

## Problema 6-E4 – Compresie imagine

Fotografia digitală este una dintre cele mai populare forme de reprezentare a informației vizuale. O fotografie este reprezentată ca o matrice de valori întregi și pozitive, din intervalul  $[0; 255]$ , ce sunt distribuite pe linii și coloane. Datorită necesarului de memorie semnificativ, imaginile sunt de cele mai multe ori stocate comprimat, prin eliminarea informației redundante.

### Cerință

Având la dispoziție o imagine,  $A[i][j]$ , de dimensiune  $m$  (număr de linii) x  $n$  (număr de coloane), să se realizeze compresia acesteia după cum urmează. Imaginea este transformată în domeniul frecvență

astfel:  $Af[u][v] = coef1(u) \cdot coef2(v) \cdot \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \cos\left[\frac{\pi \cdot u}{2 \cdot m}(2 \cdot i + 1)\right] \cos\left[\frac{\pi \cdot v}{2 \cdot n}(2 \cdot j + 1)\right] A[i][j]$ ,

$$\text{unde } u=0,\dots,m-1, \ v=0,\dots,n-1, \ coef1(u) = \begin{cases} \frac{1.0}{\sqrt{m}} & u = 0 \\ \sqrt{\frac{2.0}{m}} & u > 0 \end{cases}, \quad coef2(v) = \begin{cases} \frac{1.0}{\sqrt{n}} & v = 0 \\ \sqrt{\frac{2.0}{n}} & v > 0 \end{cases},$$

$\cos()$  reprezintă funcția cosinus ce primește la intrare un unghi reprezentat în radiani,  $\pi = 3.14$ .

$Af[i][j]$  reprezintă imaginea  $A[i][j]$  transformată în frecvență. Toate valorile din  $Af[u][v]$ , cu  $u \neq 0$  și  $v \neq 0$  sunt înlocuite cu 0. Imaginea  $A[i][j]$  este reconstituită în imaginea  $Ar[i][j]$  astfel:

$$Ar[i][j] = \sum_{u=0}^{m-1} \sum_{v=0}^{n-1} coef1(u) \cdot coef2(v) \cdot \cos\left[\frac{\pi \cdot u}{2 \cdot m}(2 \cdot i + 1)\right] \cos\left[\frac{\pi \cdot v}{2 \cdot n}(2 \cdot j + 1)\right] Af[u][v], \ i=0,\dots,m-1,$$

$j=0,\dots,n-1$ . Valorile astfel obținute sunt convertite în valori întregi prin rotunjire la cel mai mic întreg. Să se afișeze pe ecran valorile matricei  $Ar[i][j]$  astfel obținute.

### Date de intrare

Se vor citi de la tastatură (fluxul *stdin*) următoarele date:

- o valoare întreagă pentru numărul de linii  $m$ , urmată de caracterul *newline* (tasta *Enter*);
- o valoare întreagă pentru numărul de coloane  $n$ , urmată de caracterul *newline* (tasta *Enter*);
- valorile matricei  $A$ , introduse câte o valoare pe linie urmată de caracterul *newline* (tasta *Enter*), parcursă de la stânga la dreapta, și de sus în jos (parcursere pe linii și coloane).

### Date de ieșire

Programul va afișa pe ecran la ieșire, valorile matricei  $Ar[i][j]$ , câte o valoare pe linie urmată de caracterul *newline* (tasta *Enter*), parcurse de la stânga la dreapta, și de sus în jos (parcursere pe linii și coloane).

**ATENȚIE la respectarea cerinței problemei: afișarea rezultatelor trebuie făcută EXACT în modul în care a fost indicat! Cu alte cuvinte, pe stream-ul standard de ieșire nu se va afișa nimic în plus față de cerința problemei; ca urmare a evaluării automate, orice caracter suplimentar afișat, sau o afișare diferită de cea indicată, duc la un rezultat eronat și prin urmare la obținerea calificativului „Respins”.**

### Restricții și precizări

1. Dimensiunile matricei  $A$  sunt numere întregi, pozitive, mai mari strict decât 2 și mai mici strict decât 20. Valorile matricei sunt valori întregi, pozitive, în intervalul  $[0; 255]$ .
2. **Atenție:** În funcție de limbajul de programare ales, fișierul ce conține codul trebuie să aibă una din extensiile .c, .cpp, .java, sau .m. Editorul web **nu va adăuga automat** aceste extensii și lipsa lor duce la imposibilitatea de compilare a programului!

3. **Atenție:** Fișierul sursă trebuie numit de candidat sub forma: <nume>.<ext> unde nume este numele de familie al candidatului și extensia este cea aleasă conform punctului anterior. Atenție la restricțiile impuse de limbajul Java legate de numele clasei și numele fișierului!

## Exemplu

| Intrare  | Ieșire |
|--|--------|
| 3  | 2      |
| 4  | 2      |
| 2  | 2      |
| 2  | 2      |
| 1  | 2      |
| 3  | 2      |
| 4  | 2      |
| 3  | 2      |
| 2  | 2      |
| 4  | 2      |
| 2  | 2      |
| 4  | 2      |
| 4  | 2      |
| 4  | 2      |
| <p><b>Explicație:</b> <math>A = \begin{bmatrix} 2 &amp; 2 &amp; 1 &amp; 3 \\ 4 &amp; 3 &amp; 2 &amp; 4 \\ 2 &amp; 4 &amp; 4 &amp; 4 \end{bmatrix};</math></p> <p><math>Af = \begin{bmatrix} 10.103630 &amp; -0.811754 &amp; 0.857168 &amp; -1.209663 \\ -2.112864 &amp; 0.650515 &amp; 1.416999 &amp; -0.270934 \\ -0.828137 &amp; -0.908250 &amp; -1.222640 &amp; -0.066137 \end{bmatrix}</math></p> <p><i>Observație:</i> în urma aproximărilor realizate de compilatoare diferite, este posibil să obțineți mici diferențe la ultimele zecimale. Aceste diferențe nu afectează rezultatul final.</p> <p><math>Af = \begin{bmatrix} 10.103630 &amp; 0 &amp; 0 &amp; 0 \\ 0 &amp; 0 &amp; 0 &amp; 0 \\ 0 &amp; 0 &amp; 0 &amp; 0 \end{bmatrix}</math> (anularea valorilor de pe alte poziții decât 0,0);</p> <p><math>Ar = \begin{bmatrix} 2.916667 &amp; 2.916667 &amp; 2.916667 &amp; 2.916667 \\ 2.916667 &amp; 2.916667 &amp; 2.916667 &amp; 2.916667 \\ 2.916667 &amp; 2.916667 &amp; 2.916667 &amp; 2.916667 \end{bmatrix};</math></p> <p><math>Ar = \begin{bmatrix} 2 &amp; 2 &amp; 2 &amp; 2 \\ 2 &amp; 2 &amp; 2 &amp; 2 \\ 2 &amp; 2 &amp; 2 &amp; 2 \end{bmatrix}</math> (prin rotunjire la cel mai mic întreg).</p> |        |

**TimP de lucru: 120 de minute**