Nr. 189/15.01.2020

VIZAT,
INSPECTOR ŞCOLAR GENERAL ADJ,
PROF. IULIANA STANCU

SUBIECTELE PROBEI PRACTICE PENTRU EXAMENUL DE ATESTAT PROFESIONAL LA INFORMATICĂ, 2020 PROGRAMARE-INTENSIV

1. Fişierul text ATESTAT.IN conţine mai multe linii, pe fiecare linie existând câte un şir de numere naturale nenule mai mici sau egale decât 30000, despărţite prin câte un spaţiu; fiecare linie se termină cu numarul 0 (care se consideră că nu face parte din şirul aflat pe linia respectivă) şi conţine cel puţin două valori. Scrieţi programul care afişează pe ecran valoarea maximă din şirul care conţine cele mai puţine numere. În cazul în care există mai multe şiruri cu acelaşi număr minim de numere, se va afişa cea mai mare valoare care apare în unul dintre aceste şiruri.

Exemplu: dacă fişierul NUMERE.IN are conţinutul de mai jos, atunci pe ecran se va afişa numărul 88.

2 25 34 3 0
6 88 9 3 0
4 54 78 145 98 24 346 0

2. Fişierul text ATESTAT.IN conţine cel mult 10000 de numere naturale din intervalul închis [0,9], dintre care cel puţin unul este prim. Numerele se află pe mai multe rânduri, cele de pe acelaşi rând fiind separate prin câte un spaţiu. Scrieţi un program care determină şi afişează pe ecran cel mai mare număr prim care apare în fişier şi numărul de apariţii ale acestuia. Programul afişează pe ecran cele două valori determinate, separate printr-un spaţiu.

Exemplu: Se va afisa 5 2 dacă fişierul conţine numerele:
5 8 9
1 9 5 1
1 2 2

3. Fişierul text ATESTAT.IN conţine pe mai multe rânduri cel mult 50000 de numere naturale din intervalul închis [0, 99], numerele de pe acelaşi rând fiind separate prin câte un spaţiu. Scrieţi un program care afişează pe ecran, în ordine descrescătoare, acele numere din fişier care sunt mai mari decât un număr natural k, citit de la tastatură. Dacă un număr apare de mai multe ori, şi este mai mare decât k, se va afişa o singură dată. Numerele vor fi afişate câte 20 pe fiecare linie (cu excepţia ultimei linii care poate să conţină mai puţine valori), separate prin câte un spaţiu.

Exemplu: dacă fişierul conţine numerele:
15 8 99
25 3 37
15 14 2, iar pentru k se citeşte valoarea 7, se vor afişa numerele 99 37 25 15 14 8.

Str. Carpaţi nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro

4. Fişierul text ATESTAT.IN conţine cel mult 1000 de numere întregi de cel mult 9 cifre fiecare, numerele fiind separate prin câte un spaţiu; printre numerele din fişier există cel puţin două numere pozitive, aflate pe poziţii consecutive. Scrieţi un program C/C++ care afişează două numere pozitive, aflate unul după altul în fişier, a căror sumă este maximă. Dacă există mai multe soluţii, se afişează doar acea pereche pentru care diferenţa dintre cele două numere este maximă. Numerele vor fi afişate pe ecran, în ordinea din fişier, separate printr-un spaţiu.

Exemplu: dacă fişierul conţine numerele: -2 2 16 4 -1 25 -2 8 12 7 13 se vor afişa numerele 16 4, în această ordine, cu un spaţiu între ele.

5. Fişierul text ATESTAT.IN conţine pe prima linie o valoare naturală n cu exact 9 cifre nenule distincte. Scrieţi un program care citeşte din fişier numărul n şi afişează pe ecran cea mai mică valoare m formată din exact aceleaşi cifre ca şi n, astfel încât m>n. În cazul în care nu există o astfel de valoare, programul va afişa pe ecran mesajul Nu exista.

Exemplu: Dacă fişierul număr.txt conţine numărul 257869431, se va afişa pe ecran numărul 257891346.

6. Se defineşte şirul lui Fibonacci: $f_1=0$, $f_2=1$, $f_n=f_{n-1}+f_{n-2}$, $n \geq 3$. Fişierul **atestat6.in** conţine pe prima linie un număr natural n de cel mult 9 cifre. Să se descompună numărul natural n în sumă de termeni nenuli ai şirului Fibonacci, numărul termenilor din sumă trebuie sa fie minim. Se va utiliza o funcţie pentru determinarea celui de-al n-lea termen al şirului lui Fibonacci.

Exemple:
pentru n=8 se va afişa 5+3;
pentru n=24 se afişeaza 21+3.

7. În fişierul **atestat7.in** se află pe prima linie maxim un milion de numere naturale de cel mult 2 cifre fiecare. Scrieţi programul C/C++ care citeşte numerele din fişierul atestat7.in şi determină şi afişează pe ecran care dintre numerele citite apare de cele mai puţine ori în fişier. Se va utiliza un algoritm eficient din punct de vedere al spaţiului de memorie utilizat şi al timpului de executare.

Exemplu :

atestat7.in	se afişează
5 3 1 6 3 1 3 6 1 3	5

8. În fişierul **atestat8.in** se află pe prima linie un număr natural n (n<=9), iar pe a doua linie se află n numere cifre zecimale a_1, a_2, ….., a_n. Se cere să se calculeze suma $S= a_1a_{2\ldots}a_n + a_2a_{3\ldots}a_{n-1}\ a_na_1 + a_{3\ldots}\ a_na_1\ a_2 +\ldots + a_na_1\ a_2a_{3\ldots}a_{n-1}$. Se va folosi un subprogram pentru permutarea circulară la stânga, cu o cifră, a cifrelor unui număr .

Exemplu:fie n=3; $a_1=1$; $a_2=3$; $a_3=7$; S=137+371+713=1221

9. În fişierul **atestat9.in** se află un şir de cuvinte, câte unul pe linie. Să se scrie un program care citeşte cuvintele din fişier şi le afişează în fişierul **atestat8.out** în ordine crescătoare a lungimilor lor. Dacă există două sau mai multe cuvinte cu aceeaşi lungime se vor ordona alfabetic. Pentru ordonare se va scrie un subprogram în care se va utiliza unul din algoritmii de sortare studiaţi.

Str. Carpaţi nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro

Exemplu:

atestat9.in	**atestat9.out**
Mere Ana are	Ana are mere

10. Subprogramul **sum** primeşte prin intermediul parametrului n (1<n<30) dimensiunea unui tablou bidimensional pătratic, prin intermediul parametrului a tabloul bidimensional de numere reale (a_{ij} cu 1≤i≤n, 1≤j≤n) şi prin intermediul parametrului k un număr natural nenul (1<k<=2∗n). El returnează prin intermediul parametrului s suma tuturor elementelor a_{ij} cu proprietatea că i+j=k. Scrieţi programul care citeşte de la tastatură un tablou de numere reale cu n linii şi n coloane şi afişează suma elementelor din tablou aflate strict deasupra diagonalei principale a tabloului, folosind apeluri ale subprogramului **sum**, definit conform cerinţei.

Exemplu:

Date de intrare	se afişează
n=4 k=5 **2 7 3 1** **6 2 3 1** **0 3 1 5** **3 1 6 3**	20

11. Se citesc de la tastatură două numere naturale nenule p şi q (3<p<q≤999999999). Să se determine, dacă există, un număr prim x care aparţine intervalului închis [p,q] pentru care valoarea expresiei |q+p-2∗x| este minimă. S-a folosit notaţia |x| pentru modulul numărului x. Dacă nu există un astfel de număr, se va afişa valoarea 0, iar dacă există mai multe, se va afişa cel mai mic dintre ele. Pentru rezolvarea problemei se va utiliza un subprogram prim care primeşte prin intermediul parametrului n un număr natural (n>1) şi care returnează 1 dacă numărul n este prim sau 0, în caz contrar.

Exemplu:pentru p=6 şi q=18, dintre numerele prime 7, 11, 13 şi 17, se va afişa 13 deoarece |18+7-2∗13|<|18+7-2∗11|<|18+7-2∗17|<|18+7-2∗7|.

12. Subprogramul **nrcar** primeşte prin intermediul parametrului **s** un şir cu cel mult 200 de caractere şi prin parametrul **c** un caracter. El returnează prin intermediul parametrului **p** un număr natural reprezentând numărul de apariţii ale caracterului **c** în şirul **s**. Scrieţi programul care citeşte de la tastatură un şir de caractere (litere mari şi mici ale alfabetului englez şi cifre). Se cere să se determine numărul total de vocale din şirul dat, folosind apeluri ale subprogramului **nrcar**.Rezultatul se va afişa în fişierul text **atestat12.out**.

```
Exemplu :
Pentru s= "Informatica01Atestat2016" se va afisa 8 (deoarece sunt 8 vocale in şir)
```

13. Fişierul text **atestat13.in** conţine pe primul rând un număr natural **n (n>1)** şi apoi pe următorul rând **n** numere reale $\mathbf{x_1}$, $\mathbf{x_2}$, ..., $\mathbf{x_n}$. Să se determine câte dintre cele **n** numere citite se află în intervalul închis determinat de numerele $\mathbf{x_1}$ şi $\mathbf{x_n}$.

Str. Carpaţi nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro

Exemplu:
pentru **n=6** şi numerele **5**, **6.35**, **4.5**, **-10**, **-8**, **-9**, se afişează valoarea **4** (deoarece patru dintre numerele date, cele subliniate, se află în intervalul determinat de numerele **5** şi **–9**).

14. Fişierul text **atestat14.txt** conţine pe prima linie două numere naturale **n** şi **m** (**0<m<n<5000**), pe cea de a doua linie **n** numere naturale $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le 9$), iar pe cea de a treia linie **m** numere naturale $b_1, b_2, \ldots, b_m$ ($0 \le b_i \le 9$).

Scrieți un program care citește datele din fișier, verifică dacă șirul **b** se poate obține din șirul **a** și afișează pe ecran un mesaj corespunzător. Se va utiliza un algoritm eficient din punct de vedere al timpului de executare și al spațiului de memorie utilizat.

Exemplu:

Atestat14.txt	se afişează mesajul
7 5 **2 4 1 6 2 1 3** **6 1 2 3 2**	"**b se poate obtine din a**"

15. Scrieţi un program care citeşte din fişierul text **atestat15.in** un şir **v** de maxim 1.000.000 de numere naturale formate din exact două cifre fiecare şi afişează *distanţa* maximă care există între două elemente egale ale şirului. Definim *distanţa* dintre două elemente v_i si v_j prin modulul diferenţei indicilor celor două elemente, |**j-i**|. Dacă şirul conţine doar elemente distincte, se va afişa valoarea 0.

Exemplu:

Atestat15.in	se afişează	Explicaţii
10 14 **12** 10 10 14 15 10 12 90 12 13 15 80 **12** 90	12	**(v3=v15=12, 15-3=12)**

16. Pe prima linie a fişierului **atestat16.in** se găseşte un număr natural n, n≤1000, iar a doua linie un şir cu n numere naturale cu cel mult 9 cifre fiecare, separate prin cate un spaţiu. Scrieţi un program care scrie în fişierul **atestat16.out** pe prima linie numărul de valori prime din fişierul dat, iar pe următoarea linie toate numerele prime din şir, separate prin cate un spaţiu, în ordinea apariţiei lor în fişier. Pentru determinarea numerelor prime se va utiliza un subprogram **prim** cu un singur paramentru **x** număr natural (0≤x≤1000000000), care verifică dacă numărul transmis prin parametrul x este număr prim.

Exemplu:

atestat16.in	**atestat16.out**
8 12 13 2 123 5 41 77 23	5 13 2 5 41 23

17. Scrieţi un program care citeşte de la tastatură un număr natural n, 2≤n≤10 şi construieşte o matrice cu n linii şi n coloane care va conţine pe fiecare linie câte o permutare a mulţimii {1,2,…,n}, astfel încât pe linii diferite să avem permutări diferite conform exemplului de mai jos. Matricea astfel obţinută se va afişa în fişierul text a**testat17.out**, câte o linie a matricei pe câte o linie a ecranului.

Exemplu:

tastatură	**ecran**

Str. Carpaţi nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro

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

18. Pe prima linie a fişierului **atestat18.in** se găseşte un număr natural n, n≤100, iar a doua linie conţine un şir cu **maximum 1000000000** numere naturale, separate prin câte un spaţiu. Să se scrie în fişierul **atestat18.out** toate numerele din şir care sunt termeni din Şirul lui Fibonacci. Se va folosi un subprogram **fibo** care verifică daca un număr natural, dat ca parametru de intrare, este termen în şirul lui Fibonacci (f_1=0, f_2=1, f_n=f_{n-1}+f_{n-2}, pentru n>2).

Exemplu:

atestat18.in	atestat18.out
3 4 7 1 10 21 13 5 2 12	3 1 21 13 5 2

19. Se citesc din fişierul **atestat19.in** două numere naturale a şi b cu cel mult 9 cifre nenule fiecare. Scrieţi un program care scrie în fişierul **atestat19.out** cel mai mic număr natural care se poate forma cu toate cifrele celor două numere citite.

Exemplu:

atestat19.in	Atestat19.out
856331 17321	11123335678

20. Pe prima linie a fişierului **atestat20.in** se găseşte un număr natural n, n≤100, iar a doua linie conţine un şir cu n numere naturale cu cel mult nouă cifre fiecare, separate prin cate un spaţiu. Scrieţi un program care scrie în fişierul **atestat20.out** toate numerele din şir pentru care suma cifrelor este divizibilă cu 3. Se va folosi o funcţie recursivă **sumcif** care calculează şi returnează suma cifrelor parametrului de intrare x.

Exemplu:

atestat20.in	atestat20.out
7 124 51 231 7 24 31 5	51 231 24

21. Scrieţi un program care citeşte de pe prima linie a fişierului **atestat21.in** un număr natural **n** iar de pe următoarele **n** linii un tablou bidimensional cu n linii şi n coloane conţinând numere naturale şi care modifică matricea în felul următor: toate elementele liniilor care conţin valoarea minimă vor fi mărite cu valoarea maximă din matrice. Scrieţi în fişierul **atestat21.out** matricea astfel obţinută.

Exemplu:

atestat21.in	atestat22.out
4 2 4 3 1 2 2 4 4 3 2 1 2	 7 9 8 6 2 2 4 4 8 7 6 7

Str. Carpaţi nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro

5 3 5 2	5 3 5 2

22. Fişierul text **atestat22.in** conţine informaţii despre mai mulţi elevi, sub o formă nestructurată. Informaţiile sunt dispuse pe linii de maxim 200 de caracte şi pot conţine CNP-uri valide. Ştiid că CNP-ul unei persoane este un şir de 13 cifre consecutive, scrieţi un program care determină şi scrie în fişierul text **atestat21.out**, pe linii distincte, toate CNP-urile extrase din text. Dacă nu există nici un astfel de şir, se va scrie în fişier valoarea 0.

Exemplu:

atestat22.in	**atestat22.out**
Popesu George, 14 ani, 1020412342334; Gina Badea - 1031102343435, Republicii 7; Dana Marian: 2040405358687, fara nota, 2030609987654 - Janina Nalbu	1020412342334 1031102343435 2040405358687 2030609987654
atestat22.in	**atestat22.out**
Popesu George, 14 ani, 102412342334; Gina Badea - 10311023435, Republicii 7; Dana Marian: 204/040/5358687, fara nota	0

23. Spunem că un număr natural **x** este **rotund** dacă există un număr natural nenul **k**, mai mic strict decât numărul de cifre al lui **x**, astfel încât prin permutarea circulară a cifrelor numărului cu **k** poziţii la dreapta, să se obţină numărul iniţial. Scrieţi un program care citeşte din fişierul text **atestat23.in,** de pe prima linie un număr natural **n**, apoi de pe linia doi **n** numere naturale din intervalul [10, 10^9], separate prin câte un spaţiu. Programul determină şi afişează pe prima linie a ecranului câte numere rotunde sunt în fişier, iar pe linia a doua numerele rotunde în ordinea în care apar în fişierul atestat23.in, separate prin câte un spaţiu. Dacă fişierul nu conţine numere rotunde se va afişa valoarea 0.

Exemplu:

atestat23.in	**Ecran**
5 12 3232 123 144144 77	3 3232 144144 77
atestat23.in	**Ecran**
3 11231 45678 232	0

24. Scrieţi un program eficient din punct de vedere al timpului de execuţie, care generează şi scrie în fişierul text **atestat24.txt**, pe prima linie, separate prin câte un spaţiu, toate **palindroamele-munte** de nouă cifre (un palindrom are aspect de munte dacă cifrele sale sunt strict crescătoare până la jumătatea numărului. EX. 123454321). Se va scrie un subprogram care verifică dacă un număr este palindrom.

Pe a doua linie în fişier se va scrie numărul de palindroame-munte generate.

Str. Carpaţi nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro

25. Din fişierul ***vector.in*** se citeşte de pe prima linie un număr natural **n**, iar de pe a doua linie se citesc **n** elemente numere naturale. Să se calculeze CMMDC al elementelor vectorului citit, folosind metoda ***Divide et Impera.***

26. Se citeşte un şir de numere întregi din fisierul ***nr.txt***. Elementele şirului se găsesc toate pe un rând în fişier, separate prin spaţii. Folosind metoda ***Divide et Impera,*** să se determine simultan suma elementelor pare şi produsul elementelor impare.

27. Fişierul numere.txt conţine un şir de numere întregi, scrise toate pe un singur rând, separate prin spaţii. Să se creeze o listă liniară simplu înlănţuită care să conţină numai valorile divizibile cu 3 din fişierul dat şi să se afişeze pe ecran lista astfel obţinută.

28. Se dă un grup format din **n** persoane, care se cunosc sau nu între ele. De la tastatură se introduc m perechi de numere întregi (x,y) cu semnificaţia ”persoana x cunoaşte pe persoana y”. Relaţia de cunoştinţă nu este neapărat reciprocă. Numim celebritate, o persoană care este cunoscută de către toate celelalte persoane din grup, dar ea nu cunoaşte pe nici un alt membru al grupului. Să se determine dacă în grup există o astfel de celebritate.

29. Din fişierul text ***clasa.in*** se citeşte de primul rând un număr natural **n**, iar de pe următoarele n rânduri se citesc următoarele informaţii despre fiecare elev din clasă: numele, prenumele şi media. Să se afişeze în fişierul ***clasa.out*** elevii din clasă ordonaţi descrescător după medie.

30. Se dă un arbore binar cu **n** noduri prin vectorii de descendenţi S şi D citiţi de la tastatură. Afisaţi pe ecran pe rânduri separate: frunzele arborelui, vârfurile cu un singur descendent direct şi vârfurile cu doi descendenţi direcţi.

31. Se citeşte dintr-un fişier text un număr natural n, $n \geq 1$ şi apoi cele n elemente numere reale distincte ale unui vector. Fără a ordona efectiv vectorul să se determine ce poziţie ar ocupa ultimul element din vector, dacă acesta ar fi supus unui criteriu de ordonare.

32. Se dă un vector v cu n elemente numere naturale. Să se afişeze 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 fişierul text ‘fisa18.txt’, fiecare pe câte o linie a fişierului.

33. Fişierul text ’fisa18.txt’conţine pe prima linie un număr real pozitiv x, care are cel mult trei cifre la partea întreagă şi cel mult cinci cifre la partea zecimală. Scrieţi un program care afişează pe ecran, separate printr-un spaţiu, două numere naturale, al căror raport este egal cu x şi a căror diferenţă absolută este minimă.

34. Se dă fişierul text ’f19_2in’, care conţine pe prima sa linie un număr natural nenul n, iar pe urmatoarele n linii câte un număr natural format din cel mult opt cifre. Să se afişeze într-un fişier text ’f19_2.out’, pe câte o linie a fisierului, câte un număr din fişierul dat, urmat de caracterul “:” şi de cifrele care apar în scrierea acestuia, în ordine crescătoare, separate prin câte un spaţiu.

Str. Carpaţi nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro

35. Fişierul text 'f19_3.txt'conţine pe prima linie un număr natural nenul n, n≤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 mulţimii {1, 2, 3, …, n}, afişându-se un mesaj corespunzător.

36. Se dă un vector v cu n elemente numere naturale, cu cel mult sase cifre fiecare. Să se scrie într-un fişier text, pe câte o linie a fişierului, pentru fiecare număr din cele n date, cel mai apropiat număr de acesta, care începe şi se termină cu aceeaşi cifră, mai mare decât numărul respectiv.

37. Se citesc 2 şiruri de caractere. Să se construiască un şir ce conţine primele 3 caractere din fiecare şir citit. Ex: a="Informatica"; b="aplicata"; Se va construi: c="Infapl"

38. Se citesc numele şi nota de la informatică a doi elevi. Să se afişeze numele elevilor în ordinea descrescătoare a notei de la informatică. Dacă amândoi elevii au aceeaşi notă, se va afişa primul, cel care este alfabetic mai mic.

39. Se consideră un text cu cel mult 70 de caractere (litere mici ale alfabetului englez si spatii), în care cuvintele sunt separate prin unul sau mai multe spatii. Înaintea primului cuvânt si după ultimul cuvânt nu există spatii. Scrieti un program care citeste de la tastatură un text de tipul mentionat mai sus si afisează pe ecran numărul de cuvinte în care apare litera a. Exemplu: pentru textul : "voi sustine examenul la informatica" se va afisa valoarea 3.

40. Se citeşte un şir de maxim 20 caractere ce nu conţine caractere albe. Să se elimine toate consoanele şi să se afişeze şirul.

41. Fişierul atestat.txt conţine un text scris cu litere mari pe una sau mai multe linii. Se cere:
a) Să se afişeze litera (literele) care apare de cele mai multe ori;
b)Să se afişeze vocalele din text.

42. Fişierul atestat.in conţine două linii. Pe prima linie este scris un număr natural nenul n, (5<n<30). Pe cea de-a doua linie a fişierului sunt scrise n numere naturale separate prin câte un spaţiu, formate fiecare din cel mult 9 cifre, reprezentând un şir de n numere naturale.

43. Să se scrie un program în limbajul Pascal/C/C++, care:

a) să afişeze pe ecran, în linie, numerele din şir, separate prin câte un spaţiu;
b) să afişeze pe ecran, pe linii diferite, cel mai mic număr a şi cel mai mare număr b din şirul dat;
c) să scrie în fişierul atestat.out cel mai mare divizor comun al numerelor a şi b, determinate la punctul b).

Notă: Programul va conţine cel puţin un subprogram definit de utilizator.

Exemplu:

atestat.in							Date de ieşire:						
6						a)	123	55	372	3465	242	44	
123	55	372	3465	242	44	b)	44						
							3465						
							Fişierul atestat.out conţine:						

Str. Carpaţi nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro

c) | 11

44. Fişierul atestat.in conţine două linii. Pe prima linie este scris un număr natural nenul n, (5<n<30). Pe cea de-a doua linie a fişierului sunt scrise n numere naturale separate prin câte un spaţiu, formate fiecare din cel mult 9 cifre, reprezentând un şir de n numere naturale.

Să se scrie un program în limbajul Pascal/C/C++, care:
a) să afişeze pe ecran, în linie, toate numerele din şir, separate prin câte un spaţiu;
b) să afişeze pe ecran, în linie, toate numerele din şir formate numai din cifre pare (dacă nu există astfel de numere în şir se va afişa mesajul "NU EXISTĂ NUMERE NUMAI CU CIFRE PARE");
c) să citească de la tastatură două numere naturale nenule p1 şi p2 (1<p1<p2<n), să ordoneze descrescător numerele din şir situate între poziţiile p1 şi p2, inclusiv, şi să scrie noul şir în fişierul atestat.out, pe o linie, numerele separându-se prin câte un spaţiu.

Notă: Programul va conţine cel puţin un subprogram definit de utilizator.

Exemplu: de la tastatură se citesc: p1=2 şi p2=4

atestat.in		Date de ieşire:
6	a)	1233 22 1785 56 15657 457
1233 22 1785 56 15657 457	b)	22
		Fişierul atestat.out conţine:
	c)	1233 1785 56 22 15657 457

45. Fişierul atestat.in conţine două linii. Pe prima linie este scris un număr natural nenul n, (5<n<30). Pe cea de-a doua linie a fişierului sunt scrise n numere reale separate prin câte un spaţiu, reprezentând un şir de n numere reale.

Să se scrie un program în limbajul Pascal/C/C++, care:
a) să afişeze pe ecran, în linie, toate numerele din şir, separate prin câte un spaţiu;
b) să afişeze pe următoarea linie a ecranului, media aritmetică a numerelor negative din şir, cu o precizie de 2 zecimale (dacă şirul nu conţine numere negative se va afişa 0);
c) să citească de la tastatură două numere naturale nenule p1 şi p2 (1<p1<p2<n), să ordoneze crescător numerele din şir situate între poziţiile p1 şi p2, inclusiv, şi să scrie noul şir în fişierul atestat.out, pe o linie, numerele separându-se prin câte un spaţiu.

Notă: Programul va conţine cel puţin un subprogram definit de utilizator.

Exemplu: de la tastatură se citesc: p1=2 şi p2=4

atestat.in		Date de ieşire:
6	a)	-56.765 2.3 4.56 -1.2 -1.8 3
-56.765 2.3 4.56 -1.2 -1.8 3	b)	-19.92
		Fişierul atestat.out conţine:

Str. Carpaţi nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro

c)	-56.765	-1.2	2.3	4.56	-1.8	3

46. Fişierul atestat.in conţine două linii. Pe prima linie este scris un număr natural nenul n, (5<n<30). Pe cea de-a doua linie a fişierului sunt scrise n numere naturale separate prin câte un spaţiu, formate fiecare din cel mult 4 cifre, reprezentând un şir de n numere naturale distincte.

Să se scrie un program în limbajul Pascal/C/C++, care:
a) să afişeze pe ecran, în linie, toate numerele din şir, separate prin câte un spaţiu;
b) să afişeze pe ecran, pe linii diferite, cel mai mic număr din şir şi poziţia acestuia;
c) să scrie în fişierul atestat.out, pe o linie, separate prin câte un spaţiu, toate numerele *perfecte* din şirul dat (dacă nu există astfel de numere, se va se va afişa mesajul "NU EXISTĂ NUMERE PERFECTE"). Un număr este *perfect* dacă este egal cu suma divizorilor lui pozitivi, exceptându-l pe el însuşi, de exemplu: 6 = 1+2+3.

Notă: Programul va conţine cel puţin un subprogram definit de utilizator.

Exemplu:

atestat.in							Date de ieşire:					
6						a)	28	11	81	496	6	100
28	11	81	496	6	100	b)	6					
							5					
							Fişierul atestat.out conţine:					
						c)	28	496	6			

47. Fişierul **atestat.in** conţine două linii. Pe prima linie este scris un număr natural nenul **n, (5<n<30)**. Pe cea de-a doua linie a fişierului sunt scrise **n** numere naturale separate prin câte un spaţiu, formate fiecare din cel mult **4** cifre, reprezentând un şir de **n** numere naturale. Şirul conţine cel puţin două numere pare.

Să se scrie un program în limbajul **Pascal/C/C++,** care:
a) să afişeze pe ecran, în linie, toate numerele din şir, separate prin câte un spaţiu;
b) să afişeze pe următoarea linie a ecranului, media aritmetică a tuturor numerelor pare din şir ;
c) să scrie în fişierul **atestat.out**, pe o linie, separate prin câte un spaţiu, numerele de tip palindrom din şirul dat (dacă nu există astfel de numere, se va afişa mesajul "**NU EXISTĂ NUMERE PALINDROM**"). Un număr este palindrom dacă numărul citit de la stânga la dreapta este egal cu numărul citit de la dreapta la stânga, de exemplu: **33, 141, 2552.**

Notă: Programul va conţine cel puţin un subprogram definit de utilizator.

Exemplu:

atestat.in							**Date de ieşire:**					
6						**a)**	**2552**	**56**	**32**	**444**	**46**	**1221**
2552	**56**	**32**	**444**	**46**	**1221**	**b)**	**626**					
							Fişierul **atestat.out** conţine**:**					
						c)	**2552**	**444**	**1221**			

Str. Carpaţi nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro

48. Din fisierul date.in se citeste un numar natural n(n<=100) si apoi o matrice patratica cu n linii si n coloane cu elemente numere naturale. Identificati numerele prime de pe diagonala secundara a matricii si stergeti atat linia cat si coloana pe case se afla acestea. Afisati matricea rezultata in fisierul date.out. Se vor scrie si apela functii pentru:
- citirea matricii
- afisarea matricii
- verificarea daca un numar este prim
- stergerea unei linii dintr-o matrice
- stergerea unei coloane dintr-o matrice

Exemplu:
date.in
5
1 2 3 4 6
3 2 4 5 7
6 8 3 8 6
8 7 8 9 9
8 2 6 6 8
date.out
1 6
8 8

49. Se citeste un numar natural n cu cel mult 9 cifre. Scrieti o functie care calculeaza si returneaza cel mai mare numar care se poate forma cu cifrele lui n. Pe langa functia ceruta se vor scrie si apela functii pentru:
- transformarea unui numar in vector de cifre
- ordonarea descrescatoare a unui vector
- transformarea din vector de cifre in numar.

Exemplu: Cu cifrele numarului 3426096 ce mai mare numar care se poate forma este 9664320.
Subprogramul sub primeste prin intermediul parametrului n (cel mult 9) dimensiunea unei matrice patratice, prin intermediul parametrului a matricea patratica de dimensiune n cu elementele numere intregi si prin intermediul parametrului k un numar natural nenul din intervalul [2,2n]. Subprogramul sub calculeaza si returneaza suma tuturor elementelor a[i][j] cu proprietatea cã i+j=k.

50. Scrieti programul care citeste un numar natural n si un tablou de numere intregi cu n linii si n coloane si afiseaza suma elementelor din tablou aflate strict deasupra diagonalei secundare a tabloului, folosind apeluri ale subprogramului sub, definit conform cerintei.
Exemplu:
n=4
1 6 3 1
6 1 3 1
1 3 1 6
3 1 6 1
se va afisa 18

51. Fisierul DATE.IN contine pe prima linie doua numare naturale nenule n si m. Scrieti un program care memoreaza în fisierul text DATE.OUT toate numerele prime din intervalul deschis (n,m). Numerele se scriu în ordine crescătoare, câte 10 numere pe fiecare linie a fisierului.
Ex.

Str. Carpați nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro

DATE.IN
87 241
DATE.OUT
89 97 101 103 107 109 113 127 131 137
139 149 151 157 163 167 173 179 181 191
193 197 199 211 223 227 229 233 239

52. Pe prima linie a fişierului *Matrice.in* este scris un număr natural *n* iar pe fiecare din următoarele *n* linii ale fişierului, sunt scrise câte *n* numere naturale, formate fiecare din cel mult două cifre, separate prin câte un spaţiu, reprezentând valorile elementelor unei *matrici patratice A* cu *n* linii.

Se considera subprogramele:

- – s1 cu doi parametrii *n* si *a*, care determină, in urma apelului, citirea numerelor din fişierul *Matrice.in* şi returnarea, prin intermediul parametrului *n*, a numărului de linii ale matricei din fisier, iar prin intermediul parametrului *a*, returnarea unui *tablou bidimensional* pătratic cu *n* linii care memorează valorile elementelor matricei A din fişierul de intrare
- – s2 cu doi parametrii *v* (un *tablou unidimensional* cu cel mult 30 de elemente) si *k* (un număr natural reprezentând numărul efectiv de elemente ale tabloului *v*)

Cerinţă

- – scrieţi definiţia completă a subprogramelor s1 şi s2
- – scrieţi un program Pascal/C/C++ care citeşte de la tastatură un număr natural *m* şi care, folosind apeluri utile ale subprogramelor s1 şi s2, verifică dacă matricea A, scrisă în fişierul *Matrice.in*, poate fi matricea de adiacenţă a unui graf orientat cu *n* vârfuri şi *m* arce, caz în care, programul va afişa pe ecran, pe o singură linie, separate prin câte un spaţiu, gradele exterioare ale tuturor vârfurilor grafului, în ordinea crescătoare a etichetelor lor. Daca matricea A nu poate fi matricea de adiacenţă a unui graf orientat cu *n* varfuri şi *m* arce, atunci programul va afişa pe ecran mesajul IMPOSIBIL

53. Scrieţi programul C++ care citeşte de la tastatură un şir s de cel mult 30 de litere şi o literă c; programul determină dublarea fiecărei apariţii a literei c în s şi scrie noul şir obţinut în fişierul text BAC.TXT.

De exemplu, dacă se citeşte şirul: alfabetar şi caracterul a atunci fişierul BAC.TXT va conţine şirul: aalfaabetaar.

54. Se citesc de la tastatură două şiruri de caractere formate din cel mult 50 de litere fiecare. Să se afişeze pe ecran şirul format prin preluarea alternativă, din fiecare şir, a câte unei litere (prima literă a primului şir, apoi prima literă a celui de-al doilea, apoi a doua literă a primului şir, apoi a doua literă a celui de-al doilea şir etc). Când se epuizează literele din unul dintre şiruri, se vor prelua toate literele rămase din celălalt şir. Dacă se citesc şirurile abc şi mnprtxb se va afişa şirul ambncprtxb.

55. Scrieţi programul C++ care citeşte de la tastatură un cuvânt de cel mult 15 litere mici ale alfabetului englez şi care scrie pe ecran, pe linii distincte, cuvintele obţinute prin ştergerea

Str. Carpaţi nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro

succesivă a vocalelor în ordinea alfabetică a lor (a,e,i,o,u). La fiecare pas se vor şterge toate apariţiile din cuvânt ale unei vocale (ca în exemplu).

Exemplu: Dacă se citeşte cuvântul bacalaureat se afişează:

bcluret (s-au şters toate cele patru apariţii ale vocalei a)

bclurt (s-a şters unica apariţie a vocalei e)

bclrt (s-a şters unica apariţie a vocalei u)

56. Se citesc de la tastatură n propoziţii (0<n<101), având fiecare maximum 255 de caractere. Ştiind că oricare două cuvinte consecutive dintr-o propoziţie sunt despărţite printr-un singur spaţiu şi că fiecare propoziţie se termină cu Enter, să se afişeze pe ecran propoziţia care are cele mai multe cuvinte. Dacă două sau mai multe propoziţii au acelaşi număr de cuvinte se va afişa prima dintre ele, în ordinea citirii. De exemplu, pentru n = 3 şi următoarele propoziţii:

Azi sunt încă elev.

Maine am examen de bac.

Ura, voi fi student!

Se va afişa Maine am examen de bac.

57. Un şir cu maximum 255 de caractere conţine cuvinte separate prin caracterul *. Cuvintele sunt formate numai din litere mici ale alfabetului englez. Scrieţi un program C++ care citeşte un astfel de şir şi afişează pe ecran şirul obţinut prin eliminarea tuturor apariţiilor primului cuvânt, ca în exemplu. Exemplu: pentru şirul: bine*bine*e*foarte*bine* se va afişa: **e*foarte**

58. Se consideră două cuvinte. Să se verifice dacă cel de-al doilea cuvânt este obţinut din primul prin permutarea literelor. Ex. Cuvântul t=’corupt’ se poate obţine din cuvântul s=’cuptor’ prin permutarea literelor.

Str. Carpaţi nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro

59. Scrieţi un program C/C++ care citeşte de la tastatura un număr natural n (2<n<21) si apoi n linii cu cate n numere întregi de cel mult 7 cifre ce formează un tablou bidimensional a. Sa se afişeze pe ecran diferenţa dintre suma elementelor de pe diagonala principala si suma elementelor de pe diagonala secundara a matricei a.

60. Scrieţi programul C/C++ care citeşte de la tastatura un număr natural n (n<100) si un sir cu n numere întregi din intervalul [100 ;999] ; programul construieşte un sir de numere rezultat prin înlocuirea fiecărui număr din şirul citit cu numărul obţinut prin interschimbarea cifrei unitatilor cu cifra sutelor. Numerele din noul sir se vor afişa pe ecran separate printr-un singur spaţiu. De exemplu , pentru n=3 si şirul 123 , 904 , 500 , se afişează 321 , 409 , 5.

INSPECTOR ŞCOLAR PENTRU INFORMATICĂ,
PROF. DANIELA IOANA TĂTARU

Str. Carpați nr.15, 140059, Alexandria
Tel: +40 (0)247 31 27 12
Fax: +40 (0)247 31 70 92
E-mail: isjteleo@yahoo.com isjteleo@isjtr.ro
Web site: www.isjtr.ro