ZADACI C# MATRICE

Sadržaj

[MATRICE-1, C# 2](#_Toc86761187)

[1. Пример:  Написати програм који учитава матрицу целих бројева димензије m × n рачуна суму свих елемената матрице. 3](#_Toc86761188)

[2. Пример : Написати програм који учитава квадратну матрицу *a* целих бројева димензије *n × n*. Одредити највећи и најмањи елемент матрицe, као и збир елемената на главној дијагонали 4](#_Toc86761189)

[3. Пример 3. Написати програм којим се за задату квадратну матрицу *A* димензије *n × n*, израчунава квадрат те матрице (*A × A*). 6](#_Toc86761190)

[4. Пример : Учитати природан број *n* и квадратну матрицу *A* реда *n*. Формирати транспоновану матрицу дате матрице *A*. 7](#_Toc86761191)

[5. Пример : Учитати матрицу димензија *n × m*. Сортирати елементе по врстама и колонама унете матрице, а затим исписати тако формирану матрицу. 9](#_Toc86761192)

[6. Zdatak: Za unetu numeričku matricu A[n,m], kreirati niz od M elemenata koji sadrži aritmetičke sredine svih kolona matrice. 11](#_Toc86761193)

[7. Zadatak: Za unetu kvadratnu numeričku matricu A[n,n], proveriti da li su jednaki svi zbirovi elemenata po vrstama i kolonama. 11](#_Toc86761194)

[8. Zadatak: Unos,ispis, suma glavne dijagonale, min,max član 12](#_Toc86761195)

[9. Zadatak: Preslikavanje kolona matrice po vertikalnoj osi 13](#_Toc86761196)

[10. Zadatak: Transponovanje matrice –(Aleksa T) 14](#_Toc86761197)

[11. Zadatak: Matrica dobijena množenjem jednodimenzionih nizova (Ognjen S) 14](#_Toc86761198)

[12. Zadatak: Kreirati niz kao aritmetičku sredinu kolona matrice (Ognjen S) 15](#_Toc86761199)

[13. Zadatak: Pomeranje kolona matrice u-levo za k-mesta 16](#_Toc86761200)

[14. Zadatak: Ispis 0 i 1 , kao šahovska tabla 17](#_Toc86761201)

[15. Zadatak: ISPIS članova matrice u smeru kazaljke na satu 18](#_Toc86761202)

[16. Zadatak: Kretanje po matrci po zadatoj putanji 18](#_Toc86761203)

[17. Zadatak: Korisnik unosi elemente matrice a[n,n]. Takodje unosi broj k, koji predstavlja indeks reda\kolone u kome svi elementi treba da dobiju vrednost zadatog broja n. Prikazati dobijenu matricu 19](#_Toc86761204)

[18. Zadaci za domaći rad: 20](#_Toc86761205)

# MATRICE-1, C#

Jednodimenzionalni niz je složen tip podataka čiji elementi moraju biti istog tipa. Elementi niza mogu biti i podaci čiji je tip niz pa dobijamo niz nizova, tj. **dvodimenzionalni ni**z.

Tako organizovan skup podataka u matematici se zove **matrica**, a u svakodnevnom životu **tabela**i predstavlja čest način čuvanja podataka (svejedno je koji termin koristite).

Horizontalne linije u matrici zovu se **redovi**( ili vrste ), a vertikalne **kolone** matrice. Matrica sa m redova i n kolona se obeležava sa **A**mxn , a brojevi m i  n  su  **dimenzije** matrice.

Element matrice A koji se nalazi u i -tom redu i  j -toj koloni se naziva (i, j)-ti element matrice A  i zapisuje kao A**i,j** ili A**[**i,j**].** Brojevi i, j zovu se **indeksi** elementa matrice.

red

|  |  |  |  |
| --- | --- | --- | --- |
| **A[0,0]**  KOLONA | **A[0,1]** | **A[0,2]** | **A[0,3]** |
| **A[1,0]** | **A[1,1]** | **A[1,2]** | **A[1,3]** |
| **A[2,0]** | **A[2,1]** | **A[2,2]** | **A[2,3]** |
| **A[3,0]** | **A[3,0]** | **A[3,2]** | **A[3,3]** |

Ako se u programu formira proizvoljna matrica i ne znaju dimenzije, onda se u deklaraciji navode maksimalne vrednosti redova i kolona, napr. A[20][30].

Elementu  matrice može da se pristupi direktno navođenjem indeksa, napr. A[3][5]=-0.23, a ako se prolazi kroz celu matricu onda se koriste dva for ciklusa (pri unosu ili štampanju elemenata, prilikom pretraživanja i drugih obrada elemenata).

napr.  for (i=0; i<m; i++)

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

s+=a[i][j];

**Matrica koja ima isti broj redova i kolona zove se kvadratna matrica.** U vezi ove matrice uvode se pojmovi: **glavna i sporedna dijagonala, elementi iznad i ispod glavne dijagonale**.

U vezi njih važe sledeća pravila:

**1) ako je i = j onda je a[i][j] NA GLAVNOJ DIJAGONALI,**

**2) ako je i<j onda je a[i][j] IZNAD GLAVNE DIJAGONALE,**

**3) ako je i>j onda je a[i][j] ISPOD GLAVNE DIJAGONALE,**

**4) ako je i+j = n-1 onda je a[i][j] NA SPOREDNOJ DIJAGONALI ( ako je prvi element a[0][0], oznaka u C# ).**

Članove matrice možemo uneti i sa tastature.

Метод GetLength има увек повратну вредност типа int и враћа број елемената некога низа. У нашој итерацији учествује при провери услова да се бројачи увек крећу по индексима редова и врста матрице.

# Пример:  Написати програм који учитава матрицу целих бројева димензије m × n рачуна суму свих елемената матрице.

**namespace suma\_elemenata\_matrice**

**{**

**class Program**

**{**

**static void Main(string[] args)**

**{**

**//Rezervisemo proizvoljan broj vrsta i kolona u memoriji**

**int[,] a = new int[ 30,40 ];**

**int i, j;**

**//m i n kao dimenzije matrice**

**int m, n;**

**int s = 0;**

**//Unosimo dimenzije matrice**

**Console.WriteLine("Unesite broj vrsta:");**

**m = int.Parse(Console.ReadLine());**

**Console.WriteLine("Unesite broj kolona:");**

**n = int.Parse(Console.ReadLine());**

**//Unosimo elemente matrice po vrstama**

**Console.WriteLine("Unesi clanove matrice:");**

**for (i = 0; i < m; i++)**

**{**

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

**{**

**Console.Write("clan[",i,j,"] = ");**

**a [i, j] = int.Parse(Console.ReadLine());**

**}**

**}**

**//Racunamo sumu svih elemenata matrice**

**for (i = 0; i < m; i++)**

**{**

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

**{**

**//Iteracijom po promenljivoj j kroz svaku i-tu vrstu dodajemo elemente u zbir s**

**s = s + a[i, j];**

**}**

**}**

**Console.WriteLine("Suma elemenata matrice je:" + s);**

**} }}**

# Пример : Написати програм који учитава квадратну матрицу *a* целих бројева димензије *n × n*. Одредити највећи и најмањи елемент матрицe, као и збир елемената на главној дијагонали

**namespace max\_min\_matrice**

**{**

**class Program**

**{**

**//Funkcija koja racuna min i max matrice**

**static void min\_max(int[,] a, int n)**

**{**

**int i, j;**

**//Postavljamo promenljive max i min na pocetne vrednosti**

**//Elemen a[0,0]se nalazi u gornjem levom uglu date matrice a**

**int max = a[0, 0];**

**int min = a[0, 0];**

**//Spoljasnjo for petljom prolazimo kroz matricu a po vrstama**

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

**{**

**//Unutrafor petljom prolazimo kroz matricu a po kolonama**

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

**{**

**//Proveravamo da li je a[i,j] manji od teku minimuma**

**if (a[i, j] < min)**

**{**

**//Ako je manji,onda njegovu dodelju promenlj min**

**min = a[i, j];**

**}**

**/\*U slucaju da je a[i,j] veci od tekuceg maksimuma, njegovu vrednost dodeljujemo**

**promenljivoj max\*/**

**if (a[i, j] > max)**

**{**

**max = a[i, j];**

**}**

**}**

**}**

**Console.WriteLine("Najveci element matrice je:", max,"a najmanji je:", min);**

**}**

**//Funkcija dijagonala izracunava zbir elemenata matrice na dijagonali**

**static void dijagonala(int[,] a, int n)**

**{**

**int i, suma = 0;**

**//Kako je i=j dovoljna nam je jedna for petlja po vrstama i kolonama matrice a**

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

**{**

**//Clanovi na dijagonali su oblika a[i,i]**

**suma += a[i, i];**

**}**

**Console.WriteLine("Suma elemenata na glavnoj dijagonali je {0}", suma);**

**}**

**static void Main(string[] args)**

**{**

**int[,] aMatrica = new int[10, 10];**

**int i, j, max, min, dim;**

**Console.WriteLine("Unesite dimenziju matrice");**

**dim = int.Parse(Console.ReadLine());**

**Console.WriteLine("Unesite elemente matrice");**

**for (i = 0; i < dim; i++)**

**{**

**for (j = 0; j < dim; j++)**

**{**

**Console.Write("clan[",i,j,"] = ");**

**aMatrica[i, j] = int.Parse(Console.ReadLine());**

**}**

**}**

**//Standardno ispisujemo elemente unete matrice iteracijom po vrstama i kolonama**

**for (i = 0; i < dim; i++)**

**{**

**for (j = 0; j < dim; j++)**

**{**

**Console.Write(aMatrica[i, j]);**

**}**

**Console.Write('\n');**

**}**

**//Pozivamo funckiju koja racuna min i max matrice aMatrica**

**min\_max(aMatrica, dim);**

**//Pozivamo funckiju koja racuna zbir elemenata na dijagonali matrice aMatrica**

**//Ispis rezultata se vrsi pri pozivu funkcija**

**dijagonala(aMatrica, dim);**

**} }}**

# Пример 3. Написати програм којим се за задату квадратну матрицу *A* димензије *n × n*, израчунава квадрат те матрице (*A × A*).

class Program

{

//Funkcija kvadriraj izracunava vrednost matrice A\*A

static void kvadriraj(int[,] A, int n)

{

/\*Definisemo kvadratnu matricu B dimenzija n, koja ce predstavljati

rezultat kvadriranja matrice A\*/

int[,] B = new int[n, n];

int i, j, k;

//Matricu B dobijamo tako sto mnozimo odgovarajuce vrste i kolone matrice A

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

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

{

/\*Pocetnu vrednost elementa matrice B na preseku i-te vrste i j-te kolone

postavljamo na 0, kako tekuci element dobijamo sabiranjem n proizvoda

elemenata matrice A\*/

B[i, j] = 0;

/\*Pomocna for petlja po promenljivoj k omogucava da se istovremeno pomnoze

elementi u i-toj vrsti i j-toj koloni matrice A\*/

for (k = 0; k < n; k++)

{

/\*Proizvoljni element B[i,j] dobijamo tako sto medjusobno pomnozimo elemente

i-te vrste i j-te kolone matrice A\*/

B[i, j] += A[i, k] \* A[k, j];

}

}

//Ispisujemo elemente rezultujuce matrice B

Console.WriteLine("Matrica B = A\*A:");

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

{

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

{

Console.Write(" " + B[i, j]);

}

Console.WriteLine();

}

}

static void Main(string[] args)

{

//Definisemo matricu A proizvoljno izabrane dimenzije

int[,] A = new int [20,20];

int dim, i, j;

Console.WriteLine("Unesite dimenziju kvadratne matrice");

dim = int.Parse(Console.ReadLine());

//Unosimo elemente matrice A

for (i = 0; i < dim; i++)

{

for (j = 0; j < dim; j++)

{

Console.Write("clan[",i,j,"] = ");

A[i, j] = int.Parse(Console.ReadLine());

}

}

//Pozivamo funkciju kvadriraj koja ce ispisati dobijenu kvadriranu matricu A\*A

kvadriraj(A,dim);

} }}

# Пример : Учитати природан број *n* и квадратну матрицу *A* реда *n*. Формирати транспоновану матрицу дате матрице *A*.

***Транспонована матрица се добија када одговарајуће врсте и колоне замене места.***

**class Program**

**{**

**//Funkcija transponuj formira transponovanu matricu date matrice A dimenzije n**

**static void transponuj(int[,] A, int n)**

**{**

**int i, j, pom;**

**/\*Prolazimo for petljom kroz elemente matrice A i vrsimo zamenu odgovarajucih vrsta**

**i kolona, npr. prva kolona postaje prva vrsta itd.\*/**

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

**{**

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

**{**

**//Vrsimo zamenu elemenata na poziciji A[i,j] elementima na poziciji A[j,i]**

**pom = A[i, j];**

**A[i,j] = A[j, i];**

**A[j, i] = pom;**

**}**

**}**

**//Ispisujemo matricu u transponovanom obliku**

**Console.WriteLine("Transponovana matrica ");**

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

**{**

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

**{**

**Console.Write( " " + a[i, j]);**

**}**

**Console.WriteLine();**

**}**

**}**

**//Glavna funkcija Main u kojoj unosimo proizvoljnu matricu poziva funkciju transponuj**

**static void Main(string[] args)**

**{**

**int[,] aMatrica = new int[15,15];**

**int i, j, dim;**

**Console.WriteLine("Unesite dimenziju kvadratne matrice");**

**dim = int.Parse(Console.ReadLine());**

**Console.WriteLine("Unesi elemente matrice:");**

**for (i = 0; i < dim; i++)**

**{**

**for (j = 0; j < dim; j++)**

**{**

**//clan[i,j]= vrednost aMatrica[i,j]**

**Console.Write("clan[",i,j,"] = ");**

**aMatrica[i, j] = int.Parse(Console.ReadLine());**

**}**

**}**

**//Pozivamo funkciju koja vrsi transponovanje unete matrice aMatrica**

**transponuj(aMatrica,dim); } }}**

# Пример : Учитати матрицу димензија *n × m*. Сортирати елементе по врстама и колонама унете матрице, а затим исписати тако формирану матрицу.

class Program

{

static void sortiranje\_kolona(int[,] matrica, int n, int m)

{

int i, j;

int pom, t;

for (j = 0; j < m; j++)

{

//Sortiramo kolone uz pomoc insertion sort algoritma koji nam je poznat od ranije

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

{

t = i;

while (t > 0 && matrica[t, j] < matrica[t-1, j])

{

pom = matrica[t, j];

matrica[t, j] = matrica[t - 1, j];

matrica[t - 1, j] = pom;

t--;

}

}

}

//Ispisujemo matricu ciji su elementi u svakoj koloni sortirani

Console.WriteLine("Matrica cije su kolone sortirane");

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

{

for (j = 0; j < m; j++)

{

Console.Write(matrica[i, j]);

}

Console.WriteLine('\n');

}

}

static void sortiranje\_vrsta(int[,] matrica,int n,int m)

{

int i, j, t;

int pom;

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

{

for (j = 0; j < m; j++)

{

t = j;

while(t > 0 && matrica[i, t]< matrica[i, t-1])

{

//Clan na poziciji [i,t] menja mesto sa clanom na poziciji [i,t-1]

pom = matrica[i, t];

matrica[i,t] = matrica[i, t-1];

matrica[i, t-1] = pom;

t--;

}

}

}

//Ispisujemo matricu kod koje su elementi u svakoj vrsti sortirani

Console.WriteLine("Matrica cije su vrste sortirane");

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

{

for (j = 0; j < m; j++)

{

Console.Write(matrica[i, j]);

}

Console.WriteLine('\n');

}

}

static void Main(string[] args)

{

int[,] A = new int[50, 50];

int i, j, m, n;

Console.WriteLine("Unesite broj vrsta matrice");

n = int.Parse(Console.ReadLine());

Console.WriteLine("Unesite broj kolona matrice");

m = int.Parse(Console.ReadLine());

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

{

for (j = 0; j < m; j++)

{

Console.Write("clan[{0},{1}] = ", i, j);

A[i, j] = int.Parse(Console.ReadLine());

}

}

sortiranje\_vrsta(A, n, m);

sortiranje\_kolona(A, n, m); } }}

# Zdatak: Za unetu numeričku matricu A[n,m], kreirati niz od M elemenata koji sadrži aritmetičke sredine svih kolona matrice.

**for(***j***=1**..*m***){**

*s***=0**;  
**for(***i***=1**..*n***){**

*s***+=***a***[***i***,***j***]**;

**}**  
*p***[***j***]=***s***/***n*;

**}**  
**write("Proseci kolona:")**;  
***ispisNiza*(***p***,***m***)**;

1. Za unete **numeričke nizove A[N] i B[M]**, **kreirati matricu C[NxM]** (od N redova i M kolona) koja se dobija **množenjem "vektora" A i B**.**for(***i***=1**..*n***){**

**for(***j***=1**..*m***){**

*c***[***i***,***j***]=***a***[***i***]\****b***[***j***]**;

**}}**  
**write("Matrica:")**;  
***ispisMatrice*(***c***,***n***,***m***)**;

1. Za zadate matrice A i B, od N redova i M kolona, **kreirati novu matricu C** čiji se svaki elemenat dobija kao **veći od odgovarajućih elemenata matrica A i B**.

**for(***i***=1**..*n***){**

**for(***j***=1**..*m***){**

*c***[***i***,***j***]=***b***[***i***,***j***]**;  
**if(***a***[***i***,***j***]>***b***[***i***,***j***]){**

*c***[***i***,***j***]=***a***[***i***,***j***]**;

**}**

**}}**  
**write("Matrica C")**;  
***ispisMatrice*(***c***,***n***,***m***)**;

# Zadatak: Za unetu kvadratnu numeričku matricu A[n,n], proveriti da li su jednaki svi zbirovi elemenata po vrstama i kolonama.

**sum=0;  
for(j=1..n){**

**sum+=a[1,j];**

**}  
ok=true;  
for(i=2..n){**

**s=0;  
for(j=1..n){**

**s+=a[i,j];**

**}  
if(sum<>s){**

**ok=false;**

**}**

**}  
if(ok){**

**for(j=1..n){**

**s=0;  
for(i=1..n){**

**s+=a[i,j];**

**}  
if(sum<>s){**

**ok=false;**

**}**

**}**

**}  
if(ok){**

**write("SVE SUME JEDNAKE");**

**}  
else{**

**write("Nisu sve sume jednake");**

# Zadatak: Unos,ispis, suma glavne dijagonale, min,max član

**using** System;    
**namespace** b1   
{   
    class Program   
    {   
        **public** static void **Main**(string[] args)   
        {   
            **int** min=**int**.*MaxValue*;   
            **int** max=**int**.*MinValue*;   
            **int** n=**int**.**Parse**(Console.**ReadLine**());   
            **int** m=**int**.**Parse**(Console.**ReadLine**());   
               
            **int** []z=**new** **int**[n];   
            **int** [,] matr=**new** **int**[n,m];   
               
            //unos matrice   
            **for** (**int** i = 0; i < n; i++)    
            {   
                **for** (**int** j = 0; j < m; j++)    
                {   
                    Console.**Write**("clan [{0},{1}]=" ,i,j);   
                    matr[i,j]=**int**.**Parse**(Console.**ReadLine**());   
                }   
                Console.**WriteLine**();   
            }   
            Console.**WriteLine**();   
            **int** sum=0;   
            **int** s=0;               
            //suma glavne dijagonale   
            **for** (**int** i = 0; i < n; i++) {   
                **for** (**int** j = 0; j < m; j++) {   
                    **if** (i==j)   
                    {   
                        sum=sum+matr[i,j];   
                    }                 }             }   
            //minimalan clan matrice    
            **for** (**int** i = 0; i < n; i++) {   
                **for** (**int** j = 0; j < m; j++) {   
                    **if** (min>matr[i,j]) {   
                          
                        min=matr[i,j];   
                    }                 }             }   
            //maksimalan clan matrice        
            **for** (**int** i = 0; i < n; i++) {   
                **for** (**int** j = 0; j < m; j++) {   
                    **if** (max<matr[i,j]) {   
                           
                        max=matr[i,j];   
                    }                 }             }   
            //ispis normalna   
               
            **for** (**int** i = 0; i < n; i++)    
            {   
                **for** (**int** j = 0; j < m; j++)    
                {   
                    Console.**Write**(matr[i,j]+" ");   
                       
                }   
                Console.**WriteLine**();   
            }   
            Console.**WriteLine**();   
            Console.**WriteLine**("zbir glavne dijagonal je {0}", sum); //ispis sume dijagonale   
            Console.**WriteLine**("najmanji {0}", min); //ispis najmanjeg   
            Console.**WriteLine**("najeci {0}", max);//ispis najveceg   
            Console.**WriteLine**(); 

# Zadatak: Preslikavanje kolona matrice po vertikalnoj osi

class Program

{

static int[,] UnesiMat() //funkcija(metoda) za unos matrice

{

Console.WriteLine("Unesite dimenzije matrice:n,m");

int n = int.Parse(Console.ReadLine());//broj vrsta

int m = int.Parse(Console.ReadLine());//broj kolona

int[,] mat = new int[n, m];

Random r = new Random();

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

{

for (int j = 0; j < m; j++)

mat[i, j] = r.Next(11); //generise slucajne brojeve od 0-10, za 1 manje od 11

}

return mat;

}

static void Stampa(int[,] mat)//ova f-ja prikazuje(stampa) matricu

{

Console.WriteLine("Prikaz matrice");

for (int i = 0; i < mat.GetLength(0); i++)

{

for (int j = 0; j < mat.GetLength(1); j++)

Console.Write("{0,4}", mat[i, j]);

Console.WriteLine();

}

}

static void VertPresl(int[,] mat)

{

int n = mat.GetLength(0);

int m = mat.GetLength(1);

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

for (int j = 0; j < m / 2; j++)

{

int pom = mat[i, j];

mat[i, j] = mat[i, m - 1 - j];

mat[i, m - 1 - j] = pom;

}

}

static void Main(string[] args)

{

//preslikava kolone matrice po vertikalnoj osi,

int[,] mat = UnesiMat();

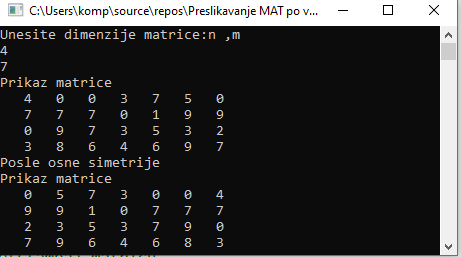
Stampa(mat);

VertPresl(mat);

Console.WriteLine("Posle osne simetrije");

Stampa(mat);

Console.ReadLine(); } }}



# Zadatak: Transponovanje matrice –(Aleksa T)

int n, m;

string[] unos = Console.ReadLine().Split();

n = int.Parse(unos[0]);

m = int.Parse(unos[1]);

int[,] a = new int[n, m];

for(int  i=0;i<m;i++)

{

    unos = Console.ReadLine().Split();

    for (int j = 0; j < n; j++)

        a[j, i] = int.Parse(unos[j]);

}

for(int i=0;i<n;i++)

{

    for (int j = 0; j < m; j++)

        Console.Write(a[i, j] + " ");

    Console.WriteLine();

}

# Zadatak: Matrica dobijena množenjem jednodimenzionih nizova (Ognjen S)

    class Program

    {

        public static void Main(string[] args)

        {

            Console.WriteLine("unesi broj el. niza a");

            int n = int.Parse(Console.ReadLine());

            Console.WriteLine("unesi broj el. niza b");

            int m = int.Parse(Console.ReadLine());

            int[] a = new int[n];

            int[] b = new int[m];

            int[,] c = new int[n, m];

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

            {

                a[i] = int.Parse(Console.ReadLine());

            }

            Console.WriteLine();

            for (int i = 0; i < m; i++)

            {

                b[i] = int.Parse(Console.ReadLine());

            }

            Console.WriteLine();

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

            {

                for (int j = 0; j < m; j++)

                {

                    c[i, j] = a[i] \* b[j];

                }

            }

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

            {

                for (int j = 0; j < m; j++)

                {

                    COonsole.Write(c[i,j]+" ");

                }

                Console.WriteLine();

            }           }     } }

# Zadatak: Kreirati niz kao aritmetičku sredinu kolona matrice (Ognjen S)

namespace lkod

{

    class Program

    {

        public static void Main(string[] args)

        {

            int n = int.Parse(Console.ReadLine());

            int m = int.Parse(Console.ReadLine());

            double[] z = new double[n];

            int[,] a = new int[n, m];

            double s;

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

            {

                for (int j = 0; j < m; j++)

                {

                    Console.Write("clan [{0},{1}]=", i, j);

                    a[i, j] = int.Parse(Console.ReadLine());

                }

                Console.WriteLine();

            }

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

            {

                for (int j = 0; j < m; j++)

                {

                    Console.Write(a[i, j] + " ");

                }

                Console.WriteLine();

            }

            for (int j = 0; j < m; j++)

            {

                s = 0;

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

                {

                    s += a[i, j];

                }

                z[j] = s/n;

            }

            for (int i = 0; i < m; i++)

            {

                Console.WriteLine("{0:0.00}",z[i]);

            }         }     } }

# Zadatak: Pomeranje kolona matrice u-levo za k-mesta

class Program

{

static int[,] UnesiMat() //funkcija(metoda) za unos matrice

{

Console.WriteLine("Unesite dimenzije matrice:n,m");

int n = int.Parse(Console.ReadLine());//broj vrsta

int m = int.Parse(Console.ReadLine());//broj kolona

int[,] mat = new int[n, m];

Random r = new Random();

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

{

for (int j = 0; j < m ; j++)

mat[i, j] = r.Next(11); //generise slucajne brojeve od 0-10, za 1 manje od 11

}

return mat;

}

static void Stampa(int[,] mat)//ova f-ja prikazuje(stampa) matricu

{

Console.WriteLine("Prikaz matrice");

for (int i = 0; i < mat.GetLength(0); i++)

{

for (int j = 0; j< mat.GetLength(1); j++)

Console.Write("{0,4}", mat[i, j]);

Console.WriteLine();

}

}

static void PomeriLevoZa1(int[,] mat)

{

int n = mat.GetLength(0);

int m = mat.GetLength(1);

int[] POM = new int[n];//pom-pomocni niz

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

POM[i] = mat[i, 0];//samo nultu kolonu kopiramo u niz POM[]

for (int i = 1; i < m; i++)

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

mat[j, i - 1] = mat[j, i];

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

mat[i, m - 1] = POM[i];

}

static void PomeriLevoZak(int[,] mat,int k)

{

for (int i = 0; i < k; i++)

PomeriLevoZa1(mat);

}

static void Main(string[] args)

{

int[,] mat = UnesiMat();

Stampa(mat);

PomeriLevoZa1(mat);

Stampa(mat);

Console.WriteLine("Unesi k=");

int k = int.Parse(Console.ReadLine());

PomeriLevoZak(mat,k);

Stampa(mat);

Console.ReadLine();

} }}

# Zadatak: Ispis 0 i 1 , kao šahovska tabla

class Program {

static int[,] MenjaMat(int[,] mat) //funkcija(metoda) za unos matrice

{

for (int i = 0; i < mat.GetLength(0); i++)

{

for (int j = 0; j < mat.GetLength(1); j++)

if ((i+j)%2!=0)

{

mat[i, j] = 1;

}

}

return mat;

}

static int[,] UnesiMat() //funkcija(metoda) za unos matrice

{

Console.WriteLine("Unesite dimenzije matrice:n,m");

int n = int.Parse(Console.ReadLine());//broj vrsta

int m = int.Parse(Console.ReadLine());//broj kolona

int[,] mat = new int[n, m];

Random r = new Random();

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

{

for (int j = 0; j < m; j++)

mat[i, j] = r.Next(0); //generise slucajne brojeve od 0

}

return mat;

}

static void Stampa(int[,] mat)//ova f-ja prikazuje(stampa) matricu

{

Console.WriteLine("Prikaz matrice");

for (int i = 0; i < mat.GetLength(0); i++)

{

for (int j = 0; j < mat.GetLength(1); j++)

Console.Write("{0,4}", mat[i, j]);

Console.WriteLine();

}

}

static void Main(string[] args)

{

//ispisuje 1 na mestima gde je zbir indeksa neparan

int[,] mat = UnesiMat();

Stampa(mat);

MenjaMat(mat);

Stampa(mat);

Console.ReadLine(); } }}

# Zadatak: ISPIS članova matrice u smeru kazaljke na satu

static void Stampa(int[,] mat)//ova f-ja prikazuje(stampa) matricu

{

Console.WriteLine("Prikaz matrice");

for (int i = 0; i < mat.GetLength(0); i++)

{

for (int j = 0; j < mat.GetLength(1); j++)

Console.Write("{0,4}", mat[i, j]);

Console.WriteLine();

}

}

static void Main(string[] args)

{//ISPIS članova matrice u smeru kazaljke na satu

Console.WriteLine("unesi dimenziju matrice");

int n = int.Parse(Console.ReadLine());

int [,] mat = new int[n, n];

int gore = 0,dole = n - 1,levo = 0,desno = n - 1;//

int br = 1;//broj od kog pocinjemo upis u matricu

while (br <= n \* n)

{

for (int i = levo; i <= desno; i++)

mat[gore, i] = br++;//upisujemo vrstu-GORNJU

gore++;

for (int i = gore; i <= dole; i++)

mat[i, desno] = br++;//upisujemo kolonu-DESNO

desno--;

for (int i = desno; i >= levo; i--)

mat[dole, i] = br++;//upisujemo vrstu-DOLE

dole--;

for (int i = dole; i >=gore; i--)

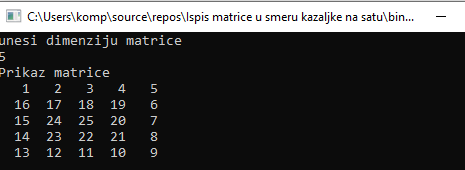
mat[i, levo] = br++;//upisujemo kolonu-LEVO

levo++;

}

Stampa(mat);

Console.ReadLine(); } }}



# Zadatak: Kretanje po matrci po zadatoj putanji

namespace KretanjePoPutanji

{

class Program

{

static void Stampa(int[,] matr)//ova f-ja prikazuje(stampa) matricu

{

Console.WriteLine("Prikaz matrice");

for (int i = 0; i < matr.GetLength(0); i++)

{

for (int j = 0; j < matr.GetLength(1); j++)

Console.Write("{0,4}", matr[i, j]);

Console.WriteLine();

}

}

static void Main(string[] args)

{ // Zadtak: Krecemo se po matrici i sabiramo clanove na kojima smo. Putanja se zadaje slovima:N-gore; S-dole;E-desno;W-leo

Console.WriteLine("Unesite broj redova matice n: ");

int n = int.Parse(Console.ReadLine());

Console.WriteLine("Unesite broj kolona matrice, m: ");

int m = int.Parse(Console.ReadLine());

Random r = new Random();

int[,] matr = new int[n, m];

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

{

for (int j = 0; j < m; j++)

matr[i, j] = r.Next(1, 11);

}

Stampa(matr);

//int v = 1, k = 3;

Console.WriteLine("pocetna pozcija od koje pocinjemo kretanje i sabiranje clanova matrice");

Console.WriteLine("Unos pocetne pozicije");

Console.WriteLine("Unesite pocetnu poziciju vrste v= ");

int v = int.Parse(Console.ReadLine());

Console.WriteLine("Unesite pocetnu poziciju kolone k= ");

int k = int.Parse(Console.ReadLine());

string put = "SEEN";

int S = matr[v, k];//sumiramo clanove matrice, pocena vrednost je

bool ispravno = true;

for (int i = 0; i < put.Length; i++)

{

if (put[i] == 'N') v--;//kretanje na gore

if (put[i] == 'S') v++;

if (put[i] == 'E') k++;

if (put[i] == 'W') k--;

if (v < 0 || v > n - 1 || k < 0 || k > m - 1)

{

Console.WriteLine("Neispravan put");

ispravno = false;

break;

}

S = S + matr[v, k];

}

if (ispravno) Console.WriteLine("Na putu {0}, suma elemenata je S= {1}", put, S);

Console.ReadLine();

}

}

}

put = "SEEN";

put[0]=’S’

put[1]=’S’

# Zadatak: Korisnik unosi elemente matrice a[n,n]. Takodje unosi broj k, koji predstavlja indeks reda\kolone u kome svi elementi treba da dobiju vrednost zadatog broja n. Prikazati dobijenu matricu

namespace IspisKoloneVrsteK

{

class Program

{

static void Stampa(int[,] matr)//ova f-ja prikazuje(stampa) matricu

{

Console.WriteLine("Prikaz matrice");

for (int i = 0; i < matr.GetLength(0); i++)

{

for (int j = 0; j < matr.GetLength(1); j++)

Console.Write("{0,4}", matr[i, j]);

Console.WriteLine();

}

}

static void Main(string[] args)

{

Console.WriteLine("Unesite broj redova matice n: ");

int n = int.Parse(Console.ReadLine());

Random r = new Random();

int[,] matr = new int[n, n];

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

{

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

matr[i, j] = r.Next(0, 0);

}

Stampa(matr);

Console.WriteLine("Unesite kolonu i vrstu k=");

int k= int.Parse(Console.ReadLine());

Console.WriteLine("Unesite vrednost zadatog broja za ispis ");

int br = int.Parse(Console.ReadLine());

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

{

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

if (i==k||j==k)

matr[i, j] = br;

}

Stampa(matr);

Console.ReadLine();

}

}

}

# Zadaci za domaći rad:

|  |
| --- |
| Suma po vrstama matrice |
| Proizvod elementata matrice na sporednoj dijagonali je  Minimalan elementat matrice na sporednoj dijagonali je  Maksimalan elementat matrice na sporednoj dijagonali je |
| Suma elementata matrice na glavnoj dijagonali je  Suma elementata matrice na sporednoj dijagonali je- |
| ucitavanje matrice po vrstama i ispis |
| Da li je matrica-Gornja trougaona, ima sve nule ispod glavne dijagonale |
| Korisnik unosi elemente matrice a[4][4]. Takodje unosi broj k, koji predstavlja indeks reda\kolone u kome svi elementi treba da dobiju vrednost zadatog broja n. Prikazati dobijenu matricu. |
| Napisati funkciju koja u matrici a[4][4] dodeljuje vrednost n svim elementima iznad glavne dijagonale, k svim elementima ispod glavne dijagonale i m svim elementima na glavnoj dijagonali. Takodje odrediti sumu elemenata na sporednoj dijagonali. Prikazati matricu I trazeni rezultat. |
| Formirati i prikazati na ekran matricu a[8][8] po ugledu na shahovsku tablu (belo=x, crno=o); |
| Korisnik unosi elemente matrice a[4][4]. Formirati niz b[4], tako da svaki njegov element predstavlja sumu odgovarajuceg rada matrice a. |
| Za zadatu matricu, prona ć i maksimum svih parnih i minimum svih neparnih kolona. |
| Izra čunati aritmeti čku sredinu elemenata na sporednoj dijagonali kvadratne matrice od N elemenata. |
| Prona ć i indeks najve ćeg elementa na glavnoj dijagonali kvadratne matrice. |