

Nr. 401/15.01.2024

Avizat,
Inspector Școlar General Adjunct,
Prof. Răzvan Delcea VASILE

**SUBIECTE PROPUSE PENTRU
EXAMENUL DE ATESTAT PROFESIONAL LA INFORMATICĂ
Programare Pascal/C/C++
Matematică informatică, intensiv informatică**

1. Se citește un vector A cu n ($n \leq 1000$) elemente numere naturale din fișierul atestat.txt. Ordonăți crescător elementele prime și plasați-le la începutul vectorului și apoi descrescător pe cele neprime, în a doua parte a vectorului. Afișați în fișierul ieșire.txt vectorul ordonat ca în cerință.

Atestat.txt

7

33 13 77 19 5 34 100

ieșire.txt

5 13 19 100 77 34 33

2. Se citește din fișierul atestat.txt o matrice pătratică cu n linii și n coloane ($n \leq 100$) cu elemente numere naturale din intervalul $[0, 1000]$. Calculați sumele celor 4 zone ale matricii în care este aceasta împărțită de cele două diagonale și afișați în fișierul ieșire.txt litera corespunzătoare zonei cu suma maximă.

Exemplu:

Atestat.txt

4

3 4 1 6

3 4 2 1

5 6 5 9

2 4 3 6

ieșire.txt

E

3. Se citește din fișierul atestat.txt o matrice $n \times m$ cu elemente întregi. Ștergeți din matrice liniile care nu au elementele ordonate strict crescător sau strict descrescător și afișați în fișierul ieșire.txt matricea rezultată.

Exemplu:

Atestat.txt

4 4

2 1 3 4

6 10 11 13

6 5 4 2

4 3 2 44

ieșire.txt

6 10 11 13

6 5 4 2

4. Se citesc din fișierul "atestat.txt", de pe prima linie, 2 numere n și m ($1 \leq n, m \leq 50$). De pe următoarele 2 linii se citesc elementele a două șiruri de lungime n , respectiv m , , numere întregi, de maxim 9 cifre, care sunt ordonate crescător. Să se scrie un program care să construiască în memorie un șir care conține rezultatul interclasării celor două șiruri. Șirul rezultat se va afișa în fișierul ieșire.txt

Atestat.txt

2 3

4 6

1 2 7

iesire.txt

1 2 3 4 6 7

5. Pe prima linie a fișierului atestat.txt se găsesc numere naturale nenule. Se cere

a. Pentru fiecare număr impar din fișier să se afișeze cel mai mare număr care se poate forma din cifrele lui.

b. Să se verifice dacă numerele din fișierul de intrare reprezintă o progresie aritmetică. În caz afirmativ să se afișeze primul element și rația, altfel să se afișeze un mesaj corespunzător.

Atestat.txt

31 38 45 52 59 66

ieșire.txt

31 38 54 52 95 66

Primul termen este 31 și rația 7

6. Fie un tablou bidimensional cu n linii și n coloane. Se cere

a. Să se afișeze transpusa matricei.

b. Să se afișeze câte perechi de numere prime între ele sunt pe diagonala secundară a matricei inițiale.

Datele de intrare se vor citi din fișierul atestat.txt, astfel

-pe prima linie se găsește n

Pe următoarele n linii se găsesc câte n numere ce reprezintă elementele matricei.

Atestat.txt

4

1 2 3 4

5 6 7 8

9 10 11 12

13 14 15 16

ieșire.txt

1 5 9 13

2 6 10 14

3 7 11 15

4 8 12 16

5 perechi de numere prime sunt pe diagonala secundară a matricei inițiale

7. Prima linie a fișierului atestat.txt conține două numere naturale m și n ($1 \leq n$, $m \leq 100$) iar următoarele m linii câte n numere întregi cu maxim 4 cifre fiecare, separate prin câte un spațiu. Se cere să se afișeze în fișierul ieșire.txt câte din cele m linii din fișier sunt simetrice. Spunem că o linie a fișierului este simetrică dacă elementele egal depărtate de capetele liniei respective sunt egale (primul element de pe linie este egal cu ultimul element al liniei, al doilea cu penultimul etc.)

Exemplu

Atestat.txt are următorul conținut

3 5

15 3 0 3 15

5 9 1 9 5

6 8 20 8 3

ieșire.txt

2

8. Fișierele atestat1.txt și atestat2.txt conțin fiecare numele a 7 persoane, câte un nume pe fiecare linie, fiecare nume având cel mult 15 litere. Știind că în fiecare fișier numele sunt memorate în ordine alfabetică, scrieți un program C++ care să citească din cele două fișiere și să afișeze în fișierul ieșire.txt toate numele din cele două fișiere în ordine alfabetică, separate printr-un spațiu.

Exemplu

Atestat1txt are următorul conținut

Ana

Dana

Daniel

Ene

Mara

Nae

Paul

Exemplu

Atestat2txt are următorul conținut

Angi

Cora

Dora

Horia

Oana

Paul

Tibi

ieșire.txt

Ana Angi Cora Dana Daniel Dora Ene Horia Mara Nae Oana Paul Paul Tibi

9. Fișierul atestat.txt conține pe prima linie un număr natural $n(1 \leq n \leq 10000)$ iar pe a doua linie n numere reale separate prin câte un spațiu. Fiecare număr real este format din cel

mult 10 cifre, inclusiv partea zecimală. Scrieți programul C++ ce determină cifrele ce nu apar în scrierea nici unui număr real din fișier. Cifrele se vor afișa în fișierul ieșire.txt în ordine crescătoare, separate prin câte un spațiu. În cazul în care toate cifrele sunt utilizate în scrierea numerelor din fișier se va afișa mesajul NICI UNA.

Exemplu

Atestat.txt are următorul conținut

-1.23 36 22.57 208

ieșire.txt

4 9

10. Fișierul atestat.txt conține cel mult 1000 de numere întregi separate prin spații. Numerele din fișier au valori cuprinse între -30000 și 30000. Să se determine și să se afișeze în fișierul ieșire.txt cel mai mare număr din fișier precum și numărul de apariții ale acestuia.

Exemplu

Atestat.txt are următorul conținut

2 7 12 3 8 12 9 5

ieșire.txt

12 2

11. Scrieți un program care creează fișierul text iesire.txt ce conține o linie unică având în ordine descrescătoare toate numerele pare mai mici sau egale cu n , unde n este un număr natural citit de la tastatură ($n \leq 1000$). Numerele scrise în fișier vor fi separate prin câte un spațiu.

Se citește $n=11$

iesire.txt

10 8 6 4 2 0

12. Fișierul atestat.txt conține pe prima linie un număr natural nenul ($n \leq 100$) și pe următoarea linie n numere reale pozitive ordonate crescător separate prin câte un spațiu.

Să se scrie un program care citește din fișier numărul n și care determină numărul minim de intervale $[x, x+1]$ cu x număr natural a căror reuniune include toate numerele reale din fișier.

Atestat.txt

6

2.3 2.8 5.1 5.7 5.9 6.3

iesire.txt

3

13. Pe prima linie a fișierului atestat.txt se află un număr natural nenul mai mic sau egal cu 1000 și pe a doua linie un șir de n numere naturale, despărțite prin câte un spațiu fiecare număr fiind format din cel mult patru cifre. Scrieți un program care citește numerele din fișier și afișează mesajul DA dacă elementele pare sunt în ordine crescătoare, iar elementele impare sunt în ordine descrescătoare și mesajul NU în caz contrar.

Atestat.txt

7

10 1133 12 331 42 1354 221

Afișează DA

14. Din fișierul atestat.txt se citesc două numere naturale n și m ($n, m \leq 100$) și apoi citește o matrice cu n linii și m coloane, cu elemente numere naturale din intervalul $[0, 10000]$.

- Afișați pe prima linie a fișierului ieșire.txt elementul maxim din matrice.
- Afișați pe a doua linie a fișierului ieșire.txt câte dintre liniile matricei au elementele în ordine strict crescătoare.

atestat.txt

8 6

1 2 3 4 5 6

4 5 6 7 9 10

3 2 3 2 3 2

9 5 4 3 2 1

4 5 3 4 4 4

9 8 7 6 5 2

3 5 4 3 3 3

ieșire.txt

10

3

15. Se citesc din fișierul atestat.txt, de pe 2 linii consecutive, 2 numere mari scrise în baza 10, fiecare având cel mult 50 de cifre. Să se scrie un program care să calculeze suma lor, folosind șiruri în care se memorează cifrele numerelor. Rezultatul se va afișa în fișierul ieșire.txt.

16. Se citesc din fișierul atestat.txt, de pe 2 linii consecutive, 2 numere mari scrise în baza 10, fiecare având cel mult 50 de cifre. Să se scrie un program care să compare numerele, folosind șiruri în care se memorează cifrele numerelor. Rezultatul se va afișa în fișierul ieșire.txt.

17. Din fișierul atestat.txt se citește un număr n mai mic decât 2000000000 și apoi se citesc n numere naturale. Afișați în fișierul ieșire.txt lungimea celei mai lungi secvențe din numerele citite care are proprietatea ca începe și se termina cu aceeași valoare și nu mai conține acea valoare (în afara de primul și ultimul element al secvenței).

Exemplu:atestat.txt

14

3 2 4 3 4 2 3 4 5 6 7 2 5 5

ieșire.txt

7

Explicație: cea mai lungă secvență care respecta cerința este 2 3 4 5 6 7 2

18. Se citește din fișierul atestat.txt un număr n și un număr q , q din intervalul $[2,9]$. Verificați dacă n este corect scris în baza q . Să se afișeze da în fișierul ieșire.txt sau nu în caz contrar.

Atestat.txt

1372 9

ieșire.txt

Da

19. Se citește din fișierul atestat.txt un tablou unidimensional cu $n \leq 100$ componente numere naturale. Să se afișeze pe primul rând în fișierul ieșire.txt cifrele numărului celui mai mare care se poate obține din cifrele elementului minim și ale celui maxim din vector.

b. Să se afișeze pe al doilea rând din fișierul ieșire câte numere din tablou sunt pătrate perfecte.

Atestat.txt

6

271 109 28 713 14 36

ieșire.txt

74311

Tabloul are 1 pătrat perfect

20. Se citește din fișierul `atestat.txt`, de pe prima linie, un număr n ($1 \leq n \leq 50$). De pe următoarea linie se citesc elementele unui șir de lungime n , numere întregi, de maxim 9 cifre. Să se scrie un program care să realizeze sortarea crescătoare a elementelor șirului folosind metoda inserției. Rezultatul se va afișa în fișierul `ieșire.txt`.

Exemplu:

`Atestat.txt`

7

3 8 4 1 9 2 4

`ieșire.txt`

1 2 3 4 4 8 9

21. Se citesc din fișierul `atestat.txt`, de pe prima linie, 2 numere n și m ($1 \leq n, m \leq 50$). De pe următoarele 2 linii se citesc elementele a două șiruri de lungime n , respectiv m , numere întregi, de maxim 9 cifre, care sunt ordonate crescător. Să se scrie un program care să construiască în memorie un șir care conține rezultatul interclasării celor două șiruri. Șirul rezultat se va afișa în fișierul `ieșire.txt`.

Exemplu:

`Atestat.txt`

5 4

1 3 4 7 8

1 2 3 6

`ieșire.txt`

1 1 2 3 3 4 6 7 8

22. Se citesc din fișierul `atestat.txt`, de pe prima linie, 2 numere n și m ($1 \leq n, m \leq 50$). De pe următoarele două linii se citesc elementele a două șiruri de lungime n , respectiv m , numere întregi, de maxim 9 cifre. Elementele șirurilor sunt distincte două câte două. Scrieți un program, care să construiască în memorie un șir ce va conține intersecția celor două șiruri. Rezultatul se va afișa în fișierul `ieșire.txt`.

Exemplu:

Atestat.txt

5 4

1 3 4 7 8

1 2 3 6

ieșire.txt

1 3

23. Se citesc din fișierul atestat.txt, de pe prima linie, 2 numere n și m ($1 \leq n, m \leq 50$). De pe următoarele două linii se citesc elementele a două șiruri de lungime n , respectiv m , numere întregi, de maxim 9 cifre. Elementele șirurilor sunt distincte două câte două. Să se scrie un program, care să construiască în memorie un șir ce va conține reuniunea celor două șiruri. Rezultatul se va afișa în fișierul ieșire.txt.

Exemplu:

Atestat.txt

5 4

1 3 4 7 8

1 2 3 6

ieșire.txt

1 2 3 4 6 7 8

24. Se citește din fișierul atestat.txt, de pe prima linie, un număr n ($1 \leq n \leq 50$). De pe următoarea linie se citesc elementele unui șir de lungime n , numere întregi, de maxim 9 cifre. Să se scrie un program care să verifice dacă șirul citit conține numai elemente distincte.

Exemplu:

Atestat.txt

5

1 3 4 7 8

ieșire.txt

Da

25. Se citește din fișierul `atestat.txt`, de pe prima linie, un număr n ($1 \leq n \leq 50$). De pe următoarea linie se citesc elementele unui șir de lungime n , numere întregi, de maxim 9 cifre. Să se scrie un program care să realizeze sortarea crescătoare a elementelor șirului folosind metoda selecției. Rezultatul se va afișa în fișierul `ieșire.txt`.

Exemplu:

`Atestat.txt`

7

3 8 4 1 9 2 4

`ieșire.txt`

1 2 3 4 4 8 9

26. Se consideră un șir de n numere întregi distincte $n \leq 20$ și două numere întregi a, b .

Să se afișeze pe ecran toate elementele șirului care nu sunt în intervalul închis ce se poate forma cu numerele a și b .

b. Să se ordoneze crescător numai elementele situate între elementul minim și elementul maxim din șirul dat și să se afișeze pe ecran șirul obținut după ordonare.

`atestat.txt`

7

3 1 8 7 2 4 -1

5 3

`ieșire.txt`

1 8 7 2 -1

3 1 8 2 4 7 -1

27. Se citește din fișierul `atestat.txt`, de pe prima linie, un număr n ($1 \leq n \leq 50$). De pe următoarea linie se citesc elementele unui șir de lungime n , numere întregi, de maxim 9 cifre. Să se scrie un program care să realizeze sortarea crescătoare a elementelor șirului folosind metoda numărării. Se va ține cont de faptul că în șir poate să apară un element și de mai multe ori. Rezultatul se va afișa în fișierul `ieșire.txt`.

Exemplu:

`Atestat.txt`

7

3 8 4 1 9 2 4

ieșire.txt

1 2 3 4 4 8 9

28. Se citește din fișierul `atestat.txt`, de pe prima linie, un număr natural n ($1 \leq n \leq 50$). De pe următoarea linie se citesc n numere naturale de maxim 9 cifre. Să se scrie un program care să calculeze cel mai mare divizor comun al celor n numere de pe linia a doua a fișierului. Rezultatul se va afișa în fișierul `ieșire.txt`.

Exemplu:

`Atestat.txt`

5

14 8 22 4 32

`ieșire.txt`

2

29. Din fișierul `numere.in` se citește un număr natural n ($n \leq 100000$) și apoi n numere naturale cu cel mult 9 cifre fiecare. Afișați în fișierul `numere.out` cea mai lungă secvență de cifre identice care se obține prin lipirea celor n numere. Dacă există mai multe secvențe de lungime maximă, atunci se va afișa cea mai din dreapta.

`numere.in`

12

36 611 1111 12 11000000 0 0 0 0 3333 43219

`numere.out`

0000000000

30. Din fișierul `date.in` se citește un număr natural n și apoi se citesc n numere naturale cu cel mult 2 cifre fiecare. Afișați în fișierul `date.out` lungimea celei mai lungi secvențe din numerele citite care are proprietatea că începe și se termină cu aceeași valoare.

Exemplu:

`date.in`

13

3 2 13 10 2 10 12 6 7 5 10 2 13

`date.out`

11

Explicație: cea mai lungă secvență cerută începe și se termină cu 2 și conține 11 numere.