

**SUBIECTELE PROBEI PRACTICE PENTRU
EXAMENUL DE ATESTAT PROFESIONAL
INFORMATIC - PROGRAMARE
SPECIALIZAREA MATEMATICĂ - INFORMATIC**

Subiectul 1.

Fișierul **atestat.in** conține două linii. Pe prima linie este scris un număr natural nenul **n**, ($5 < n < 15$). Pe cea de-a doua linie a fișierului sunt scrise **n** numere întregi separate prin câte un spațiu, formate fiecare din cel mult **4** cifre, reprezentând un șir de **n** întregi.

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

- să afișeze pe prima linie a ecranului, valorile din fișier care sunt numere palindroame. Un număr este palindrom dacă și numai dacă este egal cu oglinditul său. Exemplu **1221** este palindrom.
- să afișeze pe a doua linie a ecranului separați prin câte un spațiu, numărul de divizori proprii pentru fiecare număr din șirul din fișier;
- să scrie în fișierul **atestat.out**, pe prima linie, toate numerele impare din șir, separate prin câte un spațiu.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in			Date de ieșire:
6	a)	121 7 1551 5	
121 14 7 1551 27 5	b)	1 2 0 6 2 0	
		Fișierul atestat.out conține:	
	c)	121 7 1551 27 5	

Subiectul 2.

Fișierul **atestat.in** conține două linii. Pe prima linie este scris un număr natural nenul **n**, ($5 < n < 30$). Pe cea de-a doua linie a fișierului sunt scrise **n** numere întregi pozitive separate prin câte un spațiu, formate fiecare din cel mult **4** cifre. În șirul de valori vor fi cel puțin o valoare par și una impară.

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

- să afișeze pe prima linie a ecranului, valorile din fișier care sunt numere prime. Un număr este prim dacă și numai dacă are exact doi divizori 1 și numărul însuși. 1 nu este număr prim.

- b) să afișeze pe a doua linie a ecranului separate prin câte un spațiu, cea mai mare valoare pară și cea mai mică valoare impar care apar în irul valorilor din fișier;
- c) să scrie în fișierul **atestat.out**, pentru cea mai mare valoare pară, **max**, determinată la punctul anterior două valori impare prime care au suma egală cu **max**.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in	Date de ieșire:
6	a) 7 101 11 5
124 1 7 101 11 5	b) 124 1
	Fișierul atestat.out conține:
	c) 11 113

Subiectul 3.

Fișierul **atestat.in** conține două linii. Pe prima linie este scris un număr natural nenul **n**, ($5 < n < 30$). Pe cea de-a doua linie a fișierului sunt scrise **n** numere naturale separate prin câte un spațiu, formate fiecare din cel mult 9 cifre, reprezentând un ir de **n** numere naturale.

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

- a) să afișeze pe prima linie a ecranului valorile pare din fișier, separate prin câte un spațiu;
- b) să afișeze pe ecran, pe linii diferite, cel mai mic număr **a** și cel mai mare număr **b** din irul dat;
- c) să scrie în fișierul **atestat.out** cel mai mare divizor comun al numerelor **a** și **b**, determinate la punctul b).

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in	Date de ieșire:
6	a) 372 242 44
123 55 372 3465 242 44	b) 44
	3465
	Fișierul atestat.out conține:
	c) 11

Subiectul 4.

Fișierul **atestat.in** conține două linii. Pe prima linie este scris un număr natural nenul **n**, ($5 < n < 30$). Pe cea de-a doua linie a fișierului sunt scrise **n** numere naturale separate prin câte un spațiu, formate fiecare din cel mult **9** cifre, reprezentând un ir de **n** numere naturale.

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

- să afișeze pe prima linie a ecranului valorile impare din fișier, separate prin câte un spațiu;
- să afișeze pe a doua linie a ecranului, toate valorile din fișier care sunt formate doar din cifre pare, dacă nu există astfel de valori se va afișa mesajul: „**Nu există**”
- să scrie în fișierul **atestat.out** pentru prima valoare din fișierul **atestat.in**, cel mai mare și cel mai mic număr natural care se pot construi din cifrele distincte nenule ale numărului citit.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in		Date de ieșire:
6	a)	1233 1785 15657 457
1233 22 1785 56 15657 457	b)	22
		Fișierul atestat.out conține:
	c)	321 123

Subiectul 5.

Fișierul **atestat.in** conține două linii. Pe prima linie este scris un număr natural nenul **n**, ($5 < n < 30$). Pe cea de-a doua linie a fișierului sunt scrise **n** numere naturale separate prin câte un spațiu, formate fiecare din cel mult **4** cifre, reprezentând un ir de **n** numere naturale.

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

- să afișeze pe prima linie a ecranului valorile pare din fișier, separate prin câte un spațiu;
- să afișeze pe a doua linie a ecranului, toate valorile din fișier care sunt formate doar din cifre impare, dacă nu există astfel de valori se va afișa mesajul: „**Nu există**”
- să scrie în fișierul **atestat.out** toate numerele care au un număr **k** de divizori primi; numărul natural **k** se va citi de la tastatură.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in		Date de ieșire:
6	a)	30 8 42 700
30 8 42 15 700 37	b)	15 37 K=3
		Fișierul atestat.out conține:
	c)	30 42 700

Subiectul 6.

Fișierul **atestat.in** conține două linii. Pe prima linie este scris un număr natural nenul **n**, ($5 < n < 30$). Pe cea de-a doua linie a fișierului sunt scrise **n** numere reale separate prin câte un spațiu, reprezentând un șir de **n** numere reale.

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

- să afișeze pe ecran, pe prima linie, toate numerele din fișier, separate prin câte un spațiu;
- să afișeze pe următoarea linie a ecranului, media aritmetică a numerelor negative din șir, cu o precizie de 2 zecimale (dacă șirul nu conține numere negative se va afișa 0);
- să citească de la tastatură două numere naturale nenule p_1 și p_2 ($1 < p_1 < p_2 < n$), să ordoneze crescător numerele din fișier situate între pozițiile p_1 și p_2 , inclusiv, și să scrie noul șir în fișierul **atestat.out**, pe o linie, numerele separându-se prin câte un spațiu.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu: de la tastatură se citesc: $p_1=2$ și $p_2=4$

atestat.in		Date de ieșire:
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
		Fișierul atestat.out conține:
	c)	-56.765 -1.2 2.3 4.56 -1.8 3

Subiectul 7.

Fișierul **atestat.in** conține două linii. Pe prima linie este scris un număr natural nenul **n**, ($5 < n < 30$). Pe cea de-a doua linie a fișierului sunt scrise **n** numere naturale separate prin câte un spațiu, formate fiecare din cel mult 4 cifre, reprezentând un șir de **n** numere naturale. Șirul conține cel puțin două numere impare.

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

- să afișeze pe prima linie a ecranului, în ordinea inversă citirii, toate numerele din fișier, separate prin câte un spațiu;
- să afișeze pe a doua linie a ecranului, numărul de cifre din care este format fiecare număr din șirul inițial, numerele din linie separându-se prin câte un spațiu;
- să scrie în fișierul **atestat.out**, pe prima linie, suma tuturor numerelor impare din șir.

Not : Programul va conține cel puțin un subprogram util definit de utilizator.

Exemplu:

atestat.in		Date de ieșire:
6	a)	1001 242 2 71 555 13
13 555 71 2 242 1001	b)	2 3 2 1 3 4
		Fișierul atestat.out conține:
	c)	1640

Subiectul 8.

Fișierul **atestat.in** conține două linii. Pe prima linie este scris un număr natural nenul **n**, ($5 < n < 30$). Pe cea de-a doua linie a fișierului sunt scrise **n** numere întregi separate prin câte un spațiu, formate fiecare din cel mult 4 cifre, reprezentând un șir de **n** numere întregi. Șirul valorilor este **ordonat descrescător**.

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

- să afișeze pe prima linie a ecranului, în ordinea inversă citirii, toate numerele din fișier, separate prin câte un spațiu;
- să afișeze pe a doua linie a ecranului, mesajul “**DA**” sau “**NU**” în funcție de rezultatul citirii valorii **x**, citit de la tastatură, în șirul de valori din fișier;
- să scrie în fișierul **atestat.out**, pe prima linie, suma tuturor numerelor pozitive din șir.

Not : Programul va conține cel puțin un subprogram util definit de utilizator.

Exemplu:

atestat.in	Date de ieșire:						
6	a)	2	13	71	242	555	1001
1001 555 242 71 13 2	b)	DA					
Se citește x=555	c)	Fișierul atestat.out conține: 1884					

Subiectul 9.

Fișierul **atestat.in** conține pe prima linie numărul **n**, număr natural ($2 < n < 20$), ce reprezintă numărul de linii și coloane ale unei matrice pătratice **A**. Pe fiecare din următoarele **n** linii, sunt scrise câte **n** numere naturale, formate din exact 2 cifre fiecare, separate prin câte un spațiu, reprezentând valorile elementelor matricei pătratice **A**.

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

- să afișeze pe ecran elementele matricei **A**, linie cu linie, elementele de pe aceeași linie fiind separate prin câte un spațiu;
- să modifice matricea **A** astfel încât toate elementele situate pe diagonala secundară să aibă valoarea **k**, **k** citit de la tastatură, și să afișeze matricea rezultat pe ecran;
- să scrie în fișierul **atestat.out**, pe prima linie, separate prin câte un spațiu, toate numerele prime din matricea modificată. Dacă nu există se va scrie pe prima linie a fișierului mesajul “**NU**”.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu: unde $k=36$

atestat.in			Date de ieșire:			
3			a)	12	31	24
12	31	24		35	67	10
35	67	10		41	18	97
41	18	97	b)	12	31	36
				35	36	10
				36	18	97
				Fi ierul atestat.out con ine:		
			c)	31	97	

Subiectul 10.

Fi ierul **atestat.in** con ine pe prima linie dou valori n și m , numere naturale nenule ($2 < n, m < 20$), ce reprezint numărul de linii respectiv coloane ale unei matrice **A**. Pe fiecare din următoarele n linii, sunt scrise câte m numere naturale, separate prin câte un spațiu, reprezentând valorile elementelor matricei **A**. Pe ultima linie a fi ierului este o valoare k .

S se scrie un program, în limbajul *Pascal/C/C++*, care:

- s afi eze pe ecran elementele matricei **A**, linie cu linie, elementele de pe aceeași linie fiind separate prin câte un spațiu;
- s afi eze liniile din matrice care au k elemente nule, dac nu exist astfel de linii se va afi a mesajul "Nu exist";
- s scrie în fi ierul **atestat.out**, pe prima linie, separate prin câte un spațiu, toate numerele impare de pe coloana c a matricei, c va fi citit de la tastatură. Dac nu exist valori se va scrie pe prima linie a fi ierului mesajul "NU".

Not : Programul va con ine cel pu în un subprogram definit de utilizator.

Exemplu:

atestat.in			Date de ieșire:			
4	3		a)	0	0	24
0	0	24		0	67	10
0	67	10		41	0	0
41	0	0		212	95	101
212	95	101	b)	1	3	
2						
				Fi ierul atestat.out con ine:		
Se cite te $c=2$			c)	67	95	

Subiectul 11.

Fișierul **atestat.in** conține pe prima linie numărul **n**, natural nenul ($2 < n < 20$), ce reprezintă numărul de linii și coloane ale unei matrice pătratice **A**. Pe fiecare din următoarele **n** linii, sunt scrise câte **n** numere naturale, separate prin câte un spațiu, reprezentând valorile elementelor matricei pătratice **A**.

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

- să afișeze pe ecran elementele matricei **A**, linie cu linie, elementele de pe aceeași linie fiind separate prin câte un spațiu;
- să afișeze pe următoarea linie a ecranului, ultima cifră a produsului tuturor elementelor matricei **A** situate sub diagonala principală;
- să scrie în fișierul **atestat.out**, pe prima linie, separate prin câte un spațiu, toate numerele de tip *palindrom* din matricea **A**. Un număr este *palindrom* dacă numărul citit de la stânga la dreapta este egal cu numărul citit de la dreapta la stânga, de exemplu: **11, 22, 33**. Dacă nu există astfel de numere se va scrie pe prima linie a fișierului mesajul **"Nu exista"**.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in		Date de ieșire:
3	a)	15 20 37
15 20 37		10 21 44
10 21 44		12 40 15
12 40 15	b)	0
		Fișierul atestat.out conține:
	c)	44

Subiectul 12.

Fișierul **atestat.in** conține pe prima linie numărul **n**, natural nenul ($2 < n < 20$), ce reprezintă numărul de linii și coloane ale unei matrice pătratice **A**. Pe fiecare din următoarele **n** linii, sunt scrise câte **n** numere naturale, formate din cel mult **4** cifre fiecare, separate prin câte un spațiu, reprezentând valorile elementelor matricei pătratice **A**.

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

- să afișeze pe ecran elementele matricei **A**, linie cu linie, elementele de pe aceeași linie fiind separate prin câte un spațiu;
- să afișeze pe următoarea linie a ecranului suma elementelor matricei **A** situate pe diagonala secundară;
- să scrie în fișierul **atestat.out**, pe prima linie, separate prin câte un spațiu, minimele de pe fiecare linie a matricei.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in			Date de ieșire:			
3			a)	12	31	24
12	31	24		35	67	10
35	67	10		28	18	97
28	18	97	b)	119		
			c)	Fișierul atestat.out conține:		
				12	10	18

Subiectul 13.

Fișierul **atestat.in** conține pe prima linie numărul **n**, natural nenul ($2 < n < 10$), ce reprezintă numărul de linii și coloane ale unei matrice pătratice **A**. Pe fiecare din următoarele **n** linii, sunt scrise câte **n** numere întregi, formate din cel mult **9** cifre fiecare, separate prin câte un spațiu, reprezentând valorile elementelor matricei pătratice **A**.

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

- să afișeze pe ecran elementele matricei **A**, linie cu linie, elementele de pe aceeași linie fiind separate prin câte un spațiu;
- să citească de la tastatură un număr natural **k** de cel mult **9** cifre și să verifice dacă acesta se găsește în matricea **A**, afișându-se pe ecran un mesaj corespunzător;
- să modifice matricea **A** prin atribuirea valorii **0** tuturor elementelor situate pe ultima linie a matricei și apoi să se scrie în fișierul **atestat.out** matricea rezultat, linie cu linie, elementele de pe aceeași linie fiind separate prin câte un spațiu.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in			Date de ieșire:			
3			a)	-1	311	234
-1	311	234		13	-9	67
13	-9	67		5	-92	4
5	-92	4	b)	DA		
Se citește de la tastatură k = 13			c)	Fișierul atestat.out conține:		
				-1	311	234
				13	-9	67
				0	0	0

Subiectul 14.

Fișierul **atestat.in** conține pe prima linie numărul **n**, natural nenul ($2 < n < 10$), 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 afișeze pe ecran elementele matricei **A**, linie cu linie, elementele de pe aceeași linie fiind separate prin câte un spațiu;
- să afișeze pe următoarea linie a ecranului, media aritmetică a elementelor de pe diagonala principală;
- 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.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in		Date de ieșire:
3	a)	1.29 12.4 9.46
1.29 12.4 9.46		3.1 23.2 -5.6
3.1 23.2 -5.6		-9.9 8.5 -8.2
-9.9 8.5 -8.2	b)	5.39
Se citește de la tastatură k=2	c)	20.6
		Fișierul atestat.out conține:

Subiectul 15.

Fișierul **atestat.in** conține o singură linie pe care se află scris un text format din cel mult **250** de caractere, primul caracter fiind literă, iar cuvintele din text sunt separate prin câte un spațiu. Fiecare cuvânt este format doar din litere mari sau mici ale alfabetului limbii engleze.

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

- să afișeze textul pe o singură linie a ecranului, numai cu litere mari;
- să afișeze pe următoarea linie a ecranului numărul de consoane din text;
- să scrie în fișierul **atestat.out**, pe prima linie, numărul cuvintelor din text.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in	Creierul calculatorului este microprocesorul
Date de ieșire:	
- pe ecran :	a) CREIERUL CALCULATORULUI ESTE MICROPROCESORUL
	b) 22
- atestat.out	c) 4

Subiectul 16.

Fișierul **atestat.in** conține o singură linie pe care se află scris un text format din cel mult **100** de caractere, primul caracter fiind literă, iar cuvintele din text sunt separate prin câte un spațiu. Fiecare cuvânt este format doar din litere mari sau mici ale alfabetului limbii engleze.

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

- să afișeze textul pe o singură linie a ecranului, numai cu litere mari;
- să afișeze pe următoarea linie a ecranului numărul de vocale din text;
- să scrie în fișierul **atestat.out**, pe prima linie, numărul de cuvinte din textul inițial care au primul caracter 'a'.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in	Date de ieșire:
Se face sus aerisirea unei camere	a) SE FACE SUS AERISIREA UNEI CAMERE b) 16
	Fișierul atestat.out conține: c) 1

Subiectul 17.

Fișierul **atestat.in** conține o singură linie pe care se află scris un text format din cel mult **70** de caractere, primul caracter fiind literă, iar cuvintele din text sunt separate prin câte un spațiu. Fiecare cuvânt este format doar din litere mari ale alfabetului limbii engleze.

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

- să afișeze textul pe o singură linie a ecranului, numai cu litere mici;
- să afișeze pe următoarea linie a ecranului, numărul de cuvinte din care este format textul;
- să citească de la tastatură o literă, să modifice fiecare cuvânt care începe cu litera 'M' prin înlocuirea acesteia cu litera citită de la tastatură și să scrie în fișierul **atestat.out**, pe prima linie, textul modificat.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in	Date de ieșire:
MARIA ARE UN MAR	a) maria are un mar b) 4
Se citește de la tastatură litera 'D'	Fișierul atestat.out conține: c) DARIA ARE UN DAR

Subiectul 18.

Fișierul **atestat.in** conține o singură linie pe care se află scris un text format din cel mult **100** de caractere, primul caracter fiind literă, iar cuvintele din text sunt separate prin câte un spațiu. Fiecare cuvânt este format din litere mari sau mici ale alfabetului limbii engleze, sau din cifre.

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

- să afișeze textul pe prima linie a ecranului, numai cu litere mici;
- să afișeze pe ecran, pe linii separate, toate cuvintele din textul inițial;
- să scrie pe prima linie a fișierului **atestat.out** numărul cifrelor întâlnite în text.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in	In CLASA sunt 30 de elevi din CARE 10 Fete
Date de ieșire:	
a)	in clasa sunt 30 de elevi din care 10 fete
b)	in CLASA sunt 30 de elevi din CARE 10 Fete
atestat.out	c) 4

Subiectul 19.

Fișierul **atestat.in** conține o singură linie pe care se află scris un text format din cel mult **50** de caractere, primul și ultimul caracter fiind litere, iar cuvintele din text sunt separate prin câte un spațiu. Fiecare cuvânt este format doar din litere mari sau mici ale alfabetului limbii engleze.

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

- să afișeze textul pe ecran, pe o singură linie, numai cu litere mari;
- să afișeze pe următoarea linie a ecranului textul inițial, fără spații între cuvinte;
- să scrie în fișierul **atestat.out**, sub formă de triunghi toate secvențele mediane ale primului cuvânt din textul citit, pornind de la întregul cuvânt, până la secvența alcătuită din caracterul (sau cele 2 caractere) din mijloc.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in	Date de ieșire:
Hipopotamul este un MARE iubitor de apa	a) HIPOPOTAMUL ESTE UN MARE IUBITOR DE APA

b) HipopotamulesteunMAREiubitordeapa

Fișierul **atestat.out** conține:

c) Hipopotamul
 ipopotamu
 popotam
 opota
 pot
 o

Subiectul 20.

Fișierul **atestat.in** conține o singură linie pe care se află scris un text format din cel mult **100** de caractere, primul și ultimul caracter sunt litere, iar cuvintele din text sunt separate prin câte un spațiu. Fiecare cuvânt este format doar din litere mari sau mici ale alfabetului limbii engleze.

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

- să afișeze textul pe ecran, pe o singură linie, numai cu litere mari;
- să afișeze pe următoarea linie a ecranului textul inițial, fără spații între cuvinte;
- să scrie în fișierul **atestat.out**, perechile de cuvinte care sunt anagrame. Două cuvinte sunt anagrame dacă au lungime egală și sunt formate din aceleași litere eventual în altă ordine. Compararea celor două cuvinte se realizează fără a ține cont de litere mari sau mici ale alfabetului.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in	Date de ieșire:
Tamara are un verisor care pleaca in armata	a) TAMARA ARE UN VERISOR CARE PLEACA IN ARMATA b) Tamaraareunverisorcarepleacainarmata Fișierul atestat.out conține: c) Tamara armata

Subiectul 21.

Fișierul **atestat.in** conține o singură linie pe care se află scris un text format din cel mult **100** de caractere, primul și ultimul caracter sunt litere, iar cuvintele din text sunt separate prin câte un spațiu. Fiecare cuvânt este format doar din litere mari sau mici ale alfabetului limbii engleze.

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

- să afișeze textul pe ecran, pe o singură linie, numai cu litere mici;
- să afișeze pe următoarea linie a ecranului textul inițial, fără spații între cuvinte;

- f) să scrie în fișierul **atestat.out**, cuvintele care au proprietatea de palindrom. Un cuvânt este palindrom dacă citit de la stânga spre dreapta se obține același rezultat ca și citit de la dreapta spre stânga. Nu se face distincție între literele mari și mici.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in	Date de ieșire:
La TARG am gasit un cojoc cu MODELE geometrice	a) la targ am gasit un cojoc cu modele geometrice b) LaTARGamgasituncojoccuMODELEgeometrice
	Fișierul atestat.out conține: c) cojoc

Subiectul 22.

Fișierul **atestat.in** conține pe prima linie numărul natural nenul **n** ($2 < n < 10$). Pe următoarele **n** linii sunt scrise câte două numere naturale nenule **a** și **b**, reprezentând numărătorul (**a**) și numitorul (**b**) unei fracții algebrice. Cele două numere sunt separate printr-un spațiu.

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

- să afișeze pe ecran, pe aceeași linie, separate prin câte un spațiu, cele **n** fracții sub forma **a/b** (**numărător/numitor**);
- să afișeze pe următoarea linie a ecranului prima fracție cu numărătorul maxim;
- să scrie pe prima linie a fișierului **atestat.out**, separate prin spațiu, toate fracțiile ireductibile începând din a doua linie a fișierului **atestat.in**, sub forma **a/b**.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Se vor folosi structuri de date tip înregistrare.

Exemplu:

atestat.in	Date de ieșire:
6	a) 2/3 5/4 7/4 9/10 1/5 2/8
2 3	b) 9/10
5 4	Fișierul atestat.out va conține:
7 4	c) 2/3 5/4 7/4 9/10 1/5
9 10	
1 5	
2 8	

Subiectul 23.

Fișierul **atestat.in** conține pe prima linie numărul natural nenul **n** ($2 < n < 10$). Pe fiecare dintre următoarele **n** linii sunt scrise câte două numere naturale **x** și **y**, separate printr-un spațiu, reprezentând coordonatele unui punct din planul **xOy**.

Se scrie un program, în limbajul **Pascal/C/C++**, care:

- să afișeze pe ecran punctele din planul **xOy**, câte unul pe linie, sub forma **(x,y)**;
- să afișeze pe următoarea linie a ecranului numărul punctelor care sunt situate pe axa **Ox**;
- să scrie în fișierul **atestat.out**, pe linii distincte, toate punctele din fișierul **atestat.in** care au ambele coordonate pare, sub forma **(x,y)**.

Not : Programul va conține cel puțin un subprogram definit de utilizator.
Se vor folosi structuri de date tip înregistrare.

Exemplu:

atestat.in		Date de ieșire:	
8		a)	(0 , 0)
0	0		(1 , 2)
1	2		(3 , 0)
3	0		(2 , 2)
2	2		(2 , 3)
2	3		(0 , 4)
0	4		(6 , 8)
6	8		(5 , 0)
5	0	b)	Numarul punctelor situate pe axa Ox este 3
			Fișierul atestat.out va conține:
		c)	(0 , 0)
			(2 , 2)
			(0 , 4)
			(6 , 8)

Subiectul 24.

Fișierul **atestat.in** conține pe prima linie o valoare **n** număr natural ($1 < n < 10$) iar pe următoarele **n** linii câte un cuvânt, fiecare cuvânt fiind format din litere mari ale alfabetului englez.

Se scrie un program în limbajul **Pascal/C/C++**, care:

- să afișeze pe prima linie a ecranului, cuvintele numai cu litere mici;
- să afișeze pe ecran, cuvântul cu lungime maximă dintre cuvintele inițiale citite în fișier;
- să scrie pe prima linie a fișierului **atestat.out**, **n+1** caractere formate după regula: primul caracter din cuvânt este prima literă din primul cuvânt, al doilea caracter din cuvânt este a doua literă din al doilea cuvânt, al treilea caracter din cuvânt este a treia literă din al treilea cuvânt, etc. Ultimul caracter va fi '.' (punct). Dacă unul dintre cuvinte nu are suficiente litere, cuvântul rezultat va conține pe poziția corespunzătoare un spațiu.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in	5 ALMI COCOR MASA DO MARINA
Date de ieșire:	
	a) almi cocor masa do marina b) MARINA
atestat.out	c) AOS N.

Subiectul 25.

Fișierul **atestat.in** conține pe prima linie numărul n , natural nenul ($2 < n < 10$), ce reprezintă numărul de noduri ale unui graf neorientat. Pe următoarele linii până la sfârșitul fișierului sunt perechi de valori x y cu semnificația nodul x este legat printr-o muchie de nodul y .

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

- să afișeze pe ecran elementele matricei de adiacență atașată grafului citit;
- să citească de la tastatură un număr natural v reprezentând valoarea unui nod al grafului, și să afișeze pe ecran gradul nodului v ;
- să modifice matricea A astfel încât graful să devină complet și să afișeze în fișierul **atestat.out** matricea rezultat, linie cu linie, elementele de pe aceeași linie fiind separate prin câte un spațiu.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in		Date de ieșire:
4	a)	0 0 1 1
1 3		0 0 1 0
1 4		1 1 0 0
2 3		1 0 0 0
	b)	1
Se citește de la tastatură $v = 2$	c)	Fișierul atestat.out conține:
		0 1 1 1
		1 0 1 1
		1 1 0 1
		1 1 1 0

Subiectul 26.

Fișierul **atestat.in** conține pe prima linie numărul n , natural nenul ($2 < n < 10$), ce reprezintă numărul de noduri ale unui graf orientat. Pe următoarele linii până la sfârșitul fișierului sunt perechi de valori x y cu semnificația arcul (x, y) aparține grafului.

S se scrie un program, în limbajul *Pascal/C/C++*, care:

- a) să afișeze pe ecran elementele matricei de adiacență atașată grafului citit;
- b) să citească de la tastatură un număr natural v reprezentând valoarea unui nod al grafului, și să afișeze pe ecran gradul interior și gradul exterior al nodului v ;
- c) să afișeze în fișierul **atestat.out**, dacă există, toate drumurile de lungime **2** care pleacă din vârful v , v citit de la tastatură.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in		Date de ieșire:
5	a)	0 1 0 0 1
1 2		0 0 1 1 0
1 5		0 0 0 0 0
2 3		0 0 0 0 0
2 4		1 0 1 0 0
5 3		
5 1		
Se citește de la tastatură $v = 1$		b) 1 2
		c) Fișierul atestat.out conține:
		1 2 3
		1 2 4
		1 5 1

Subiectul 27.

Fișierul **atestat.in** conține pe prima linie numărul n , natural nenul ($2 < n < 10$), ce reprezintă numărul de noduri ale unui graf neorientat. Pe următoarele linii până la sfârșitul fișierului sunt perechi de valori $x y$ cu semnificația nodul x este legat printr-o muchie de nodul y .

S se scrie un program, în limbajul *Pascal/C/C++*, care:

- a) să afișeze pe ecran elementele matricei de adiacență atașată grafului citit;
- b) să afișeze pe ecran, dacă există, nodurile izolate ale grafului;
- c) să modifice matricea **A** astfel încât nodul v , valoarea v va fi citit de la tastatură, să fie adiacent cu toate celelalte noduri, iar matricea astfel modificată să afișeze în fișierul **atestat.out**.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in		Date de ieșire:
5	a)	0 0 1 1 0
1 3		0 0 1 0 0
1 4		1 1 0 0 0
2 3		1 0 0 0 0
		0 0 0 0 0

Se citește de la tastatură $v = 2$

b) 5

c) Fișierul **atestat.out** conține:

```
0 1 1 1 0
1 0 1 1 1
1 1 0 0 0
1 1 0 0 0
0 1 0 0 0
```

Subiectul 28.

Fișierul **atestat.in** conține pe prima linie numărul n , natural nenul ($2 < n < 10$), ce reprezintă numărul de noduri ale unui graf neorientat. Pe următoarele linii până la sfârșitul fișierului sunt perechi de valori x y cu semnificația nodul x este legat printr-o muchie de nodul y .

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

- să afișeze pe ecran elementele matricei de adiacență atașată grafului citit;
- să afișeze pe ecran, nodul sau nodurile care au gradul maxim;
- să modifice matricea **A** astfel încât nodul v , valoarea v va fi citită de la tastatură, să devină nod izolat, iar matricea astfel modificată să o afișeze în fișierul **atestat.out**.

Not : Programul va conține cel puțin un subprogram definit de utilizator.

Exemplu:

atestat.in		Date de ieșire:
5	a)	0 1 1 1 0
1 2		1 0 1 0 0
1 3		1 1 0 1 0
1 4		1 0 1 0 1
2 3		0 0 0 1 0
3 4		
4 5		
	b)	1 3 4
	c)	Fișierul atestat.out conține:
		0 0 1 1 0
		0 0 0 0 0
		1 0 0 1 0
		1 0 1 0 1
		0 0 0 1 0

Se citește de la tastatură $v = 2$

Subiectul 29.

Fișierul **atestat.in** conține pe prima linie numărul **n**, natural nenul ($2 < n < 10$), ce reprezintă numărul de noduri ale unui graf neorientat. Pe următoarele linii până la sfârșitul fișierului sunt perechi de valori **x y** cu semnificația nodul **x** este legat printr-o muchie de nodul **y**.

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

- să afișeze pe ecran elementele matricei de adiacență atașată grafului citit;
- să citească de la tastatură o secvență de noduri și să afișeze mesajul **"DA"** dacă secvența de noduri este lanț în graf și **"NU"** în caz contrar;
- să citească de la tastatură valoarea unui nod **v** și să afișeze în fișierul **atestat.out** care este lista de adiacență a nodului citit.

Not : Programul va conține cel puțin în un subprogram definit de utilizator.

Exemplu:

atestat.in			Date de ieșire:				
5		a)	0	1	1	1	0
1	2		1	0	1	0	0
1	3		1	1	0	1	0
1	4		1	0	1	0	1
2	3		0	0	0	1	0
3	4						
4	5						
Se citește de la tastatură 1 4 3 2			b)	DA			
v=3			c)	Fișierul atestat.out conține: 1 2 4			

Subiectul 30.

Fișierul **atestat.in** conține pe prima linie numărul **n**, natural nenul ($2 < n < 10$), ce reprezintă numărul de noduri ale unui graf orientat. Pe următoarele linii până la sfârșitul fișierului sunt perechi de valori **x y** cu semnificația arcul **(x, y)** aparține grafului.

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

- să afișeze pe ecran elementele matricei de adiacență atașată grafului citit;
- să afișeze pe ecran toate nodurile care au proprietatea că numărul arcelor ce pleacă din nodul respectiv este egal cu numărul arcelor care intră în nodul respectiv. (nodurile cu gradul exterior egal cu cel interior)
- să citească de la tastatură o secvență de noduri ale grafului și să afișeze în fișierul **atestat.out**, mesajul **"DA"** dacă secvența este drum în graf citit și **"NU"** în caz contrar.

Not : Programul va conține cel puțin în un subprogram definit de utilizator.

Exemplu:

atestat.in			Date de ieșire:				
5		a)	0	1	0	0	1

1	2	0	0	1	1	0
1	5	0	0	0	0	0
2	3	0	0	0	0	0
2	4	1	0	1	0	0
5	3					
5	1					

Se citește de la tastatură 1 2 3

b) 3

c) Fișierul `atestat.out` conține:
DA