scanf("%d",&a[i][j]);
    scanf("%d",&k);

    for (i=0;i<m;i++)
        for (j=0;j<n;j++)
            if((i*n+j+1)%k == 0) printf("%d\n",a[i][j]);
}
```

- Napisati algoritam i program kojim se za unete prirodne brojeve  $m$  i  $n$  ( $1 \leq m, n \leq 20$ ) koji predstavljaju dimenzije matrice, unose realni brojevi koji predstavljaju elemente matrice po vrstama, a zatim formira niz koji sadrži proizvod elemenata po kolonama.

- Napisati algoritam i program kojim se za unete prirodne brojeve  $m$  i  $n$  ( $1 \leq m, n \leq 20$ ) koji predstavljaju dimenzije matrice, unose realni brojevi koji predstavljaju elemente matrice po vrstama, a zatim formira niz koji sadrži parne elemente matrice.

- Napisati algoritam i program kojim se za unete prirodne brojeve  $m$  i  $n$  ( $1 \leq m, n \leq 20$ ) koji predstavljaju dimenzije matrice, unose celi brojevi koji predstavljaju elemente matrice po vrstama i prirodni brojevi  $l$  i  $k$  ( $1 \leq l, k \leq m$ ), a zatim razmenjuje vrednosti elementima u vrstama  $l$  i  $k$ .