

Sesiunea 8

- Agenda
 - Terminare continut sesiune 7 (nu am apucat sa vorbim asa mult de matricile patratice)
 - Rezolvare tema
 - Exerciții "clasa"
 - Exerciții pentru acasa

Terminare continut sesiune 7

- Pareri/dificultati vectori de frecventa
- Pareri/dificultati matrici (e.g vecinii unui numar)

Exerciții clasa

- 1 => Subiect Bacalureat 2025 Iunie varianta de rezerva - Subiectul III - Ex. 2

```
#include <iostream>
using namespace std;

int main() {
    int nr, np;
    cin >> nr >> np;
    int parcare[51][51]; // dimensiuni maxime conform enunțului

    for (int i = 0; i < nr; i++) {
        for (int j = 0; j < np; j++) {
            cin >> parcare[i][j];
        }
    }

    int contorTriplete = 0;

    // verificăm doar primul rând (i=0) și ultimul rând (i=nr-1)
    for (int i = 0; i < nr; i++) {
        if (i == 0 || i == nr-1) {
            for (int j = 0; j <= np - 3; j++) { // mergem doar până la
np-3
                if (parcare[i][j] == 0 && parcare[i][j+1] == 0 &&
parcare[i][j+2] == 0) {
                    contorTriplete++;
                }
            }
        }
    }

    cout << contorTriplete << endl;
    return 0;
}
```

- 2 Subiect Bacalureat 2025 August - Subiectul III - Ex. 2
 - Practic problema presupune urmatoarii pasi:
 - Identificarea valorii minime
 - Salvarea valorii pe care vrem sa o folosim cand vrem sa inlocuim
 - Nota: trebuie sa faci asta inainte de a incepe inlocuirea deoarece este posibil ca si in ultima coloana sa inlocuim.
 - Iteram prin matrice si cand gasim un element egal cu valoarea minima, inlocuim toate elementele de pe coloana sa cu elementul dorit.

```
#include <iostream>

using namespace std;

int main() {

    int m, n;
    cin >> m >> n;
    int matrice[m][n];

    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            cin >> matrice[i][j];
        }
    }

    int minim = 101;
    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            if (matrice[i][j] < minim) {
                minim = matrice[i][j];
            }
        }
    }

    int elementPentruInlocuire = matrice[m-1][n-1];
    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            if (matrice[i][j] == minim) {
                for (int k = 0; k < m; k++) {
                    matrice[k][j] = elementPentruInlocuire;
                }
            }
        }
    }

    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            cout << matrice[i][j] << " ";
        }
        cout << endl;
    }
}
```

```
    }  
  
    return 0;  
}
```

Rezolvare tema

1. Rezolvare:

```
#include <iostream>  
  
using namespace std;  
  
int main() {  
    int n,m;  
    cin >> n >> m;  
    int matrice[n][m];  
  
    for (int i = 0; i < n; i++) {  
        for (int j = 0; j < m; j++) {  
            cin >> matrice[i][j];  
        }  
    }  
  
    for (int i = 0; i < n; i++) {  
        for (int j = 0; j < m; j++) {  
            cout << matrice[i][j] << " ";  
        }  
        cout << endl;  
    }  
    return 0;  
}
```

2. Rezolvare:

```
#include <iostream>  
  
using namespace std;  
  
int main() {  
    int n,m;  
    cin >> n >> m;  
    int matrice[n][m];  
    int suma = 0;
```

```
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            cin >> matrice[i][j];
        }
    }

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            suma += matrice[i][j];
        }
    }

    cout << suma;

    return 0;
}
```

3. Rezolvare:

```
#include <iostream>

using namespace std;

int main() {
    int n,m;
    cin >> n >> m;
    int matrice[n][m];

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            cin >> matrice[i][j];
        }
    }

    for (int i = 0; i < n; i++) {
        int sumaLinie = 0;
        for (int j = 0; j < m; j++) {
            sumaLinie += matrice[i][j];
        }
        cout << "Linia " << (i+1) << ": " << sumaLinie << endl;
    }

    return 0;
}
```

4. Rezolvare

```
#include <iostream>

using namespace std;

int main() {
    int n,m;
    cin >> n >> m;
    int matrice[n][m];

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            cin >> matrice[i][j];
        }
    }

    for (int i = 0; i < m; i++) {
        int sumaColoana = 0;
        for (int j = 0; j < n; j++) {
            sumaColoana += matrice[j][i];
        }
        cout << "Coloana " << (i+1) << ": " << sumaColoana << endl;
    }

    return 0;
}
```

- Nota: in enunt, exemplul este gresit. Suma a fost calculata eronat.

5. Rezolvare

```
#include <iostream>

using namespace std;

int main() {
    int n;
    cin >> n;
    int matrice[n][n];

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cin >> matrice[i][j];
        }
    }

    int suma = 0;
    cout << "Elemente diagonala: ";
    for (int i = 0; i < n; i++) {
```

```

        for (int j = 0; j < n; j++) {
            if (i == j) {
                cout << matrice[i][j] << " ";
                suma += matrice[i][j];
            }
        }
    }
    cout << "Suma: " << suma << endl;

    return 0;
}

```

6. Rezolvare:

```

#include <iostream>

using namespace std;

int main() {
    int n;
    cin >> n;
    int matrice[n][n];

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cin >> matrice[i][j];
        }
    }

    int produs = 1;
    cout << "Elemente diagonala secundara: ";
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            if (i + j == n - 1) {
                cout << matrice[i][j] << " ";
                produs *= matrice[i][j];
            }
        }
    }
    cout << "Produs: " << produs << endl;

    return 0;
}

```

7. Rezolvare

```

#include <iostream>

```

```
using namespace std;

int main() {
    int n;
    cin >> n;
    int matrice[n][n];

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cin >> matrice[i][j];
        }
    }

    cout << "Deasupra: ";
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            if (j > i) {
                cout << matrice[i][j] << " ";
            }
        }
    }

    cout << endl << "Sub: ";
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            if (j < i) {
                cout << matrice[i][j] << " ";
            }
        }
    }

    return 0;
}
```

8. Rezolvare

```
#include <iostream>

using namespace std;

int main() {
    int n;
    cin >> n;
    int matrice[n][n];

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cin >> matrice[i][j];
        }
    }
}
```

```
int esteSimetrica = 1;
for (int i = 0; i < n; i++) {
    for (int j = 0; j < n; j++) {
        if (matrice[i][j] != matrice[j][i]) {
            esteSimetrica = 0;
            break;
        }
    }
}

if (esteSimetrica) {
    cout << "Matricea este simetrica";
} else {
    cout << "Matricea nu este simetrica";
}

return 0;
}
```

9. Rezolvare

```
#include <iostream>

using namespace std;

int main() {
    int n, m;
    cin >> n >> m;
    int matrice[n][m];
    int transpusa[m][n];
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            cin >> matrice[i][j];
        }
    }

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            transpusa[j][i] = matrice[i][j];
        }
    }

    for (int i = 0; i < m; i++) {
        for (int j = 0; j < n; j++) {
            cout << transpusa[i][j] << " ";
        }
        cout << endl;
    }
}
```



```
    return 0;
}
```

10. Rezolvare

- Nota: cand inmultim doua matrici (`matrice1[x][y]` si `matrice2[w][z]`) rezultatul este tot o matrice dar care va avea acelasi numar de linii ca matrice1 (adica x) si acelasi numar de coloane ca matricea a doua, deci avem matricea `rezultat[x][z]`
- Pentru a calcula rezultatul practic

```
#include <iostream>

using namespace std;

int main() {
    int n;
    cin >> n;
    int matrice1[n][n];
    int matrice2[n][n];
    int produs[n][n];

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cin >> matrice1[i][j];
        }
    }

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cin >> matrice2[i][j];
        }
    }

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            int suma = 0;
            for (int k = 0; k < n; k++) {
                suma += matrice1[i][k] * matrice2[k][j];
            }
            produs[i][j] = suma;
        }
    }

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cout << produs[i][j] << " ";
        }
        cout << endl;
    }
}
```

```
        return 0;
    }
```

11. Rezolvare:

```
#include <iostream>
using namespace std;

int main() {
    int n, zona;
    cin >> n >> zona;
    int matrice[n][n];
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cin >> matrice[i][j];
        }
    }
    if (zona == 1) {
        int suma = 0;
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                if (j > i && i+j < n - 1) {
                    suma += matrice[i][j];
                }
            }
        }
        cout << suma;
    } else if (zona == 2) {
        int suma = 0;
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                if (j > i && i + j > n - 1) {
                    suma += matrice[i][j];
                }
            }
        }
        cout << suma;
    } else if (zona == 3) {
        int suma = 0;
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                if (i > j && i + j > n - 1) {
                    suma += matrice[i][j];
                }
            }
        }
        cout << suma;
    } else if (zona == 4) {
        int suma = 0;
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                if (i > j && i + j < n - 1) {
```

```

        suma += matrice[i][j];
    }
}
}
cout << suma;
}
return 0;
}

```

12. Rezolvare

```

#include <iostream>

using namespace std;

int main() {

    int n;
    cin >> n;
    int matrice[n][n];
    int estePatratMagic = 1;
    int sumaDiagonalaPrincipala = 0, sumaDiagonalaSecundara=0;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            cin >> matrice[i][j];
            if (matrice[i][j] > n * n || matrice[i][j] < 1) {
                estePatratMagic = 0;
                break;
            }
        }
    }

    if (!estePatratMagic) {
        cout << "false";
        return 0;
    }

    for (int i = 0; i < n; i++){
        sumaDiagonalaPrincipala += matrice[i][i]; // 00 11 22 33
        sumaDiagonalaSecundara += matrice[i][n-1-i];
    }

    if (sumaDiagonalaSecundara != sumaDiagonalaPrincipala) {
        estePatratMagic = 0;
    } else {
        for (int i = 0; i < n; i++) {
            int sumaLinieCurenta = 0;
            int sumaColoanaCurenta = 0;
            for (int j = 0; j < n; j++) {
                sumaLinieCurenta += matrice[i][j];
                sumaColoanaCurenta += matrice[j][i];
            }
        }
    }
}

```

```

    }

    if (sumaLinieCurenta != sumaDiagonalaPrincipala ||
        sumaColoanaCurenta != sumaDiagonalaPrincipala) {
        estePatratMagic = 0;
        break;
    }
}

if (estePatratMagic == 1) {
    cout << "true";
} else {
    cout << "false";
}

return 0;
}

```

Exercitii pentru acasa

1. Se citeste de la tastatura o matrice patratica unde fiecare celula reprezinta un pixel dintr-o poza. Sa se construiasca in memorie o matrice care reprezinta imaginea blurata. O imagine se blureaza prin setarea fiecarui pixel cu media valorii vecinilor sai.

- Exemplu:

Pentru $n = 4$ si matricea

```

1  2  3  4
5  6  7  8
9  10 11 12
13 13 15 16

```

Obtinem matricea:

```

3.5  3.33  5.5  5.5
5.33  6    7   7.67
10   10   11  11.67
11.5 12.67 13.67 13.5

```

- Nota: e posibil ca a doua virgula sa difere la tine in rezultat insa e ok.

2. Se citeste de la tastatura o matrice patratica. Daca desenam cele doua diagonale, se creeaza 4 zone, pe care le putem considera nord, sud, est,vest. Scrie un program care afiseaza diferenta dintre suma zonelor est-vest si suma zonelor nord-sud excluzand elementele care se afla fix pe una din cele 2 diagonale.

- Exemplu:

Pentru $n = 5$ si matricea

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

Suma nord: $2 + 3 + 4 + 8 = 17$

Suma sud: $18 + 22 + 23 + 24 = 87$

Suma vest: $6 + 11 + 12 + 16 = 45$

Suma est: $10 + 14 + 15 + 20 = 59$

Suma (nord+sud) = 104

Suma (est+vest) = 104

Diferanta = $104 - 104 = 0$;

3. Exercițiu Subiect III - E 2 din modelul propus pentru BAC anul 2025

(https://www.pbinfo.ro/resurse/9dc152/examene/2025/E_d_informatica_2025_sp_MI_C_var_model.pdf)

4. Exercițiu Subiectul II - E 3 Varianta Bac Iunie 2024

(https://www.pbinfo.ro/resurse/9dc152/examene/2024/E_d_Informatica_2024_sp_MI_C_var_03_LRO.pdf)