Tema 3

Traistaru Alexandru Mihai

Grupa 324AA

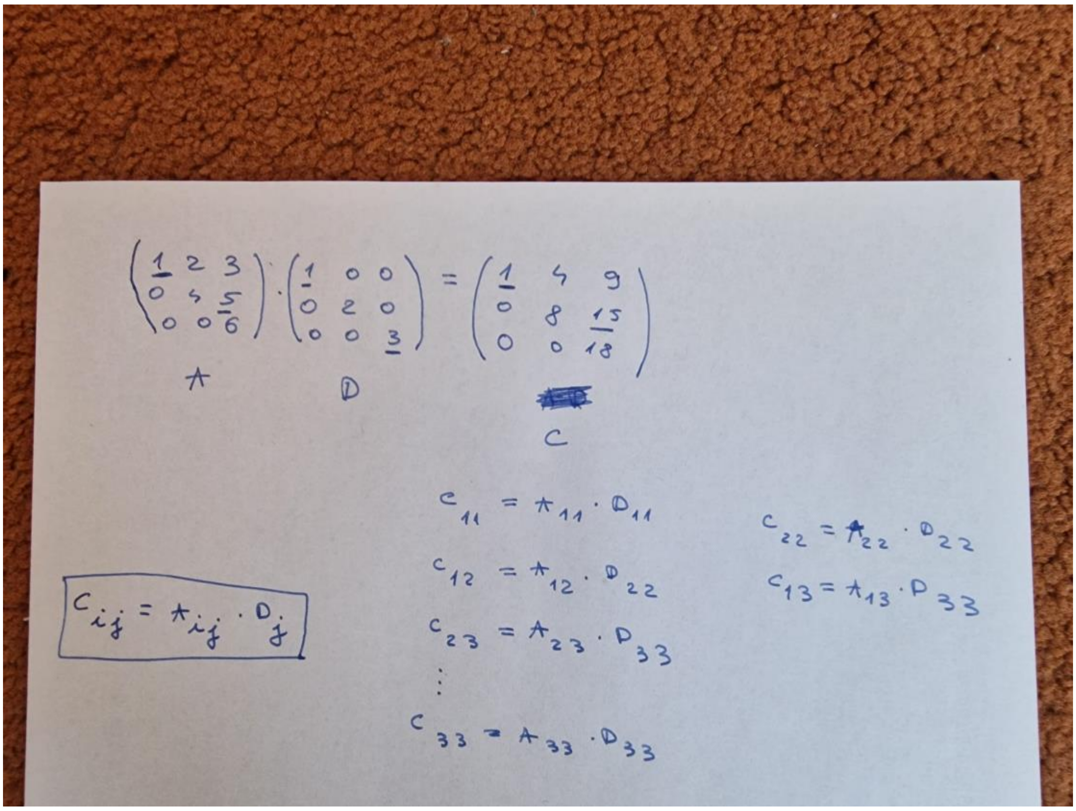
tema_3.m

```
n = input('introduceti dimensiunea matricelor: ');
%numarul meu de la teams este par, deci o sa am o matrice superior
%triunghiulara
for i = 1 : n
    for j = 1 : n
        if i <= j
            A(i, j) = input('A: ');
        else
            A(i, j) = 0;
        end
    end
end

disp('vectorul D: ');
D = zeros(n, 1);
for i = 1 : n
    D(i) = input('D: ');
end

for i = 1 : n
    for j = i : n
        C(i, j) = A(i, j) * D(j);
    end
end

disp('matricea A:');
disp(A);
disp('vectorul D:');
disp(D);
disp('matricea rezultat C:');
disp(C);
```

Stiu ca A este o matrice superior triunghiulara deci am citit doar ce era nevoie din matrice, iar restul am initializat cu 0. A este o matrice superior triunghiulara iar D un vector coloanal, deci C (matricea in care salvez rezultatul) este tot o matrice superior triunghiulara.

Mai intai am luat pe foaie un exemplu si am observat ca fiecare element din C respecta o “formula”, adica:

$$C_{ij} = A_{ij} * D_j$$



Stiu ca am zis ca D este un vector coloana, asa este in matlab dar eu pe foaie am luat o matrice ca sa respecte cerinta sa fie o matrice diagonala.

```
matricea A:
     1     2     3
     0     4     5
     0     0     6

vectorul D:
     1
     2
     3

matricea rezultat C:
     1     4     9
     0     8    15
     0     0    18
```

$$\sum_{i=1}^{n} \sum_{j=i}^{n} 1 =$$

$$= \sum_{i=1}^{n} \left(\sum_{j=1}^{n} 1 - \sum_{j=1}^{i-1} 1 \right) =$$

$$= \sum_{i=1}^{n} (n - i + 1) = \sum_{i=1}^{n} n - \sum_{i=1}^{n} i + \sum_{i=1}^{n} 1 =$$

$$= {}^{2/}n^2 - \frac{n(n+1)}{2} + {}^{2/}n =$$

$$= \frac{2n^2 - n^2 - n + 2n}{2} = \frac{n^2 + n}{2}$$

$$= \frac{n(n+1)}{2}.$$

Am calculat numarul de operatii pentru acest algoritm si este mai efficient decat inmultirea clasica unde complexitatea era $O(n^3)$.