

**VIZAT,
INSPECTOR ȘCOLAR GENERAL ADJ,
PROF. BOGDAN BUZATU**

**SUBIECTELE PROBEI PRACTICE PENTRU
EXAMENUL DE ATESTAT PROFESIONAL LA INFORMATICĂ, 2019
PROGRAMARE-NEINTENSIV**

1. Scrieți un program care construiește și afișează pe ecran un șir de n numere (n citit de la tastatură) cu proprietatea că numerele din șir sunt în ordine crescătoare și au toate cifrele din reprezentarea binară egale cu 1. De exemplu, pentru $n=4$ se vor scrie valorile 1, 3, 7, 15.
2. Construiți un vector cu n numere întregi (n citit de la tastatură) format din primele n numere prime care au suma cifrelor multiplu de 2. De exemplu, pentru $n=4$ se va construi șirul 2, 11, 13, 17. Afișați pe ecran șirul obținut.
3. Scrieți un program care citește numerele din fișierul "Numere.in" (cel mult 15 de numere întregi, fiecare număr cu maxim 4 cifre) și calculează suma numerelor care sunt palindroame. Construiți fișierul „Numere.out” care conține suma obținută.

Exemplu

Numere.in	Numere.out
121 10 4412 717 252 101 3648	1191

4. Construiți un vector format din primele n numere ale șirului 2, 2, 2, 2, 3, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 4, 5 etc., unde n se citește de la tastatură ($n \leq 100$).
Exemplu: dacă n este 7 se va forma șirul 2, 2, 2, 2, 3, 3, 3.
5. Verificați dacă un șir de n numere reale citite de la tastatură are proprietatea că fiecare număr din șir are exact trei cifre la partea zecimală și afișați pe ecran un mesaj corespunzător ("Da" sau "Nu").
Numărul n se citește de la tastatură ($2 \leq n \leq 10$).
Exemplu:
Pentru $n=4$ și numerele 4.312 7313.245 252.212 231.277 se va afișa "Da".
pentru $n=4$ și numerele 4.31 713. 457 252.212 231.277 se va afișa "Nu".
6. Scrieți programul care citește un număr natural nenul p și afișează toate tripletele de numere naturale ce reprezintă laturile unor triunghiuri de perimetru p .
7. Scrieți programul care citește un număr natural nenul n și afișează numărul posibilităților de descompunere a numărului n în sumă de numere consecutive.
8. Scrieți programul care citește un număr natural nenul n cu cel mult 4 cifre și determină cel mai mic număr prim mai mare decât n .

9. Scrieți programul care citește un număr natural nenul n cu cel mult patru cifre și determină cel mai mare termen al șirului lui Fibonacci, mai mic sau cel mult egal cu n .
10. Se citește un număr întreg strict pozitiv cu cel mult noua cifre. Să se verifice dacă cifrele lui sunt în ordine strict crescătoare.
11. Se citesc trei numere întregi strict pozitive. Să se afișeze în ordine crescătoare. Să se verifice dacă ele formează sau nu o progresie geometrică.
12. Fișierul text **bac.txt** conține, pe o singură linie, cel mult **100** de numere naturale nenule cu cel mult **4** cifre fiecare, numerele fiind separate prin câte un spațiu. Scrieți un program **Pascal/C/C++** care citește de la tastatură un număr natural nenul n ($n \leq 999$) și numerele din fișierul **bac.txt** și care afișează pe ecran, separate prin câte un spațiu, toate numerele din fișier care sunt divizibile cu n . Dacă fișierul nu conține niciun astfel de număr, atunci se va afișa pe ecran mesajul **NU EXISTA**.

Exemplu: dacă fișierul **bac.txt** conține numerele: **3 100 40 70 25 5 80 6 3798**,
pentru $n=10$ atunci pe ecran se va afișa: **100 40 70 80**

13. Fișierul text **NR.TXT** conține pe o singură linie, separate prin câte un singur spațiu, cel mult **100** de numere **întregi**, fiecare număr având cel mult **4** cifre. Scrieți un program **Pascal/C/C++** care citește numerele din fișierul **NR.TXT** și afișează pe ecran, separate prin câte un spațiu, în ordine crescătoare, toate numerele **naturale nenule** din fișier. Dacă nu există astfel de numere se va afișa pe ecran mesajul **NU EXISTA**.

Exemplu: dacă fișierul **NR.TXT** conține numerele: **-3 -10 0 7 -5 7 51 -800 6 3798**,
atunci pe ecran se va afișa: **6 7 7 51 3798**

14. Fișierului text **NR.TXT** conține pe o singură linie, separate prin câte un singur spațiu, cel mult **100** de numere naturale, fiecare număr având cel mult **4** cifre. Scrieți un program **Pascal/C/C++** care citește toate numerele din fișierul **NR.TXT** și afișează pe ecran, separate prin câte un spațiu, în ordine crescătoare, toate numerele din fișier care au cel puțin **3** cifre. Dacă fișierul nu conține astfel de numere se va afișa pe ecran mesajul **NU EXISTA**

15. Fișierul text **NR.TXT** conține pe o singură linie, separate prin câte un singur spațiu, cel mult **100** de numere naturale, fiecare număr având cel mult **4** cifre. Scrieți un program **Pascal/C/C++** care citește numerele din fișierul **NR.TXT** și afișează pe ecran, separate prin câte un spațiu, în ordine descrescătoare, toate numerele din fișier care au cel mult **2** cifre. Dacă fișierul nu conține astfel de numere se va afișa pe ecran mesajul **NU EXISTA**.

16. Scrieți un program **Pascal/C/C++** care citește de la tastatură un număr natural n cu cel mult **8** cifre ($n \geq 10$) și care creează fișierul text **NR.TXT** ce conține numărul n și toate prefixele nenule ale acestuia, pe o singură linie, separate prin câte un spațiu, în ordine descrescătoare a valorii lor.

Exemplu: pentru $n=10305$ fișierul **NR.TXT** va conține numerele:

10305 1030 103 10 1

17. Scrieți un program C/C++ care citește de la tastatură două numere naturale nenule n și m ($2 \leq m \leq 10$, $2 \leq n \leq 10$) și care construiește în memorie și apoi afișează o matrice A cu n linii (numerotate de la 1 la n) și m coloane (numerotate de la 1 la m) cu proprietatea că fiecare element A_{ij} memorează cea mai mică dintre valorile indicilor i și j ($1 \leq i \leq n$, $1 \leq j \leq m$).

Matricea se va afișa pe ecran, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spațiu.

Exemplu: pentru $n=4$ și $m=5$ se va afișa matricea alăturată.

```
1 1 1 1 1
1 2 2 2 2
1 2 3 3 3
1 2 3 4 4
```

18. Scrieți un program Pascal/C/C++ care citește de la tastatură un număr natural n ($2 \leq n \leq 24$) și construiește în memorie o matrice cu n linii și n coloane ale cărei elemente vor primi valori după cum urmează:

- elementele aflate pe diagonala principală a matricei vor primi valoarea **0**
- elementele de pe prima coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **n**
- elementele de pe a doua coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **n-1**
- ...
- elementele de pe ultima coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **1**

Programul va afișa matricea astfel construită pe ecran, câte o linie a matricei pe câte o linie a ecranului, cu câte un spațiu între elementele fiecărei linii (ca în exemplu).

Exemplu: pentru $n=4$ se va afișa matricea alăturată.

```
0 3 2 1
4 0 2 1
4 3 0 1
4 3 2 0
```

20. Scrieți un program Pascal/C/C++ care citește de la tastatură un număr natural n ($2 \leq n \leq 24$) și construiește în memorie o matrice cu n linii și n coloane ale cărei elemente vor primi valori după cum urmează:

- elementele aflate pe diagonala secundară a matricei vor primi valoarea **0**
- elementele de pe prima linie, cu excepția celui aflat pe diagonala secundară vor primi valoarea **n**
- elementele de pe a doua linie, cu excepția celui aflat pe diagonala secundară vor primi valoarea **n-1**
- ...
- elementele de pe ultima linie, cu excepția celui aflat pe diagonala secundară vor primi valoarea **1**

Programul va afișa matricea astfel construită pe ecran, câte o linie a matricei pe câte o linie a ecranului, cu câte un spațiu între elementele fiecărei linii (ca în exemplu).

Exemplu: pentru $n=4$ se va afișa matricea alăturată.



4 4 4 0
3 3 0 3
2 0 2 2
0 1 1 1

21. Scrieți un program **Pascal/C/C++** care citește de la tastatură două numere naturale **n** și **m** ($2 \leq m \leq 10$, $2 \leq n \leq 10$) și care construiește în memorie și apoi afișează o matrice **A** cu **n** linii (numerotate de la **1** la **n**) și **m** coloane (numerotate de la **1** la **m**) cu proprietatea că fiecare element **A_{ij}** memorează cea mai mare dintre valorile indicilor **i** și **j** ($1 \leq i \leq n$, $1 \leq j \leq m$).

Matricea se va afișa pe ecran, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spațiu.

Exemplu: pentru **n=4** și **m=5** se va afișa matricea alăturată.

```
1 2 3 4 5
2 2 3 4 5
3 3 3 4 5
4 4 4 4 5
```

22. Scrieți un program **Pascal/C/C++** care citește de la tastatură două numere naturale **n** și **p** ($2 \leq n \leq 20$, $1 \leq p \leq 20$) și construiește în memorie un tablou bidimensional cu **n** linii și **p** coloane. Tabloul va fi construit astfel încât, parcurgând tabloul linie cu linie de sus în jos și fiecare linie de la stânga la dreapta, să se obțină șirul primelor **n*p pătrate perfecte impare**, ordonat strict crescător, ca în exemplu. Tabloul astfel construit va fi afișat pe ecran, fiecare linie a tabloului pe câte o linie a ecranului, cu câte un spațiu între elementele fiecărei linii.

Exemplu: pentru **n=2**, **p=3** se va afișa tabloul alăturat:

```
1 9 25
49 81 121
```

23. Scrieți un program **Pascal/C/C++** care citește de la tastatură un număr natural **n** ($2 < n < 16$), construiește în memorie și afișează pe ecran o matrice cu **n** linii și **n** coloane în care elementele de pe cele două diagonale sunt egale cu **0**, elementele care se află deasupra ambelor diagonale sunt egale cu **1**, elementele care se află sub ambele diagonale sunt egale cu **2**, iar restul elementelor sunt egale cu **3**.

Elementele matricei vor fi afișate pe ecran, câte o linie a matricei pe câte o linie a ecranului cu câte un spațiu între elementele fiecărei linii.

Exemplu: pentru **n=5** se va afișa matricea alăturată.

```
0 1 1 1 0
3 0 1 0 3
3 3 0 3 3
3 0 2 0 3
0 2 2 2 0
```

24. Să se afișeze toate perechile de numere palindromice, mai mici sau egale decât un număr natural n dat, $n \geq 2$. O pereche de numere se numește palindromică dacă al doilea număr al perechii este egal cu oglinditul primului număr.
25. 2. Să se genereze toate numerele prime de trei cifre, cu proprietatea că prefixele lor sunt de asemenea numere prime (prefixele se obțin din numărul inițial eliminând o cifră, apoi două cifre, șamd).
26. Dintre numerele naturale mai mici sau egale decât o valoare naturală n , dată, să se afișeze acelea care sunt divizibile cu "cifra lor de control" (cifra de control se obține calculând suma cifrelor, apoi suma cifrelor numărului obținut, șamd, până se obține o sumă formată dintr-o singură cifră).
27. Să se afișeze toate numerele de forma \overline{abcd} , cu proprietatea că suma inverselor cifrelor a, b, c, d este un număr subunitar, iar $a < b < c < d$.
28. Să se determine toate numerele de patru cifre care au cel puțin un zero și suma tuturor cifrelor este egală cu 14.
29. Se consideră un număr natural n cu maxim opt cifre și o cifră c . Să se afișeze numărul rezultat din n prin eliminarea tuturor aparițiilor cifrei c .
30. Se citește un șir de n numere naturale nenule. Să se afișeze numărul format din prima cifră a fiecărui număr citit, în ordinea inversă citirii lor.
31. Se citește de la tastatură un număr natural, nenul n ($n \leq 500$). Să se afișeze cel de-al n -lea termen al șirului: 1, 1, 2, 2, 1, 2, 3, 3, 3, 1, 2, 3, 4, 4, 4, 4, ... (grupa k este formată din 1, 2, ..., $k-1$ urmate de k scris de k ori).
32. Scrieți un program care afișează cea mai mică dintre puterile lui 2 care apare în descompunerea în factori primi a două numere naturale nenule date, x și y .
33. Să se afișeze cel mai mic și cel mai mare divizor prim al unui număr natural n , citit de la tastatură.
34. Să se scrie un program care determină cea mai mică cifră a unui număr natural x , citit de la tastatură, respectiv numărul de poziții pe care se află acesta în numărul x .
35. Se consideră un număr natural n , cu maxim opt cifre. Să se afișeze diferența dintre cea mai mare cifră și cea mai mică cifră a numărului dat.
36. Se citește un număr natural nenul cu maxim opt cifre. Să se formeze din n alte două numere x, y , astfel încât x va conține numai cifrele pare ale numărului dat, iar y doar pe cele impare ale acestuia, în ordinea apariției lor în n .

37. Să se scrie un program care determină dacă un număr natural nenul cu cel mult nouă cifre, citit de la tastatură este un număr "fierăstrău" (este format numai din cifre distincte a și b și are una din formele \overline{aba} , \overline{ababa} , $\overline{abababa}$, $\overline{ababababa}$, $a > b$).
38. Se citește de la tastatură un număr natural nenul n , $n \leq 100$. Să se afișeze al n -lea termen al șirului. 11, 22, 33, ..., 99, 111, 222,
39. Se citesc de la tastatură trei numere naturale nenule a , b și m de cel mult trei cifre fiecare, $m \leq 20$. Determinați și afișați primii m multiplii comuni ai numerelor a și b .
40. Scrieți un program care citește de la tastatură două numere naturale a și b , $1 \leq a \leq b \leq 32000$ și care verifică dacă aceste numere sunt termeni consecutivi în șirul lui Fibonacci, afișând un mesaj corespunzător.
18. Se citește de la tastatură un număr natural k , cu cel mult opt cifre. Să se determine și afișeze dacă există un număr natural n astfel încât $k=1 \cdot 2 \cdot 3 \cdot \dots \cdot n$ sau mesajul 'NU EXISTĂ', în caz contrar.
19. Se consideră un număr natural nenul n , cu cel mult opt cifre. Să se afișeze, dacă există un număr natural, mai mare decât n , și care are exact aceleași cifre ca și n , sau mesajul 'Nu există' în caz contrar.
20. Pentru un număr natural n cu cel mult opt cifre, să se afișeze un număr natural k astfel încât produsul $1 \cdot 2 \cdot 3 \cdot \dots \cdot k$ să fie cât mai apropiat ca valoare de n .
21. Se consideră un număr natural n cu cel mult nouă cifre. Scrieți un program care afișează cel mai mare număr natural ce se poate forma din toate cifrele numărului n .
22. Scrieți un program care citește de la tastatură un număr natural x , cu cel mult opt cifre și care verifică dacă acesta este superprim sau nu, afișându-se un mesaj corespunzător. Un număr este superprim dacă atât el, cât și toate prefixele lui sunt numere prime.
23. Realizați un program care înlocuiește fiecare apariție a unei cifre $c1$ într-un număr natural nenul n , dat, cu o cifră $c2$, și invers afișând valoarea obținută.
24. Să se scrie un program care pentru un număr natural nenul n , citit de la tastatură, afișează numărul format din cifrele lui n , luate o singură dată, în ordine descrescătoare.
25. Se citesc de la tastatură două numere naturale nenule n și k . Să se calculeze produsul cifrelor pare din ultimele k cifre ale numărului n , numerotarea cifrelor făcându-se de la dreapta la stânga începând cu poziția 0.
26. Se citește de la tastatură un număr natural nenul n și apoi n valori naturale. Să se calculeze și afișeze maximul dintre cifrele zecilor ale celor n valori citite.
27. Se citește de la tastatură un șir de numere naturale până la întâlnirea valorii 0, care nu face parte din șir. Să se afișeze cel mai mic număr pătrat perfect dintre cele citite, precum și numărul său de apariții.

28. Se citesc 3 numere naturale n , a și b . Afișați primele n puteri ale lui 2 din intervalul $[a, b]$. Dacă nu există cel puțin n puteri ale lui 2 în interval, atunci se vor afișa cele care există.
29. Se citesc numere naturale până când se citește numărul 0. Să se afișeze care dintre numerele citite are număr minim de divizori primi. Dacă există mai multe numere cu același număr minim de divizori se va afișa doar ultimul dintre ele.
30. Se citește un număr natural n cu cel puțin 3 cifre. Să se calculeze și să se afișeze numărul obținut din n prin eliminarea primei și a ultimei cifre.
31. Se citesc două numere naturale a și b . Să se determine câte cifre egale se află pe poziții indentice în cele două numere.
32. Fiind dat un vector v cu n elemente numere întregi, să se afișeze de câte ori găsim două elemente egale aflate pe poziții consecutive.
33. Să se afișeze poziția pe care apare primul element par, într-un vector cu n elemente numere întregi, $2 \leq n \leq 20$.
34. Se citește un vector a cu n elemente numere întregi. Ștergeți toate aparițiile primului element și afișați vectorul rămas.
35. Fiind dat un vector v cu n elemente numere întregi, $2 \leq n \leq 30$, se cere să se insereze între oricare două elemente suma lor, afișând vectorul rezultat.
36. Se dă un vector v cu n elemente numere naturale, $n \geq 2$. Să se afișeze toate perechile de elemente ale șirului cu proprietatea că cele două valori din pereche au aceeași sumă a cifrelor.
37. Fie un vector v cu n elemente numere întregi, $2 \leq n \leq 100$. Să se afișeze toate perechile de elemente consecutive din vector cu proprietatea că primul număr din pereche este egal cu suma cifrelor celui de-al doilea număr.
38. Se consideră un vector v cu n elemente numere naturale, $n \geq 2$. Să se afișeze numărul perechilor de elemente consecutive din v , cu proprietatea că al doilea număr al perechii reprezintă numărul de apariții ale unei cifre date k în pătratul primului număr.
39. Fie un vector v cu n elemente numere naturale, cu cel mult patru cifre fiecare. Să se afișeze cel mai mare divizor comun și cel mai mic multiplu comun, pentru oricare două elemente aflate pe poziții consecutive în vectorul dat.
40. Fiind dat un vector v cu n elemente numere întregi, $n \geq 2$, să se determine cele mai mici două elemente ale vectorului dat.
41. Se consideră un vector v cu n elemente numere întregi, $n \geq 2$. Să se determine cel mai mare element al șirului a căror sumă a cifrelor este un număr par, precum și numărul de apariții ale acestuia.
42. Fiind dat un vector v cu n elemente numere naturale, $n \geq 2$, să se afișeze cel mai mare palindrom al șirului, precum și numărul său de apariții.

43. Se consideră un vector v cu n elemente numere naturale nenule, $n \geq 3$. Să se afișeze cele mai mari trei elemente ale șirului care au cel mult k divizori, k citit de la tastatură. Dacă nu există trei astfel de elemente se va afișa un mesaj corespunzător.
44. Fiind dat un vector v cu n elemente numere întregi, $2 \leq n \leq 100$, să se construiască și afișeze un alt vector w , în care $w[i]$ reprezintă poziția valorii minime din șirul $v[i], v[i+1], \dots, v[n]$, $\forall i = \overline{1, n}$.
45. Se citește de la tastatură patru numere naturale nenule: n, k, a, b . Să se construiască o matrice pătratică cu n linii și n coloane astfel: pe linia k matricea va conține numai valoarea a , pe coloana k va conține numai valoarea b , și în rest, 0 . Să se afișeze matricea construită.
46. Se citește o matrice cu n linii și m coloane având elemente numere naturale. Să se înlocuiască fiecare element din matrice cu numărul său de divizori și să se afișeze matricea astfel formată.
47. Se citește o matrice a cu n linii și m coloane (numerotate începând cu 1) și elemente numere întregi. ($2 \leq n, m \leq 10$). Să se ordoneze crescător elementele pare de pe liniile cu număr par, și să se afișeze matricea astfel obținută.
48. Se citește o matricea cu $n * m$ elemente numere naturale. Afișați elementele care au proprietatea că sunt egale cu cel mai mare divizor comun al sumei elementelor de pe linia lor și al produsului elementelor de pe coloana lor.
49. Fișierul text 'f11_1.txt' conține pe prima linie un număr natural n , iar pe cea de-a doua linie n numere întregi separate prin spațiu. Să se determine suma elementelor pare și produsul elementelor impare din fișierul dat.
50. Se consideră fișierul text 'f11_2.txt', care conține un șir de cel mult 100 numere reale, scrise fiecare pe câte o linie a fișierului. Să se calculeze numărul valorilor din fișier care au partea întreagă un număr par, format din exact trei cifre.
51. Din fișierul text 'f11_3.txt' se citește un șir de cel mult 100 numere naturale, numerele fiind scrise pe același rând, separate prin câte un spațiu. Să se afișeze acele numere din fișier care sunt palindrom, și în plus sunt prime.
52. Se citește de la tastatură două numere naturale p și q , $0 < p < q \leq 100$. Să se afișeze în fișierul text 'f12_21.txt' acele numere din intervalul $[p, q]$ care au proprietatea că inversul pătratului lor este un număr prim.
53. Din fișierul text 'f12_2.in' se citește un șir de numere întregi. Numerele sunt scrise pe o singură linie a fișierului, fiind separate prin câte un spațiu. Să se scrie în fișierul text 'f12_2.out', pe câte o linie, suma cifrelor fiecărui număr citit.
54. Să se scrie într-un fișier text 'f12_3.txt' acele numere mai mici sau egale decât un număr natural n , citit de la tastatură, $2 \leq n \leq 10000$, care sunt formate din cifre identice. Numerele vor fi scrise pe aceeași linie a fișierului, separate prin câte un spațiu.
55. Fișierul text 'f12_4.txt' conține un șir cu cel mult 100000 de numere naturale cu exact 2 cifre fiecare. Să se afișeze în ordine crescătoare, numerele care apar exact o singură dată în fișierul dat.

56. Se dă fișierul text 'f13_2.in' care conține mai multe linii, pe fiecare linie fiind un șir de numere reale. Separate prin câte un spațiu. Să se scrie în fișierul text 'f13_2.out', pentru fiecare linie citită, separate prin câte un spațiu, cel mai mic și cel mai mare dintre valorile citite de pe aceea linie.
57. Se citește un șir de numere întregi, cu cel mult două cifre fiecare, din fișierul 'f13_3.txt'. Numerele sunt conținute pe același rând, cu spațiu între ele. Să se afișeze pe ecran, pe primul rând numerele pare, distincte, ordonate crescător, iar pe al doilea rând, numerele impare, distincte, ordonate descrescător, fără a folosi niciun algoritm de sortare.
58. Un fișier text conține cel mult 1000000 de cifre. Să se afișeze cel mai mare număr format din cifre distincte două câte două, din fișierul dat.
59. Se consideră un vector v cu n elemente aparținând mulțimii $\{1, 2, 3\}$, $n \geq 3$, citit din fișierul text „fisa14.in”. Folosind o singură parcurgere a vectorului și fără a folosi un alt vector, să se ordoneze crescător vectorul dat.
60. Se dau doi vectori a cu n ($1 \leq n \leq 50$) elemente numere întregi și b cu m ($1 \leq m \leq 50$) elemente numere întregi, ordonați crescător. Să se afișeze elementele comune și necomune din cei doi vectori, în ordine crescătoare, în fișierul text „f16_2.out”.
61. Se consideră două numere naturale nenule m și n și două mulțimi cu m , respectiv n elemente numere reale, memorate sub formă de vectori. Să se afișeze intersecția celor două mulțimi, în fișierul text “f16_3.out”.
62. Se dă un vector v cu n elemente numere reale, n număr natural, $n \geq 2$, având prima jumătate ordonată crescător și a doua jumătate ordonată descrescător. Folosind algoritmul de interclasare să se afișeze elementele vectorului în ordine descrescătoare, în fișierul text “f16_4.out”.
63. Se dă un vector v cu n elemente numere naturale ($2 \leq n \leq 100$). Numerele se citesc din fișierul text “f17_1.in”, care conține pe prima linie numărul n , iar pe următoarea linie n cele n elemente ale vectorului. Să se afișeze numărul format din cea mai mica cifră pară a fiecăruia dintre cele n numere date. Dacă numerele date nu conțin nicio cifră pară se va afișa un mesaj corespunzător.
64. Să se scrie un program care, pentru un număr natural nenul n și un vector v cu n elemente numere întregi, afișează pe câte o linie a unui fișier text suma primelor i elemente ale vectorului, $\forall i = \overline{1, n}$.
65. Se citește dintr-un fișier text un număr natural n , $n \geq 1$ și apoi cele n elemente numere reale distincte ale unui vector. Fără a ordona efectiv vectorul să se determine ce poziție ar ocupa ultimul element din vector, dacă acesta ar fi supus unui criteriu de ordonare.
66. Se dă un vector v cu n elemente numere naturale. Să se afișeze pentru fiecare pereche de elemente consecutive din v , cel mai mic număr cuprins în intervalul determinat de cele două valori, care este divizibil cu un număr natural k , citit de la tastatură, sau 0 dacă nu există un astfel de număr. Valorile cerute vor fi scrise în fișierul text ‘fisa18.txt’, fiecare pe câte o linie a fișierului.
67. Fișierul text 'fisa18.txt' conține pe prima linie un număr real pozitiv x , care are cel mult trei cifre la partea întreagă și cel mult cinci cifre la partea zecimală. Scrieți un program care afișează pe ecran, separate printr-un spațiu, două numere naturale, al căror raport este egal cu x și a căror diferență absolută este minimă.

68. Se dă fișierul text 'f19_2.in', care conține pe prima sa linie un număr natural nenul n , iar pe următoarele n linii câte un număr natural format din cel mult opt cifre. Să se afișeze într-un fișier text 'f19_2.out', pe câte o linie a fișierului, câte un număr din fișierul dat, urmat de caracterul "." și de cifrele care apar în scrierea acestuia, în ordine crescătoare, separate prin câte un spațiu.
69. Fișierul text 'f19_3.txt' conține pe prima linie un număr natural nenul n , $n \leq 100$, iar pe următoarea linie n numere naturale nenule, de maximum patru cifre, reprezentând elementele unui vector. Să se verifice dacă elementele vectorului dat reprezintă sau nu o permutare a mulțimii $\{1, 2, 3, \dots, n\}$, afișându-se un mesaj corespunzător.
70. Se dă un vector v cu n elemente numere naturale, cu cel mult șase cifre fiecare. Să se scrie într-un fișier text, pe câte o linie a fișierului, pentru fiecare număr din cele n date, cel mai apropiat număr de acesta, care începe și se termină cu aceeași cifră, mai mare decât numărul respectiv.
71. Se citesc 2 șiruri de caractere. Să se construiască un șir ce conține primele 3 caractere din fiecare șir citit. Ex: $a = \text{"Informatica"}$; $b = \text{"aplicata"}$; Se va construi: $c = \text{"Infapl"}$
72. Se citesc numele și nota de la informatică a doi elevi. Să se afișeze numele elevilor în ordinea descrescătoare a notei de la informatică. Dacă amândoi elevii au aceeași notă, se va afișa primul, cel care este alfabetic mai mic.
73. Se consideră un text cu cel mult 70 de caractere (litere mici ale alfabetului englez și spații), în care cuvintele sunt separate prin unul sau mai multe spații. Înaintea primului cuvânt și după ultimul cuvânt nu există spații. Scrieți un program care citește de la tastatură un text de tipul menționat mai sus și afișează pe ecran numărul de cuvinte în care apare litera a . Exemplu: pentru textul : "voi sustine examenul la informatica" se va afișa valoarea 3.
74. Se citește un șir de maxim 20 caractere ce nu conține caractere albe. Să se elimine toate consoanele și să se afișeze șirul.
75. Fișierul atestat.txt conține un text scris cu litere mari pe una sau mai multe linii. Se cere:
a) Să se afișeze litera (literele) care apare de cele mai multe ori;
b) Să se afișeze vocalele din text.
76. Se dă un vector cu n componente numere întregi, $2 \leq n \leq 20$. Scrieți un subprogram recursiv care să verifice dacă vectorul are cel puțin o componentă nulă.
77. Se dă un vector cu n componente numere întregi, $2 \leq n \leq 20$. Scrieți un subprogram recursiv care să verifice dacă vectorul are cel mult o componentă impară.
78. Se dă un vector cu n componente numere întregi, $2 \leq n \leq 20$. Scrieți un subprogram recursiv care să verifice dacă vectorul are exact o componentă pătrat perfect.
79. Se dă un vector cu n componente numere întregi, $2 \leq n \leq 20$. Scrieți un subprogram recursiv care să verifice dacă vectorul nu are componente negative.

80. Se dă un vector cu n componente numere întregi, $2 \leq n \leq 20$. Scrieți un subprogram recursiv care determină câte numere prime sunt în vector.
81. Se dă un vector cu n componente numere întregi, $2 \leq n \leq 20$. Scrieți un subprogram recursiv care calculează suma componentelor care au exact 3 cifre.
82. Se dă un vector cu n componente numere întregi, $2 \leq n \leq 20$. Scrieți un subprogram recursiv care determină suma cifrelor componentelor aflate pe poziții impare în vector.
83. Se dă un vector cu n componente numere întregi, $2 \leq n \leq 20$. Să se afișeze diferența dintre cea mai mare componentă impară negativă și cea mai mică componentă pară negativă din vector.
84. Se dă un vector cu n componente numere întregi, $2 \leq n \leq 20$. Scrieți un program care afișează cel mai mare număr natural ce se poate forma din toate cifrele componentei maxime din vector.
85. Se dă un vector cu n componente numere întregi, $2 \leq n \leq 20$. Afișați componentele vectorului după eliminarea cifrei/cifrelor din mijlocul fiecărei componente din vector.
86. Se da o matrice A cu m linii și n coloane, cu elemente numere întregi. Sa se afle suma elementelor aflate pe prima și ultima linie, pe prima și ultima coloana. Citirea se va face de la tastatură iar afisarea se va face în fișierul REZ.OUT.
87. Scrieți un program care citește de la tastatură un număr natural n ($n < 100$) și un șir cu n numere întregi din intervalul $[100, 999]$. Programul construiește un șir de numere rezultat prin înlocuirea fiecărui număr din șirul citit cu numărul obținut prin interschimbarea cifrei unităților cu cifra sutelor. Numerele din noul șir se vor afișa pe ecran separate printr-un singur spațiu.

Exemplu : pentru $n=3$ și șirul 123 , 904 , 500
se afișează 321 , 409 , 5.

88. Scrieți un program care afișează toate numerele naturale, mai mari decât 10 și mai mici decât o valoare data n , $n \leq 2.000.000$, formate doar din cifre identice.

Exemplu: pentru $n=195$ se afișează : 11 , 22 , 33 , 44 , 55 , 66 , 77 , 88 , 99 , 111.

89. În fișierul *Numere.txt* pe prima linie este memorat un număr natural n ($n < 10000$), iar pe linia următoare un șir de n numere naturale distincte cu maximum 4 cifre fiecare, separate prin câte un spațiu. Afișați pe prima linie a fișierului de ieșire *Rezultat.out* poziția pe care s-ar găsi primul element din șirul aflat pe linia a doua a fișierului, dacă șirul ar fi ordonat crescător. Numerotarea pozițiilor elementelor în cadrul șirului este de la 1 la n .

Exemplu: Dacă fișierul *Numere.txt* are următorul conținut:

6

267 13 45 628 7 79

Fișierul *Rezultat.out* va avea următorul conținut:5

90. Se dă un șir de caractere a . Să se afișeze șirul de caractere obținut prin transformarea literelor mari cuprinse între 'D' și 'O' în litere mici, respectiv literelor mici cuprinse între 'f' și 't' în litere mari. Ex : tEst => teSt.
91. Se citește un cuvânt s format din cel mult 100 de litere mici. Se citește un număr natural p ($p \leq 100$). Afișați cuvintele obținute prin eliminarea unei secvențe de p litere din s .
Exemplu: $s = \text{"adina"}$, și $p = 3$, rezultă cuvintele: na aa ad
92. Se citește un număr natural n ($n \leq 20$). Construiți și afișați o matrice patratică cu n linii și n coloane care să conțină primii $n \cdot n$ multipli ai lui n aranjați ca în exemplul de mai jos.
Exemplu: Pentru $n = 4$ se obține matricea:
4 8 12 16
20 24 28 32
36 40 44 48
52 56 60 64
93. Se citește o matrice patratică cu n linii și n coloane ($n \leq 100$), cu elemente numere naturale din intervalul $[0, 1000]$. Ștergeți liniile și coloanele pe care se află cele mai mici două elemente de pe diagonală principală
Exemplu:
matrice.in
5
3 2 4 3 2
2 1 5 4 6
3 2 4 3 5
4 3 2 2 1
4 3 5 6 7
matrice.out
3 4 2
3 4 5
4 5 7
(s-au șters liniile și coloanele 2 și 4)
94. Se citește o matrice patratică cu n linii și n coloane ($n \leq 100$) cu elemente numere naturale din intervalul $[0, 1000]$. Să se interschimbe elementele simetrice față de diagonală principală care au aceeași paritate și să se afișeze matricea rezultată.
Exemplu:
4
3 4 1 6
3 4 2 1
5 6 5 7

2 4 3 6

=>

3 4 5 2

3 4 6 1

1 2 5 3

6 4 7 6

95. Se citește un vector A cu n ($n \leq 1000$) elemente numere naturale. Afișați numărul de perechi de elemente din vector care au suma divizibilă cu n .
Exemplu: $n=4$, $A=\{2, 6, 8, 10\} \Rightarrow 3$, adică perechile $(2, 6)$, $(2, 10)$ și $(6, 10)$
96. Se citește un vector A cu n ($n \leq 1000$) elemente numere naturale. Afișați numărul de perechi egal depărtate de capetele vectorului care au proprietatea că încep cu aceeași cifră.
Exemplu: $n=6$, $A=\{34, 56, 77, 733, 45, 3\} \Rightarrow 2$, adică perechile $(34, 3)$ și $(77, 773)$
97. Se citește un vector cu n elemente numere întregi, n fiind cel mult egal cu 1000. Calculați și afișați câte dintre elementele vectorului sunt egale cu diferența dintre cel mai mare și cel mai mic element din vector.
Exemplu: Pentru $n=7$ și vectorul cu elementele $\{5, 4, 5, 2, 7, 6, 5\}$ se va afișa 3 (5 apare de 3 ori și e egal cu diferența dintre 7 și 2).
98. Se citește o matrice cu n linii și m coloane și elemente numere naturale. Să se ordoneze liniile matricii crescător după suma elementelor. Afișați matricea după realizarea ordonării cerute.
99. Se citește un vector cu componente numere naturale. Să se afișeze numărul cifrelor nule cu care se termină numărul format din produsul celor n componente.
Exemplu:
 $n=6$
tabloul: 34 25 100 444 16 32321
numărul de cifre nule cerut este 4
100. Fișierul text INTRARE.TXT conține pe prima linie o valoare naturală n ($1 < n < 100$), iar pe a doua linie, separate prin câte un spațiu, n numere reprezentând cele n elemente ale unui tablou unidimensional a . Fiecare dintre aceste numere are cel mult patru cifre. Scrieți un program care scrie în fișierul text IESIRE.TXT toate valorile obținute ca sumă a tuturor perechilor (a_i, a_j) , $1 \leq i < j \leq n$.
Exemplu: dacă fișierul INTRARE.TXT are următorul conținut:
4
1 4 3 2
atunci fișierul IESIRE.TXT va avea următorul conținut:
5 4 3 7 6 5 Soluție
101. Se dă un vector a cu n elemente numere naturale și o valoare naturală s . Afișați toate secvențele de valori din vectorul a care au suma egală cu s .

Exemplu:

$n=6$ $s=28$

$a=20$ 8 11 14 3 11

sevențele sunt:

20 8

11 14 3

14 3 11

102. Se citesc 2 numere naturale n și m . Să se construiască și să se afișeze o matrice cu n linii și m coloane care să conțină numerele de la 1 la $n*m$ ca în exemplul de mai jos.
Pentru $n=4$ și $m=5$ se obține matricea
- 1 2 3 4 5
10 9 8 7 6
11 12 13 14 15
20 19 18 17 16
103. Se citește un vector a cu n elemente numere întregi. Să se insereze între oricare 2 elemente alăturate de semne diferite un element obținut prin lipirea valorilor absolute ale celor 2 elemente, în ordinea lor din vector.
Exemplu: vectorul 3 -1 65 4 -9 se transformă în 3 31 -1 165 4 49 -9.
104. Se citește un vector a cu n elemente naturale. Să se elimine în mod repetat toate perechile de elemente alăturate identice până când nu mai există astfel de perechi.
Exemplu: din vectorul 3 7 4 5 5 6 6 4 7 9 8 8 vor rămâne doar elementele 3 și 9.
105. Se citesc 3 cuvinte s , a și b , s având cel mult 50 de litere, iar a și b având același cel mult 10 litere fiecare. Înlocuiți în cuvântul s toate aparițiile lui a cu b .
Exemplu:
 $s="abracadabra"$
 $a="bra"$
 $b="12345"$
va rezulta $s="a12345cada12345"$
106. Se citește un cuvânt a cu cel mult 100 de litere. Să se ștergă toate perechile de câte 2 litere alăturate identice. Dacă prin ștergere se obțin alte perechi de litere identice să se ștergă și acestea.
Exemplu: din cuvântul $abbapxsiisxh$ se obține ph (prin eliminări succesive: $aapxssxh$, $pxxh$ și apoi ph)
107. Se citesc din fișierul `fractii.txt` câte două numere, reprezentând numărătorul și numitorul fiecărei fracții. Să se afișeze în fișierul `rezultate.txt` aceste fracții simplificate. Exemplu: `fractii.txt` 6 4
10 5 150 20 rezultate.txt 3 2 2 1 15 2

108. Se citesc mai multe cuvinte din fișierul de intrare fraze.txt. Să se afișeze, pe câte un rând, cuvintele din fișier, lungimile lor și să se afișeze în final lungimea maximă și minimă a cuvintelor citite din fișier.
Exemplu:
eu sunt printre cei cuminti din scoala
Eu 2
Sunt 4
Printre 7
Cei 3
Cuminti 7
Din 3
Scoala 6
Lungimea minima a cuvintelor este 2
Lungimea maxima a cuvintelor este 7
109. Fișierul text numere.txt conține pe prima linie un număr natural n ($0 < n < 100000$), iar pe a doua linie, separate prin câte un spațiu, n numere naturale formate din cel mult 2 cifre. Scrieți un program care determină toate numerele conținute de a doua linie a fișierului care apar de cel puțin două ori în această linie. Programul va afișa pe ecran numerele determinate, o singură dată, în ordine crescătoare, pe aceeași linie, separate prin câte un spațiu.
Exemplu: dacă fișierul numere.txt are următorul conținut: 44 2 54 74 2 449 2 atunci pe ecran se va afișa: 2 44.
110. Fișierul text numere.txt conține pe prima linie un număr natural n ($0 < n < 100000$) iar pe doua linie, separate prin câte un spațiu, n numere naturale formate din cel mult 2 cifre. Scrieți un program care determină dacă numerele situate pe a doua linie a fișierului sunt în ordine strict crescătoare. În caz afirmativ, programul va afișa pe ecran mesajul DA, altfel va afișa mesajul NU. Exemplu: dacă fișierul numere.txt are următorul conținut:
7
3 5 2 1 5 23 1, atunci pe ecran se va afișa: NU.
111. Să se realizeze un program care să afișeze numerele din fișierul atestat.in care au numărul cifrelor pare egal cu numărul cifrelor impare. Afișați și numărul acestor numere. Exemplu: atestat.in :
12 1524 99 6543
Se va afișa : 12 1524 6543
112. Realizați un program care să folosească un subprogram recursiv pentru afișarea numerelor ce conțin numărul maxim de cifre distincte, citirea celor n valori întregi se face de la tastatură. Exemplu: $n=5$, 121 38 7 1188 22 se va afișa: 1188 38 121.
113. Se citește un număr natural $n \leq 100$ și apoi părțile reale și cele imaginare a n numere complexe. Apoi se citesc două numere reale a și b . Afișați în ordine crescătoare valorile modulelor numerelor complexe care au proprietatea că au modulul în afara intervalului $[a, b]$.

114. Se citesc coordonatele a n puncte din plan. Afișați lungimea laturii celui mai mare pătrat care se poate forma cu punctele citite.
Exemplu:
8
0 0
2 3
4 6
0 3
3 0
3 3
6 0
3 -3
Se va afișa 4.24264
115. Să se definească o structură pentru memorarea unei progresii aritmetice de rație r și termen inițial a . Să se afișeze primii 10 termeni ai progresiei date prin a și r .
116. În fișierul sărituri.in sunt notate numele, prenumele și lungimea săriturii pentru cei n elevi dintr-o clasă. Numele și prenumele sunt șiruri de caractere, iar lungimea săriturii este număr real. Să se construiască două fișiere text baieti.out și fete.out în care să se afișeze băieții și respectiv fetele din clasă, ordonați/ordonate descrescător după lungimea săriturii. Se consideră că toate fetele din clasă au prenumele terminat cu litera a .
117. Se citesc n triplete de numere naturale. Să se afișeze toate tripletele de numere citite consecutiv în care al treilea număr este restul împărțirii primului la al doilea.
118. Se citesc numere naturale până la întâlnirea numărului 0. Să se afișeze numerele din sir care sunt divizibile cu suma cifrelor lor.
119. Fie vectorii X și Z având fiecare n componente numere reale. Să se determine expresia $e = \max(x_1, y_n) + \max(x_2, y_{n-1}) + \dots + \max(x_n, y_1)$.
120. Fie un vector X de numere întregi. Să se formeze un vector Y de numere întregi în care $Y[i]$ să fie restul împărțirii lui $X[i]$ la suma divizorilor lui $X[i]$.
121. Fie vectorul X cu n componente numere întregi. Să se determine maximul componentelor divizibile cu 3 din vector.

122. Se dă un șir v cu n elemente numere întregi. Să se determine media aritmetică a elementelor pare.
123. Fiind dată o matrice simetrică cu elemente numere întregi, să se afișeze suma elementelor de pe diagonala secundară și elementele aflate deasupra și sub diagonala secundară.
124. Se citește de la tastatură o matrice a cu m linii și n coloane. Să se afișeze produsul elementelor pozitive aflate pe linii pare și coloane impare.
125. Se citește un șir de numere întregi din fișierul 'numere.txt'. Fișierul conține pe primul rând numărul n de elemente ale șirului, apoi, pe fiecare din următoarele n rânduri, câte un element. Să se afișeze suma elementelor pozitive ale șirului.
126. Să se scrie un program, în limbajul Pascal/C/C++, care să afișeze media aritmetică a elementelor de pe diagonala principală dintr-o matrice cu n linii și n coloane.
127. Fișierul atestat.in conține pe prima linie numărul n , natural nenul ($2 < n < 20$), ce reprezintă numărul de linii și de coloane ale unei matrice, iar pe următoarele n linii câte n numere reale separate prin câte un spațiu, reprezentând elementele matricei. Să se scrie un program, în limbajul Pascal/C/C++, care să determine suma elementelor matricei situate pe linia k , unde k ($0 < k \leq n$) este un număr natural citit de la tastatură și să scrie pe prima linie a fișierului atestat.out valoarea acestei sume.
128. Se dă un vector cu maxim 10 elemente, numere întregi. Să se calculeze media aritmetică a elementelor impare din vector.
129. Se dă un vector cu maxim 20 elemente, numere întregi. Să se afișeze elementele pozitive și impare ale vectorului.
130. Se dau 2 vectori a , b primul cu n elemente iar al doilea cu m elemente numere întregi. Să se constituie vectorul c în care să se depună elementele ce rezultă din intersecția celor 2 vectori.

131. Se dă o matrice cu n linii și n coloane numere naturale. Să se calculeze numărul elementelor divizibile cu 5 de pe diagonala principală.
132. Se citește de la tastatură o matrice cu n linii și n coloane cu elemente numere întregi. Să se afișeze pe un rând al ecranului media aritmetică a elementelor de pe diagonala principală iar pe alt rând produsul elementelor impare de pe diagonala secundară.
133. Se citește de la tastatură o matrice cu n linii și m coloane cu elemente numere întregi. Să se calculeze produsul elementelor prime din matrice.
134. Se citesc de la tastatură două numere întregi n și m . Să se construiască o matrice cu n linii și m coloane în care fiecare element al matriciei să fie egal cu produsul indicilor de linie, respectiv de coloană.
135. Se citește de la tastatură un șir de caractere s format din litere mici ale alfabetului englez și spații. Să se afișeze șirul astfel încât fiecare consoană a lui să fie transformată în majusculă.
136. Să se calculeze valoarea funcției $f: [-9,9]$ în punctul x , dacă:

$$f(x) = \begin{cases} 1, & \text{pentru } x \leq 1 \\ 3 + \ln(1+x), & \text{pentru } -1 < x \leq 0 \\ x^2 + 2, & \text{pentru } 0 < x \leq 1 \\ 10 - 3x, & \text{pentru } x > 1 \end{cases}$$

137. Se dă numărul natural $n > 1$. Să se genereze toți divizorii pozitivi d_1, d_2, \dots, d_m ai numărului n .
138. Să se genereze toate numerele prime mai mici decât un număr natural n dat
139. Se dau numărul natural $n > 1$ și numerele x_1, x_2, \dots, x_n . Să se determine cel mai mare număr negativ și pozițiile pe care se află el în șirul dat
140. Să se afișeze în fișierul "trei.out" toate numerele naturale de trei cifre care sunt divizibile cu suma cifrelor lor.

141. Se citește un număr natural n . Afișați termenul de rang n al șirului lui Fibonacci. Folosiți o funcție care returnează termenul de rang n al șirului lui Fibonacci.
142. Se citește un număr natural n ($n \leq 30$). Construiți și afișați o matrice patratică cu n linii și n coloane după modelul de mai jos, obținut pentru $n=4$.
- ```
63 25 7 1
25 13 5 1
7 5 3 1
1 1 1 1
```
143. Se dă un număr  $n$ . Construiți și afișați o matrice patratică cu  $n$  linii și  $n$  coloane care să conțină primii  $2n-1$  multipli ai lui  $n$ , ca în exemplu:  
Exemplu: Pentru  $n=4$  se obține matricea:
- ```
28 24 20 16
24 20 16 12
20 16 12 8
16 12 8 4
```
144. Se citește o matrice patratică cu n linii și n coloane ($n \leq 100$), cu elemente numere naturale din intervalul $[0,1000]$.
Ordonăți crescător elementele de pe diagonala principală prin interschimbări de linii și coloane.
Afișați matricea rezultată.
Exemplu:
matrice.in
- ```
5
4 2 3 4 5
2 3 4 5 6
4 3 2 5 4
6 7 5 6 7
1 2 5 4 2
```
- matrice.out
- ```
2 4 3 4 5
5 2 2 1 4
4 6 3 2 5
3 5 2 4 4
5 7 7 6 6
```
145. Scrieți un program care citește un număr natural $n > 1$ cu maximum 9 cifre, și afișează valoarea celui mai mic divizor prim a lui n , precum și puterea la care acest divizor apare în descompunerea în factori primi a numărului n .
146. Să se scrie un program care să calculeze și să afișeze pe ecran produsul cifrelor pare și suma cifrelor impare a unui număr natural n citit de la tastatură.

147. Să se verifice dacă doua cuvinte introduse de la tastatura (cuvintele având maxim 50 de caractere fiecare) sunt rime, adică ultimele p caractere ale celor doua cuvinte coincid, unde p ($2 \leq p \leq 10$) este dat.
148. Scrieți un program care construiește o matrice pătrată de ordin n formată din valorile 1 și 2 astfel încât elementele de pe diagonala principală și secundară să fie egale cu 1 iar restul elementelor cu 2. Valoarea lui n se citește de la tastatură: n număr natural ($2 < n < 23$), iar matricea se va afișa pe ecran, câte o linie a matricei pe fiecare rând pe ecran cu spații între elementele fiecărei linii.
149. Scrieți programul C/C++ care citește de la tastatură un număr natural n ($n < 100$) și un șir cu n numere întregi din intervalul $[100; 999]$; programul construiește un șir de numere rezultat prin înlocuirea fiecărui număr din șirul citit cu numărul obținut prin interschimbarea cifrei unităților cu cifra sutelor. Numerele din noul șir se vor afișa pe ecran separate printr-un singur spațiu. De exemplu, pentru $n=3$ și șirul 123, 904, 500, se afișează 321, 409, 5.
150. Se citește un număr natural $n \leq 100$ și apoi lungimile laturilor a și b dreptunghiuri. Afișați dreptunghiul de arie maximă și pe cel de perimetru minim. La afișare se vor scrie cele 2 laturi, aria și perimetrul dreptunghiurilor.
Exemplu:
4
5 6
3 4
7 8
3 8
se vor afișa:
7 8 56 30 (pentru dreptunghiul de arie maximă)
3 4 12 14 (pentru dreptunghiul de perimetru minim)
151. Se citesc coordonatele a n puncte din plan ($n \leq 100$). Afișați coordonatele colțului stânga jos al pătratului de latura n care conține număr maxim de puncte dintre cele citite. Dacă un punct se află pe latura pătratului se consideră în interiorul acestuia. Datele de intrare se citesc din fișierul puncte.in și se afișează în puncte.out.
Exemplu:
puncte.in
6
-10 -5
4 1
3 3
1 2
10 8
0 0
puncte.out
0 0
152. În fișierul sarituri.in sunt notate numele, prenumele și lungimea săriturii pentru cei n elevi dintr-o clasă. Numele și prenumele sunt șiruri de caractere, iar lungimea săriturii este număr real. Să se construiască două fișiere text baieti.out și fete.out în care să se afișeze baietii și respectiv fetele din



clasă, ordonați/ordonate descrescător după lungimea săriturii. Se consideră că toate fetele din clasă au prenumele terminat cu litera a.

153. Să se facă conversia unui număr natural dintr-o bază b în baza 10. ($2 \leq b; b \neq 10$)
154. Fișierele `cuv1.txt` și `cuv2.txt` conțin cuvinte, câte un cuvânt pe linie în ordine alfabetică. Să se construiască fișierul `cuv3.txt` care să conțină toate cuvintele din cele două fișiere, în ordine alfabetică.
155. La o stație de benzină există mai multe mașini caracterizate prin număr, marcă și culoare. Se cere să se scrie un program care
 - a) Determină numărul mașinilor Dacia;
 - b) Ordonează mașinile după culoare.

**INSPECTOR ȘCOLAR PENTRU INFORMATICĂ,
PROF. DANIELA IOANA TĂTARU**