FIȘĂ DE LUCRU · CLASA A X-A · FIȘA 9 · ACOPERĂ LECȚIILE X.12–X.13

struct și recapitulare BAC de clasa a X-a

Durată: 120 min Nivel: mate-info

CE VEI ȘTI DUPĂ LECȚIE

- Declari structuri, accesezi corect câmpurile și sortezi vectori de structuri.
- Rezolvi grile de BAC care amestecă struct, matrice și șiruri.
- Rezolvi o problemă III.3 într-o parcurgere, cu memorie constantă.

Ce trebuie să știi (o pagină)

Regula	Exemplu	Observație
declararea tipului se termină cu `;`	`struct punct { int x, y; };`	nu creează nicio variabilă
variabile, vectori, inițializare	`punct a = {1, 2}, v[101];`	câmpurile în ordinea declarării
câmpul unei **variabile**	`a.x`, `v[i].y`, `p.nastere.an`	niciodată `punct.x`
atribuire întreagă	`b = a;` (copiază și tablourile din interior)	`swap(v[i], v[j])` mută toată fișa
comparare	`a.x == b.x && a.y == b.y`	`a == b` **nu** compilează
șir într-un câmp	`strcpy(e.nume, "Ana");`, `strcmp(a.nume, b.nume)`	`e.nume = "Ana"` nu compilează
sortare după un câmp	`bool inainte(elev a, elev b) { return a.medie > b.medie; }`	`sort(v + 1, v + n + 1, inainte);`
nume de tip	`dataCal`, nu `data`	`data` se bate cu `std::data`

ATENȚIE — GREȘELI FRECVENTE

La III.3 („fișierul conține cel mult 10^6 numere…", „eficient") se punctează separat **descrierea metodei** (în cuvinte, cu justificarea eficienței) și programul. De obicei ajung o singură parcurgere și câteva variabile, **fără vector**.

Nivelul 1 · Prezice

★ citește și prezice

P1. Cu declarațiile de mai jos, care instrucțiuni se compilează? Pentru cele greșite, spune de ce.

```
struct dataCal { int zi, luna, an; };
struct elev { char nume[31]; double medie; dataCal nas; };
elev e, f, v[10];
```

a) `e.nume[0] = 'A';`	**b)** `elev.medie = 9;`	**c)** `bool b = v[2].medie > v[1].medie;`	**d)** `bool b = e == f;`
e) `e = v[3];`	**f)** `v.medie[2] = 7;`	**g)** `int r = e.nas.an % 4;`	**h)** `int r = e.nas % 4;`

P2. Ce afișează?

```
struct punct { int x, y; };
punct a = {1, 2}, b = a;  b.x = 5;
punct v[3] = {a, b, {a.y, b.x}};
swap(v[0], v[2]);
for (int i = 0; i < 3; i++) cout << v[i].x << ',' << v[i].y << ' ';
cout << a.x;
```

P3. Magazinul. Ce afișează? Ce s-ar afișa dacă `>=` devine `>`?

```
struct produs { int cod, pret, stoc; };
produs p[5] = {{101, 20, 3}, {102, 45, 0}, {103, 15, 10}, {104, 45, 2}, {105, 8, 0}};
int t = 0, k = 0;
for (int i = 0; i < 5; i++) {
    if (p[i].stoc > 0) t += p[i].pret * p[i].stoc;
    if (p[i].pret >= p[k].pret) k = i; }
cout << t << ' ' << p[k].cod;
```

P4. Aici `elev` are doar câmpurile `nume` și `nas`. Ce afișează? Ce reprezintă numărul calculat și de ce compararea lui funcționează?

```
elev e[3] = {{"Ana", {14, 7, 2010}}, {"Dan", {3, 11, 2009}}, {"Eva", {28, 2, 2010}}};
int k = 0;
for (int i = 1; i < 3; i++)
    if (e[i].nas.an * 10000 + e[i].nas.luna * 100 + e[i].nas.zi <
        e[k].nas.an * 10000 + e[k].nas.luna * 100 + e[k].nas.zi) k = i;
cout << e[k].nume << ' ' << e[2].nas.an * 10000 + e[2].nas.luna * 100 + e[2].nas.zi;
```

P5. BAC. Variabila `t` memorează ora la care pleacă un tren (`t.h` între 0 și 23, `t.m` între 0 și 59). Care expresii au ca valoare numărul de minute rămase până la miezul nopții? (Pot fi mai multe.) Calculează-le pentru `t = {21, 35}`.

a. `24 * 60 - t.h * 60 - t.m` b) `(24 - t.h) * 60 + t.m` c) `(23 - t.h) * 60 + 60 - t.m` d) `24 * 60 - t.h * 60 + t.m`

P6. BAC. Tabloul `a` are 4 linii și 4 coloane, numerotate de la 1, cu `a[i][j] = (i + j) % 3`. Câte elemente **nule** se află strict **sub** diagonala principală?

a. 1 b) 2 c) 3 d) 4

P7. Ce afișează?

```
struct cuv { char s[10]; int l; };
cuv w[3] = {{"bac", 0}, {"info", 0}, {"x", 0}};
int s = 0;
for (int i = 0; i < 3; i++) { w[i].l = strlen(w[i].s); s += w[i].l; }
cout << s << ' ' << w[1].s[w[1].l - 1] << ' ' << w[2].s[0] - w[0].s[0];
```

P8. BAC, pseudocod.

```
citește n (număr natural nenul)
k ← 0; ant ← -1
┌pentru i ← 1,n execută
│ citește x (număr natural)
│┌dacă x > ant atunci k ← k + 1
│└■
│ ant ← x
└■
scrie k
```

a) Ce se afișează pentru `n = 7` și `3 5 5 2 8 9 1`? **b)** Pentru `n = 5`, dă un șir de numere din `[1, 9]` pentru care se afișează `5` și unul pentru care se afișează `1`. **c)** Spune în cuvinte ce numără `k`.

Nivelul 2 · Găsește și repară

★★ găsește și repară

P9. Programul are **patru** greșeli și nu se compilează. Găsește-le și repară-le, astfel încât să afișeze `diferiti` (al doilea elev e o copie a primului, cu media 10).

```cpp
struct elev { char nume[31]; double medie; }
int main() {
    elev e1, e2;
    e1.nume = "Ana"; e1.medie = 9.5;
    elev.medie = 10;
    if (e1 == e2) cout << "egali"; else cout << "diferiti";
}
```

P10. Clasamentul descrescător după medie al elevilor `Ana 8.5, Dan 9.7, Eva 7.2, Ion 9.1` iese `Ana 9.7, Dan 9.1, Eva 8.5, Ion 7.2`. Ce a interschimbat elevul? Repară.

```cpp
for (int i = 0; i < n - 1; i++)
    for (int j = i + 1; j < n; j++)
        if (v[j].medie > v[i].medie) swap(v[i].medie, v[j].medie);
```

P11. *Puzzle cu rânduri amestecate.* `produse.in` conține `n`, apoi `n` rânduri `nume pret stoc`. Afișează numele celui mai ieftin produs **aflat pe stoc** (primul, la egalitate) sau `nu exista`. Un rând este în plus.

```
A)  if (k == -1) cout << "nu exista"; else cout << p[k].nume;
B)      fin >> p[i].nume >> p[i].pret >> p[i].stoc;
C)  ifstream fin("produse.in");
D)  for (int i = 1; i <= n; i++) {
E)      if (p[i].stoc > 0 && (k == -1 || p[i].pret < p[k].pret)) k = i;
F)  fin >> n;  int k = -1;
G)      if (p[i].stoc > 0 && p[i].pret < p[k].pret) k = i;
H)  }
```

P12. Completează funcția ca `sort` să ordoneze elevii **descrescător după medie**, iar la medii egale **alfabetic**. Test: `Mara 9.5, Alex 8, Bogdan 9.5, Cara 10, Ana 8` → `Cara Bogdan Mara Alex Ana`.

```
bool inainte(elev a, elev b)
{
    if (________________) return _________________;
    return ____________________________;
}
```

P13. III.3: lungimea maximă a unei secvențe de numere **egale** consecutive din `bac.txt`. Pentru `4 4 4 1 1` programul afișează `2`, iar pentru `5` afișează `0`. Ce numără de fapt? Repară două valori.

```
int ant, x, lung = 0, lmax = 0;
fin >> ant;
while (fin >> x) {
    if (x == ant) lung++; else lung = 0;
    if (lung > lmax) lmax = lung;
    ant = x;
}
cout << lmax;
```

P14. Ce greșeală a făcut elevul? Ca să afle dacă data `d1` este înaintea datei `d2`, scrie `d1.an < d2.an || d1.luna < d2.luna || d1.zi < d2.zi`. Arată că expresia spune că `5.3.2026` este înaintea lui `10.1.2025`. Găsește două date pentru care expresia este adevărată **în ambele sensuri**. Scrie condiția corectă.

Nivelul 3 · Scrie cod

★★ scrie cod

Folosește `struct` acolo unde datele descriu același obiect. Verifică **toate** testele.

P15. Bursa de merit. Se citesc `n` elevi, fiecare cu `nume n1 n2 n3`. Afișează elevul cu media cea mai mare (primul, la egalitate) și media, cu două zecimale.

intrare	ieșire
`3` / `Ana 9 10 8` / `Dan 10 10 9` / `Eva 7 8 9`	`Dan 9.67`
`2` / `Ion 9 9 9` / `Ema 10 8 9`	`Ion 9.00`
`1` / `Max 5 6 6`	`Max 5.67`

P16. Parcarea. Parcarea e un dreptunghi cu laturile paralele cu axele, dat prin colțul stânga-sus și colțul dreapta-jos (`struct dreptunghi { punct ss, dj; };`). Se dau `n` mașini (puncte). Câte sunt în parcare (inclusiv pe margine)?

intrare	ieșire
`0 5 4 0` / `4` / `1 1` / `4 5` / `5 2` / `-1 3`	`2`
`-2 2 2 -2` / `3` / `0 0` / `2 -2` / `3 0`	`2`
`1 3 2 1` / `1` / `0 0`	`0`

P17. Clasamentul ligii. Se citesc `n` echipe, fiecare cu `nume victorii egaluri`. Punctaj = 3 · victorii + egaluri. Afișează clasamentul (descrescător după punctaj, la egalitate alfabetic), câte o echipă pe rând, cu punctajul.

intrare	ieșire
`4` / `Rapid 5 2` / `Dinamo 4 5` / `Poli 6 0` / `Arges 3 8`	`Poli 18` / `Arges 17` / `Dinamo 17` / `Rapid 17`
`2` / `B 0 0` / `A 0 0`	`A 0` / `B 0`

P18. Următorul tren. Se citește ora curentă `h m`, apoi `n` ore de plecare (în orice ordine). Afișează, în formatul `hh:mm`, primul tren care pleacă la ora curentă sau după ea, în aceeași zi; dacă nu există, `NU`.

intrare	ieșire
`14 20` / `4` / `9 15` / `14 35` / `22 5` / `14 20`	`14:20`
`23 50` / `2` / `8 0` / `12 30`	`NU`
`0 0` / `3` / `7 5` / `6 45` / `23 59`	`06:45`

P19. III.3: perechea vecină. `bac.txt` conține cel puțin 2 și cel mult 10^6 numere naturale $\leq 10^9$. Afișează cea mai mare sumă a două numere **vecine** din fișier și de câte ori apare. Scrie întâi descrierea metodei (3–4 rânduri), apoi programul.

`bac.txt`	ieșire
`3 9 1 8 2 7 3`	`12 1`
`5 5 5 5`	`10 3`
`1 2`	`3 1`

P20. Agenda telefonului. Se citesc `n` contacte (`nume telefon`, ambele fără spații), apoi un prefix. Afișează alfabetic contactele al căror număr începe cu prefixul sau `niciunul`.

intrare	ieșire
`4` / `Maria 0721111222` / `Andrei 0745123456` / `Bianca 0721999000` / `Cosmin 0356123123` / `072`	`Bianca 0721999000` / `Maria 0721111222`
`2` / `Ion 0740` / `Ana 0750` / `03`	`niciunul`

Nivelul 4 · Pune mintea la contribuție

★★★ pune mintea la contribuție

P21. Recunoaște tiparul. Pentru fiecare enunț scurt, spune ce unealtă folosești și ce complexitate obții (nu scrie codul): **a)** `bac.txt` are 10^6 numere din `[0, 100]`; afișează valoarea care apare cel mai des. **b)** Se dau `n` numere și `q` întrebări „suma de la poziția `x` la poziția `y`". **c)** `n ≤ 100` elevi cu nume și trei note; afișează-i descrescător după medie. **d)** Două fișiere cu câte 10^6 numere crescătoare; afișează valorile comune. **e)** Tablou `n × n`; înlocuiește cu 0 elementele de sub diagonala secundară. **f)** Cel mai lung cuvânt dintr-un text.

P22. III.3: cele mai mici două. `bac.txt` conține cel mult 10^6 numere naturale, cu cel puțin două valori diferite. Afișează cele mai mici două valori **distincte**, crescător, într-o singură parcurgere și cu

memorie constantă. Teste: `7 3 9 3 5 12` → `3 5`; `4 4 4 2` → `2 4`; `1 100` → `1 100`. De ce e nevoie de condiția `x > m1` pe ramura a doua?

P23. Contraexemplul. Un elev descrie așa metoda pentru „există în `bac.txt` două numere egale?": „citesc numerele pe rând și compar fiecare număr cu cel citit înaintea lui". Dă un fișier cu 3 numere pe care metoda greșește. Pentru ce fel de fișiere metoda este corectă? Ce ai folosi, eficient, dacă numerele sunt ≤ 10^6?

P24. Ce se întâmplă dacă... Ce afișează prima secvență? Se compilează a doua? Ce îți spune asta despre tablourile din interiorul unei structuri?

```
struct V { int t[3]; };
V a = {{1, 2, 3}}, b;  b = a;  b.t[0] = 9;
cout << a.t[0] << ' ' << b.t[0];
int x[3], y[3] = {1, 2, 3};  x = y;   // a doua
```

P25. Mini-proiect: calendarul. Un eveniment are `zi luna an ora minut titlu`. Ca să le sortezi cronologic, construiește **o singură cheie numerică** din cele cinci câmpuri, astfel încât ordinea cheilor să fie ordinea în timp. Ce înmulțitori alegi pentru fiecare câmp și ce tip trebuie să aibă cheia? De ce `zi + luna + an` nu merge? Sortează: `12 5 2026 18 30 Concert`, `3 11 2025 9 0 Olimpiada`, `12 5 2026 9 15 Test`.

Probleme de lucru în clasă (pbinfo)

Le rezolvăm împreună în ședință, în ordinea de mai jos.

- [ ] `#922` **puncte** — ușoară
  https://www.pbinfo.ro/probleme/922/puncte
- [ ] `#4193` **bac-fractii** — ușoară
  https://www.pbinfo.ro/probleme/4193/bac-fractii
- [ ] `#2809` **crescator1** — medie
  https://www.pbinfo.ro/probleme/2809/crescator1

Temă pentru acasă (pbinfo)

Trimite soluțiile pe pbinfo și notează punctajul obținut. Bifează ce ai terminat.

- [ ] `#1013` **aniversari** — ușoară
  https://www.pbinfo.ro/probleme/1013/aniversari
- [ ] `#4192` **bac-angajati** — ușoară
  https://www.pbinfo.ro/probleme/4192/bac-angajati
- [ ] `#2834` **sir13** — medie
  https://www.pbinfo.ro/probleme/2834/sir13