FIȘĂ DE LUCRU · CLASA A X-A · FIȘA 1 · ACOPERĂ LECȚIILE X.1–X.2

Fișiere text și primii vectori

Durată: 90 min Nivel: toate profilurile

CE VEI ȘTI DUPĂ LECȚIE

- Citești corect dintr-un fișier, fie când știi câte valori sunt, fie până la sfârșit.
- Urmărești pe hârtie bucle pe vectori: indici simetrici, vecini, poziții circulare.
- Alegi singur când ai nevoie de vector și când ajunge prelucrarea „din mers”.

Ce trebuie să știi (o pagină)

Ce vrei	Cod	De reținut
deschizi fișierele	`ifstream fin("date.in");` `ofstream fout("date.out");`	`#include <fstream>`, numele exact din enunț
știi câte valori sunt	`fin >> n; for (i = 1; i <= n; i++) fin >> x;`	`for`
citești până la sfârșit	`while (fin >> x) { ... }`	citirea e chiar condiția
citești perechi până la sfârșit	`while (fin >> a >> b) { ... }`	se oprește dacă lipsește oricare
declari un vector	`int a[1001];` (global)	global = plin de 0, poate fi mare
citești / parcurgi	`for (i = 1; i <= n; i++) fin >> a[i];`	indici `1..n`
invers	`for (i = n; i >= 1; i--)`	
perechea simetrică a lui `i`	`a[n - i + 1]`	bucla până la `n / 2`
vecinul următor, circular	`a[i % n + 1]`	pentru `i = n` dă `a[1]`
construiești vectorul `y`	`y[++m] = x[i];`	la final `m` = câte elemente are `y`
prima / ultima poziție a maximului	`a[i] > mx` / `a[i] >= mx`	o singură literă diferă

ATENȚIE — GREȘELI FRECVENTE

Două greșeli care nu dau eroare de compilare, doar rezultate greșite: `while (!fin.eof())` (ultima valoare poate fi prelucrată de două ori) și `a[i + 1]` într-o buclă care merge până la `n` (citești o căsuță din afara datelor).

Nivelul 1 · Prezice

★ citește și prezice

P1. Fișierul `date.in` conține:

```
3
4 -2 7
5 1
```

Ce scrie programul în `date.out`?

```
int n, x, s = 0, r = 0;
fin >> n;
for (int i = 1; i <= n; i++) { fin >> x; s += x; }
while (fin >> x) r++;
fout << s << ' ' << r;
```

P2. Fișierul `perechi.in` conține `1 2 3 4 5`. Ce valori au `cnt`, `s` și `a` după buclă?

```
int a = 0, b = 0, s = 0, cnt = 0;
while (fin >> a >> b) { s += a * b; cnt++; }
```

P3. Pentru `x = (2, 5, 9, 4)` și `n = 4`, ce conține `y`? Calculează suma elementelor lui `y`. Încearcă și cu alt vector: ce observi? Explică **de ce**.

```
for (int i = 1; i <= n; i++) y[i] = x[i % n + 1] - x[i];
```

P4. Elevul a vrut să inverseze vectorul `a = (4, 9, 2, 7, 5, 1)`. Ce afișează programul? Ce trebuie schimbat?

```
for (int i = 1; i <= n; i++) swap(a[i], a[n - i + 1]);
for (int i = 1; i <= n; i++) cout << a[i] << ' ';
```

P5. Ce conține `a = (3, 1, 4, 1, 5)` după secvență? Spune **în cuvinte** ce reprezintă `a[i]` la final.

```
for (int i = 2; i <= n; i++) a[i] = a[i] + a[i - 1];
```

P6. Completează tabelul pentru `a[i] = i * 3 % 7`, apoi spune câte elemente au `a[i] > i`.

`i`	1	2	3	4	5	6	7
`a[i]`							

P7. Vectorul `a = (5, 9, 2, 9, 7)`. Ce poziție găsește bucla de mai jos? Dar dacă înlocuiești `>` cu `>=`?

```
int mx = a[1], poz = 1;
for (int i = 2; i <= n; i++)
    if (a[i] > mx) { mx = a[i]; poz = i; }
```

P8. Indice în indice. Fișierul `perm.in` conține `5` pe primul rând și `3 5 1 2 4` pe al doilea. Programul citește vectorul `a[1..n]`, apoi execută `for (i = 1; i <= n; i++) cout << a[a[i]] << ' ';`. Ce afișează? De ce nu iese niciodată din vector pentru acest fișier?

Nivelul 2 · Găsește și repară

★★ găsește și repară

P9. Media greșită. Fișierul `temp.in` conține `5 8 2`, urmat de un Enter (rând gol la final). Ce afișează programul? Media corectă este 5. Repară-l.

```
int x, s = 0, cnt = 0;
while (!fin.eof()) { fin >> x; s += x; cnt++; }
cout << s / cnt;
```

P10. Trei greșeli. Enunțul cere: „Citește toate numerele din `maxim.in` și scrie cel mai mare în `maxim.out`." Pentru fișierul `-5 -2 -9` programul de mai jos nu trece testul. Găsește **trei** greșeli.

```
#include <fstream>
using namespace std;
ifstream fin("maxim.in");
ofstream fout("rezultat.out");
int main() {
    int x, mx = 0;
    while (fin >> x) if (x > mx) mx = x;
    cout << mx;
}
```

P11. Urcări numărate greșit. Vectorul `int a[101]` este global, `n = 4`, `a = (3, 5, 4, -2)`. Secvența ar trebui să numere „urcările" (`a[i] < a[i + 1]`). Ce afișează și de ce? Pentru `a = (3, 5, 4, 2)` ar fi greșit rezultatul?

```
int cnt = 0;
for (int i = 1; i <= n; i++) if (a[i] < a[i + 1]) cnt++;
cout << cnt;
```

P12. Rânduri amestecate. Fișierul `note.in` are pe primul rând `n`, apoi `n` note. Programul scrie în `note.out` notele mai mari decât **ultima** notă. Pune rândurile în ordine; un rând este în plus. Ce scrie programul pentru `5` / `7 10 5 9 8`?

```
A)  for (int i = 1; i <= n; i++) fin >> a[i];
B)  ifstream fin("note.in"); ofstream fout("note.out");
C)  for (int i = 1; i < n; i++)
D)      if (a[i] > a[n]) fout << a[i] << ' ';
E)  fin >> n;
F)  while (fin >> x) n++;
G)  int a[101], n;
```

P13. Completează startul. Scrie ce trebuie pus în locul lui `___`, ca să meargă pentru orice `n ≥ 1`:

a. elementele de pe pozițiile multiple de 3, de la coadă spre început: `for (int i = ___; i >= 1; i -= 3)`;

b. elementele de pe pozițiile impare, de la coadă spre început: `for (int i = ___; i >= 1; i -= 2)`.

Verifică pe `a = (10, 20, 30, 40, 50, 60, 70)`, apoi și cu `80` adăugat la final.

P14. Ce greșeală a făcut elevul? Programul se compilează, dar uneori se blochează sau dă rezultate ciudate:

```
int n;
int a[n];
cin >> n;
for (int i = 1; i <= n; i++) cin >> a[i];
```

Nivelul 3 · Scrie cod

★★ scrie cod

Pentru fiecare program, verifică **toate** testele date înainte să spui că e gata. Unde e un fișier, testul arată conținutul lui.

P15. Pașii zilnici. `pasi.in` conține numărul de pași din fiecare zi (nu știi câte zile). Scrie în `pasi.out` câte zile au cel puțin 10 000 de pași și cea mai lungă serie de astfel de zile **consecutive**.

`pasi.in`	`pasi.out`
`12000 8000 10500 11000 10000 3000`	`4 3`
`500 900`	`0 0`
`10000`	`1 1`

P16. Playlistul. `playlist.in`: `n`, apoi durata în secunde a celor `n` melodii. În `playlist.out` scrie pe primul rând durata totală ca `minute:secunde` (secundele mereu cu două cifre), iar pe al doilea pozițiile melodiilor **strict mai lungi decât media** (sau `0` dacă nu există).

`playlist.in`	`playlist.out`
`4` / `200 185 240 215`	`14:00` / `3 4`
`3` / `61 59 60`	`3:00` / `1`
`1` / `45`	`0:45` / `0`

P17. Locul în clasament. Se citesc `n` punctaje. Pentru fiecare jucător, afișează locul lui: 1 + câți jucători au punctaj strict mai mare (egalii împart locul). Teste: `5` / `70 95 70 100 60` → `3 2 3 1 5`; `3` / `10 10 10` → `1 1 1`.

P18. Zilele de vârf. Se dau temperaturile din `n` zile. Afișează pozițiile zilelor (fără prima și ultima) mai calde decât **ambele** zile vecine, sau `niciunul`.

intrare	ieșire
`7` / `3 5 4 4 6 2 8`	`2 5`
`3` / `1 2 3`	`niciunul`
`5` / `1 3 1 3 1`	`2 4`

P19. Copiii în cerc. `n ≥ 3` copii stau în cerc și fiecare are un număr de bomboane. Pentru fiecare copil, afișează câte bomboane are împreună cu cei doi vecini (primul și ultimul sunt vecini). Teste: `5` / `1 2 3 4 5` → `8 6 9 12 10`; `3` / `7 0 1` → `8 8 8`.

P20. Bonul de la magazin. Fiecare rând din `bon.in` conține `cantitate preț` (nu știi câte rânduri sunt). Scrie în `bon.out` totalul de plată și numărul rândului cu cea mai mare valoare (`cantitate ·`

`preț`); la egalitate, primul.

`bon.in`	`bon.out`
`2 3` / `1 10` / `4 2`	`24 2`
`5 5`	`25 1`
`1 7` / `7 1`	`14 1`

Nivelul 4 · Pune mintea la contribuție

★★★ pune mintea la contribuție

P21. Problema inversă. După secvența de la P5, vectorul a devenit `(2, 5, 5, 9, 15)`. Ce vector era la început? Scrie secvența care îl reface **pe loc** și explică de ce trebuie parcurs de la coadă.

P22. Fără vector. Fișierul conține multe numere (cel puțin 3). Afișează **ultimele trei**, în ordinea din fișier, folosind doar câteva variabile simple. Test: `4 8 15 16 23 42` → `16 23 42`.

P23. Numărul lipsă. `lipsa.in` conține `n`, apoi `n − 1` numere distincte din `1..n`, în orice ordine: unul lipsește. Găsește-l într-o singură parcurgere, fără vector. Test: `6` / `3 1 6 2 5` → `4`. De ce funcționează? Ce tip de date alegi dacă `n` ajunge la 10^6?

P24. Contraexemplul. Un elev spune: „Un vector cu `n` par e palindrom dacă primul element este egal cu ultimul **și** suma primei jumătăți este egală cu suma celei de-a doua.” Găsește un vector cu 6 elemente care respectă ambele condiții, dar nu e palindrom.

P25. Rotirea. Ce face secvența de mai jos pentru `a = (1, 2, 3, 4, 5)` și `k = 2`? Dar pentru `k = 5` și `k = 7`? Explică de ce nu iese niciodată din vector, oricât de mare ar fi `k`.

```
for (int i = 1; i <= n; i++) b[(i - 1 + k) % n + 1] = a[i];
```

Probleme de lucru în clasă (pbinfo)

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

- [ ] **#1 sum** — ușoară
  https://www.pbinfo.ro/probleme/1/sum
- [ ] **#3391 sirmaxim** — ușoară
  https://www.pbinfo.ro/probleme/3391/sirmaxim
- [ ] **#493 constr** — ușoară
  https://www.pbinfo.ro/probleme/493/constr
- [ ] **#553 pozminmax** — ușoară
  https://www.pbinfo.ro/probleme/553/pozminmax

Temă pentru acasă (pbinfo)

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

- [ ] #2269 **maximpar** — ușoară
  https://www.pbinfo.ro/probleme/2269/maximpar
- [ ] #488 **afisare** — ușoară
  https://www.pbinfo.ro/probleme/488/afisare
- [ ] #492 **numarare3** — ușoară
  https://www.pbinfo.ro/probleme/492/numarare3
- [ ] #161 **inlocuire** — ușoară
  https://www.pbinfo.ro/probleme/161/inlocuire
