

Indicații pentru Tema 4

- **Tema 4** va avea **7 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, 13 noiembrie 2019, ora 23:59 (hard deadline)**.
- **TOATE PROBLEMELE SUNT OBLIGATORII!!**
- Vor fi încărcate 7 fișiere cu extensia **.c** denumite astfel: **p1.c, p2.c, p3.c, p4.c, p5.c, p6.c** și **p7.c** corespunzătoare rezolvărilor problemelor 1, 2, 3, 4, 5, 6, respectiv 7.
- **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 2!!
- 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 4

Problema 1

Să se citească de la tastatură un șir de numere întregi fără semn, încheiat cu cifra 0. Toate valorile introduse vor fi mai mici sau egale cu 200. Scrieți un program care determină dacă numărul total de biți de 1 care apar în reprezentările binare ale numerelor din șirul citit este par sau impar. Dacă numărul total de biți de 1 este par, respectiv impar, se va afișa pe ecran textul "par", respectiv "impar". **Trebuie folosite operațiile pe biți.** *Exemplu:* dacă se citesc numerele 7, 100, 13, 99 și 0. Se va afișa "impar".

Problema 2

Se introduc de la tastatură 8 numere întregi fără semn, mai mici sau egale cu 255 și se memorează într-un vector. Din fiecare din cele 8 numere introduse se extrage al treilea cel mai puțin semnificativ bit (LSB). Se obțin astfel 8 biți care se așează unul lângă altul, în ordinea în care au fost introduse numerele din care au fost extrași. Primul bit extras este introdus ca MSB, iar ultimul bit extras va fi introdus ca LSB. Scrieți un program care afișează valoarea numărului care se obține prin alăturarea celor 8 biți. **Trebuie folosite operațiile pe biți.** *Exemplu:* Se introduc numerele 10, 76, 45, 89, 6, 199, 231 și 33. Reprezentările lor pe biți sunt 00001010, 01001100, 00101101, 01011001, 00000110, 11000111, 11100111 și 00100001. Biții evidențiați sunt alăturați exact în ordinea în care au fost introduse numerele de la tastatură: 01101110 și programul nostru va afișa pe ecran valoarea acestei reprezentări binare, și anume, 110. **Indicație:** pentru că numerele sunt mai mici sau egale cu 255, le putem memora pe un tip care are numai 8 biți ca spațiu de stocare.

Problema 3

Să se citească de la tastatură un șir de numere întregi într-un vector cu maxim 200 de elemente. Să se scrie un program utilizând pointeri care creează doi vectori noi, unul cu elementele impare, iar celălalt cu elementele pare din vectorul citit de la tastatură. În final, va fi creat încă un vector care va avea elementele intercalate: unul impar, unul par, elemente preluate din cei doi vectori anterior creați. Programul va afișa vectorul inițial, cei doi vectori cu elemente impare și elemente pare și vectorul cu elementele impare și pare intercalate unul sub celălalt. **Trebuie folosiți pointeri!** *Exemplu:* Dacă avem vectorul $v=[8,7,9,0,2,4,1,5,3]$, acesta va fi împărțit în doi vectori, unul cu numerele impare $[7,9,1,5,3]$ și unul cu numerele pare $[8,0,2,4]$. Vectorul final va fi constituit din elementele intercalate din cei doi vectori obținuți anterior $[7,8,9,0,1,2,5,4,3]$.

Problema 4

Se citește de la tastatură un vector cu $n \leq 20$ numere întregi fără semn. După citirea și memorarea tuturor elementelor vectorului. Scrieți un program care ia fiecare element din vector și tipărește 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. **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 5

Se dă un șir format din $n \leq 50$ valori reale. Să se memoreze într-un vector cele n valori citite de la tastatură. După citirea și memorarea tuturor numerelor în vector, scrieți un program care afișează un nou vector astfel: dacă avem două elemente consecutive a și b , iar diferența dintre a și b este strict mai mică decât 10, programul va insera între cele două elemente maximul lor în valoare absolută și va afișa apoi vectorul creat. **Trebuie folosiți pointeri.** *Exemplu:* pentru $n=5$ și $\text{vect} = [-20, -5, 15, -13, 14]$, programul va afișa un nou vector $\text{vect} = [-20, 20, -5, 15, 15, -13, 14, 14]$. Pentru primele două numere consecutive, $-20 - (-5) = -15$ și este strict mai mic decât 10, ca urmare vom pune maximul lor în valoare absolută, adică pe 20, între cele două numere. Între 15 și -13 nu punem nimic deoarece $15 - (-13) = 28$ care nu este strict mai mic decât 10.

Problema 6

Se citesc de la tastatură doi vectori a și b , neordonați, cu $n \leq 50$, respectiv $m \leq 50$ numere întregi distincte. După citirea și memorarea tuturor numerelor în vectori, scrieți un program care verifică dacă acești vectori conțin aceleași numere (fără a face ordonarea vectorilor!) și afișați un mesaj corespunzător rezultatului verificării. **Vor fi utilizați pointeri pentru cei doi vectori.** *Exemplu:* se citesc vectorii $a = [1, 0, 4]$ și $b = [4, 1, 0]$, $n = m = 3$. Programul poate afișa un mesaj în felul următor "Vectorii conțin aceleași elemente."

Problema 7

Fie a un vector cu n elemente și b un vector cu m elemente, amândoi ordonați crescător. Să se construiască un program care să afișeze un nou vector c care să conțină atât elementele vectorului a , cât și elementele vectorului b , în ordine crescătoare. **Trebuie folosiți pointeri.** *Exemplu:* $a = [0, 1, 4, 10]$ și $b = [2, 3, 5, 7, 8]$, programul scris va afișa un vector $c = [0, 1, 2, 3, 4, 5, 7, 8, 10]$.