

## Indicații pentru Tema 6

- **Tema 6** va avea **9 probleme**, fiecare punctată cu note de la 1 la 10 astfel:
  - Indentare corespunzătoare = **1 puncte**
  - Utilizarea acoladelor, lizibilitatea codului = **1 puncte**
  - Utilizarea corespunzătoare a instrucțiunilor de tip **if, for, while, do...while** = **1 punct**
  - Declararea corespunzătoare de variabile = **1 punct**
  - Funcționalitatea corespunzătoare a programului (adică programul face ce trebuie să facă) = **3 puncte**
  - Utilizarea corespunzătoare a pointerilor = **2 puncte**
  - Citirea și afișarea corespunzătoare a variabilelor/ meniurilor = **1 punct**
- Tema trebuie încărcată pe Campus Virtual până cel târziu **MIERCURI, 27 noiembrie 2019, ora 23:59 (hard deadline)**.
- **TOATE PROBLEMELE SUNT OBLIGATORII!!**
- Vor fi încărcate 9 fișiere cu extensia **.c** denumite astfel: **p1.c, p2.c, p3.c, p4.c, p5.c, p6.c, p7.c, p8.c** și **p9.c** corespunzătoare rezolvărilor problemelor 1, 2, 3, 4, 5, 6, 7, 8, respectiv 9.
- **NU încărcăți proiecte CodeBlocks!!!** Numai fișierele cu extensia **.c** denumite corespunzător după cele menționate la subpunctul anterior.
- **IMPORTANT!!** Dacă nu compilează codul, problema este notată cu 0. Punctul din oficiu pe fiecare problema se acordă în cazul în care codul compilează și nu are nicio eroare de compilare!
- Tema este **OBLIGATORIE** și cei care nu o fac nu vor putea da Testul 3!!
- Copiatul este strict interzis! Dacă sunt găsite astfel de cazuri, toți studenții implicați (adică cei care copiază și cei de la care se copiază) vor fi exmatriculați.
- Încercați să rezolvați voi singuri problemele. Dacă aveți nevoie de ajutor, vă stau la dispoziție.
- Pentru orice fel de întrebări nu ezitați să mă contactați.

## Tema 6

### Problema 1

Să se citească de la tastatură un număr  $n \leq 50$  și un număr  $m \leq 100$  și să se declare o matrice de maxim 100 de linii și 50 de coloane cu numere întregi. Scrieți o funcție care poate citi de la tastatură o matrice având ca parametri matricea,  $n$  și  $m$ . Scrieți o altă funcție care poate afișa pe ecran matricea, fiecare linie din matrice pe câte un rând, iar elementele de pe aceeași linie despărțite printr-un spațiu având ca parametri matricea,  $n$  și  $m$ . Scrieți un program care utilizează cele două funcții create pentru a citi și afișa pe ecran o matrice. **Trebuie utilizați pointeri în cazul matricilor.**

### Problema 2

Să se citească de la tastatură o matrice pătratică cu  $n \leq 100$  linii și coloane cu numere întregi. Scrieți o funcție care numără câte elemente din matrice au ca vecini numai elemente numere pare și afișați cu ajutorul unei alte funcții un mesaj corespunzător pe ecran: "Matricea are k elemente care au ca vecini numai elemente pare.". Două numere sunt numite vecine dacă sunt adiacente pe verticală sau pe orizontală. **Trebuie utilizați pointeri în cazul matricilor.**

**Exemplu:** dacă se citesc  $n=4$  și matricea  $\begin{pmatrix} 0 & 2 & 0 & 1 \\ 3 & 4 & 0 & 0 \\ 6 & 1 & 0 & 1 \\ 4 & 8 & 0 & 1 \end{pmatrix}$  atunci se va afișa mesajul

"Matricea are 6 elemente care au ca vecini numai elemente numere pare". Indicație: Elementele respective apar îngroșate în matricea de mai sus.

### Problema 3

Să se citească de la tastatură un număr  $n \leq 10$  și să se declare o matrice pătratică cu maxim 10 linii și 10 coloane cu elemente întregi. Să se genereze într-un program o

matrice de forma:  $\begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 4 & 0 & 1 & 0 & 2 \\ 4 & 4 & 0 & 2 & 2 \\ 4 & 0 & 3 & 0 & 2 \\ 0 & 3 & 3 & 3 & 0 \end{pmatrix}$  pentru  $n=5$  folosind următoarele 8 funcții:

- 1) Prima funcție va inițializa toată matricea cu -1.
- 2) A doua funcție va pune zero-urile din matrice.
- 3) A treia funcție va seta doar elementele 1 din matrice.
- 4) A patra funcție va seta doar elementele 2 din matrice.
- 5) A cincea funcție va seta doar elementele 3 din matrice.
- 6) A șasea funcție va seta doar elementele 4 din matrice.
- 7) A șaptea funcție va genera matricea utilizând funcțiile 1,2,3,4,5 și 6.
- 8) A opta funcție va afișa matricea generată, fiecare linie din matrice pe câte un rând, iar elementele de pe aceeași linie despărțite printr-un spațiu

Indicație: Toate funcțiile vor avea ca parametri matricea și numărul n. **Trebuie utilizați pointeri în cazul matricilor utilizate în funcții sau în program. NU utilizați variabile globale.**

#### Problema 4

Să se citească de la tastatură un număr  $n \leq 10$  și o matrice pătratică cu n linii și n coloane cu elemente întregi. Scrieți o funcție care are ca parametri matricea și numărul n și modifică în memorie matricea astfel încât fiecare rând să fie oglindit (primul element să devină ultimul, și așa mai departe, vezi exemplul). Programul va utiliza funcția pentru modificarea matricii, după care va afișa pe ecran matricea, fiecare linie din matrice pe câte un rând, iar elementele de pe aceeași linie despărțite printr-un spațiu. **Trebuie utilizați pointeri în cazul matricii utilizate în funcții sau în program.**

**Exemplu:** dacă se citesc  $n=3$  și matricea  $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$  se va afișa matricea  $\begin{pmatrix} 3 & 2 & 1 \\ 6 & 5 & 4 \\ 9 & 8 & 7 \end{pmatrix}$ .

#### Problema 5

Să se citească de la tastatură un număr  $n \leq 20$  și o matrice pătratică cu elemente întregi. Scrieți o funcție care "îndoaie" matricea după diagonala principală, o altă funcție care "îndoaie" matricea după diagonala secundară, și o a treia funcție care "îndoaie" matricea după axa centrală verticală. Funcțiile vor avea ca argumente matricea și numărul n. Elementele unei jumătăți se vor aduna la cealaltă jumătate, iar apoi se vor înlocui cu 0 (vezi exemplul). Valorile de pe diagonale și axa centrală verticală se păstrează. Programul va apela cele trei funcții în ordinea îndoirii matricii: după diagonala principală, după diagonala secundară, după axa centrală verticală, și apoi va afișa pe ecran matricea rezultată, fiecare linie din matrice pe câte un rând, iar elementele de pe aceeași linie despărțite printr-un spațiu. **Trebuie utilizați pointeri în cazul matricilor utilizate în funcții sau în program.**

**Exemplu:** dacă se citesc  $n=4$  și matricea  $\begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 2 & 2 & 1 \\ 1 & 2 & 1 & 3 \\ 3 & 4 & 0 & 2 \end{pmatrix}$ , se va îndoi prima dată după

diagonala principală și va rezulta  $\begin{pmatrix} 1 & 2+3 & 3+1 & 4+3 \\ 0 & 2 & 2+2 & 1+4 \\ 0 & 0 & 1 & 3+0 \\ 0 & 0 & 0 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 5 & 4 & 7 \\ 0 & 2 & 4 & 5 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 2 \end{pmatrix}$ , apoi se

va îndoi după diagonala secundară  $\begin{pmatrix} 1+1 & 5+3 & 4+5 & 7 \\ 0 & 2+1 & 4 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 3 & 8 & 9 & 7 \\ 0 & 3 & 4 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$ , iar

apoi se va îndoi după axa centrală verticală  $\begin{pmatrix} 3+7 & 8+9 & 0 & 0 \\ 0 & 3+4 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 10 & 17 & 0 & 0 \\ 0 & 7 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$ .

## Problema 6

Să se citească un număr  $n \leq 10$  și să se declare o matrice pătratică cu maxim 10 linii și coloane cu elemente întregi. Scrieți o funcție care generează o matrice pătratică cu toate numerele de la 1 până la  $n \cdot n$ . După apelarea acelei funcții în programul principal, se consideră că un vierme pornește din colțul stânga sus al matricii și consumă elementele (le setează valoarea pe 0) unul câte unul, în spirală. Se va afișa pe ecran cu ajutorul unei alte funcții un mesaj de tipul "Elementul consumat este x, iar matricea rezultată este: ", mesaj urmat de afișarea matricii pe ecran, fiecare linie din matrice pe câte un rând, iar elementele de pe aceeași linie despărțite printr-un spațiu. Funcția aceasta trebuie apelată după consumarea fiecărui element. Programul se încheie când toate elementele sunt consumate. **Trebuie utilizați pointeri în cazul matricilor utilizate în funcții sau în program.**

**Exemplu:** dacă se citește  $n=3$ , se va genera matricea  $\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$ . Programul va avea

următorul rezultat:

Elementul consumat este 1, iar matricea rezultată este:  $\begin{pmatrix} 0 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$

Elementul consumat este 2, iar matricea rezultată este:  $\begin{pmatrix} 0 & 0 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$

Elementul consumat este 3, iar matricea rezultată este:  $\begin{pmatrix} 0 & 0 & 0 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}$

Elementul consumat este 6, iar matricea rezultată este:  $\begin{pmatrix} 0 & 0 & 0 \\ 4 & 5 & 0 \\ 7 & 8 & 9 \end{pmatrix}$

Elementul consumat este 9, iar matricea rezultată este:  $\begin{pmatrix} 0 & 0 & 0 \\ 4 & 5 & 0 \\ 7 & 8 & 0 \end{pmatrix}$

Elementul consumat este 8, iar matricea rezultată este:  $\begin{pmatrix} 0 & 0 & 0 \\ 4 & 5 & 0 \\ 7 & 0 & 0 \end{pmatrix}$

Elementul consumat este 7, iar matricea rezultată este:  $\begin{pmatrix} 0 & 0 & 0 \\ 4 & 5 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

Elementul consumat este 4, iar matricea rezultată este:  $\begin{pmatrix} 0 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

Elementul consumat este 5, iar matricea rezultată este:  $\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

### Problema 7

Să se citească de la tastatură  $n \leq 10$ ,  $m \leq 20$  și o matrice de  $m$  linii și  $n$  coloane, cu elemente reale. Scrieți o funcție care să calculeze suma elementelor din interiorul matricii și o altă funcție care să calculeze media aritmetică a elementelor de pe marginea matricii. Să se apeleze din programul principal cele două funcții și să se afișeze rezultatul returnat de acestea cu o precizie de 2 zecimale exacte. **Trebuie utilizați pointeri în cazul matricilor utilizate în funcții sau în program.**

**Exemplu:** dacă se citesc  $n=3$ ,  $m=4$  și matricea  $\begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{pmatrix}$  atunci suma va fi

$6 + 7 = 13$ , iar media aritmetică  $\frac{1+2+3+4+8+12+11+10+9+5}{10} = 6.5$  și se va afișa pe ecran

"Suma este 13.00"

"Media aritmetică este 6.50"

### Problema 8

Scrieți o funcție care primește ca parametru un număr întreg fără semn și returnează acel număr schimbat astfel: dacă numărul este par, se vor seta toți biții, mai puțin ultimii 8 (LSB), iar dacă numărul este impar, acesta va fi nemodificat. Se citește de la tastatură un vector cu  $n \leq 20$  numere întregi fără semn. După citirea și memorarea tuturor elementelor vectorului, se va prelucra fiecare element folosind funcția descrisă anterior și apoi va fi afișată valoarea fiecărui element (adică valoare modificată sau nemodificată în funcție de paritatea elementului). **Trebuie utilizați pointeri și operații pe biți.** **Exemplu:** dacă se citesc numerele 12 și 1023 ( $n=2$ ), vor fi afișate valorile 4 294 967 052 și 1023.

### Problema 9

Într-o variabilă de tip `int`, care ocupă 4 octeți, se pot memora 32 de biți, câte o valoare de 0 sau 1 pentru fiecare bit al variabilei. **Folosind operatori pe biți**, scrieți o funcție care memorează 0 sau 1 într-un anumit bit al unei variabile de tip întreg fără semn, bitul fiind specificat prin indicele lui, indicele fiind un număr între 0 și 31. Utilizând

din nou operațiile pe biți scrieți o altă funcție care returnează valoarea memorată într-un anume bit, bitul fiind specificat prin indicele lui, indicele fiind un număr între 0 și 31. Declarați apoi un vector de tip întreg fără semn de dimensiune 10. În acest vector se pot memora până la  $32 \cdot 10 = 320$  de biți. Citiți  $n \leq 320$  biți de la tastatură pe care îi memorați în vector, folosind funcțiile anterior declarate. Tot folosind funcțiile descrise anterior, citiți valorile din vector și afișați-le (afișați biții, nu valoarea în baza 10, deoarece a doua funcție menționată anterior poate returna valoarea memorată într-un anume bit, adică un 0 sau un 1). **Exemplu:** Dacă citim 0000 0000 0000 0000 0000 0000 0000 1100 0000 0001 ( $n=40$  biți), primul element din vector,  $v[0]$ , va avea memorați primii 32 de biți, iar următorul element din vector,  $v[1]$ , va avea memorați următorii 8 biți citiți, restul de 24 de biți din totalul de 32 de biți din reprezentarea lui  $v[1]$  fiind considerați 0. Pe ecran vor fi afișați 00000000 00000000 00000000 00001100 pe un rând, corespunzător lui  $v[0]$  și 00000001 00000000 00000000 00000000 pe următorul rând, corespunzător lui  $v[1]$ . **Indicație:** Puteți să afișați grupuri de câte 8 biți consecutivi separate printr-un spațiu, așa cum sunt afișați în exemplul de mai sus, pentru o citire mai ușoară a celor 32 de biți.