Subiectul nr.1

Se citeşte de la tastatură un număr natural **n** (**1 ≤ n ≤ 1000**). Se cere:

a. Să se creeze un vector care să conţină primele n numere prime; elementele listei astfel obţinute se vor afişa pe ecran.
b. Se citesc două numere naturale a,b de la tastatură. Să se scrie numerele ce aparţin intervalului [a,b] şi vectorului, în fişierul date.out.

Exemplu:
a) Pentru **n=7**, se va afişa pe ecran lista:
```
2   3   5   7   11   13   17
```
b) Pentru **a=15**, **b=4** introduse de la tastatură, fişierul **date.out** va conţine:
```
5   7   11   13
```

Subiectul nr.2

Se dă un tablou unidimensional cu **n** (**1 ≤ n ≤ 100**) componente numere naturale.

a. Să se afişeze cifrele numărului celui mai mare care se poate obţine din cifrele elementului minim şi ale celui maxim din vector.
b. Să se afişeze câte numere din tablou sunt pătrate perfecte.

Datele de intrare se vor citi din fişierul **date.in** astfel:
- pe prima linie se găseşte n
- pe linia a doua se găsesc elementele tabloului.

Datele de ieşire se vor scrie pe ecran, astfel:
- pe prima linie, numărul obţinut la punctul a
- pe linia a doua, un mesaj în care să fie inclus numărul cerut la punctul b.

Exemplu:
Fişierul de intrare **date.in**:
```
6
271   109   28   713   14   36
```
Se afişează:
```
74311
Tabloul  are 1 patrat perfect
```

Subiectul nr. 3

Se citeşte de la tastatură un număr **n** (**0<n<100**) şi apoi un şir de **n** numere întregi.

a. Să se memoreze în fişierul date1.out doar numerele naturale ale şirului, în ordinea în care au fost introduse.
b. Să se insereze în faţa fiecărui număr de tip palindrom din fişierul date1.out succesorul său. Un număr este palindrom dacă citit de la stânga la dreapta este egal cu numărul citit de la dreapta la stânga.

Datele de ieşire se vor scrie în fişierul **date.out**, astfel:
- pe prima linie, elementele listei obţinute la punctul a)
- pe linia a doua, elementele listei obţinute după inserare.

Exemplu:

a.) Pentru **n=7** şi elementele **343 -73 12 34 767 454 -101** introduse de la tastatură, fişierul **date1.out** va conţine:
```
343   12   34   767   454
```
b.) Fişierul date.out va conţine:
```
344   343   12   34   768   767   455   454
```

Subiectul nr. 4

Se consideră un şir de **n** numere întregi distincte (**1 ≤ n ≤ 20**) şi două numere intregi **a** şi **b**.

a. Să se afişeze pe ecran toate elementele şirului care nu sunt în intervalul închis ce se poate forma cu numerele **a** şi **b**;
b. Să se ordoneze crescător numai elementele situate între elementul minim şi elementul maxim din şirul dat şi să se afişeze pe ecran şirul obţinut după ordonare.

Datele de intrare se vor citi din fişierul **date.in** astfel:

- pe prima linie se găseşte **n**
- pe linia a doua se găsesc cele **n** elemente ale şirului
- pe linia a treia se găsesc valorile lui **a**, respctiv **b**.

Exemplu:
Pentru fişierul **date.in**:

```
6
3  1  8  7  2  4  -1
5  3
```

Se vor afişa pe ecran şirurile

```
1   8   7   2   -1
3   1   8   2   4   7   -1
```

Subiectul nr. 5

Fie un şir de **n** (**0<n<1000**) numere întregi. Se cere:

a. Să se verifice dacă numerele din şirul dat pot forma o mulţime;
b. Să se afişeze cel mai mare divizor propriu al elementului maxim din şirul dat.

Datele de intrare se vor citi din fişierul **date.in** astfel:

- pe prima linie se găseşte **n**
- pe linia a doua se găsesc elementele şirului

Datele de ieşire se vor scrie pe ecran astfel:

- pe primul rând un mesaj
- pe rândul următor factorii primi separaţi prin spaţii

Exemplu:
Fisierul de intrare **date.in**:

```
5
80   64   50   96   45
```

Se afişează pe ecran:

```
Elementele din fişier pot forma o multime
48
```

Subiectul nr. 6

Se citeşte de la tastatură un număr natural **n** (**0<n<1000**). Se cere:

a. Să se verifice daca n este număr perfect. Un număr este perfect dacă este egal cu suma tuturor divizorilor lui, afară de numărul însuşi.
b. Să se creeze şi afişeze o matrice nxn, n<10, cu numere naturale astfel: pentru n=4, va fi:

```
1  2  3  4
5  6  7  8
9 10 11  12
```

Datele de ieşire se vor scrie în fişierul **date.out** astfel:

- pe prima linie, un mesaj: *n este/nu este numar perfect*

Exemplu:
Pentru **n=6**

Fişierul **date.out** poate să conţină:

6 este numar perfect

Subiectul nr. 7

Se dă un tablou bidimensional cu **n** linii şi **m** coloane ce conţine numere naturale nenule distincte.

a. Să se determine, dacă există, un număr superprim în matrice şi să se afişeze linia pe care se găseşte acesta. Un număr este superprim dacă atât el, cât şi răsturnatul lui sunt numere prime. Dacă există mai multe astfel de numere, se va afişa unul singur, oricare dintre ele.
b. Să se determine şi să se afişeze toate elementele vecine elementului maxim din matrice.

Datele de intrare se vor citi din fişierul **date.in** astfel:

- pe prima linie se găsesc **n** şi **m**
- pe următoarele **n** linii se găsesc câte **m** numere ce reprezintă elementele matricei.

Datele de ieşire se vor scrie pe ecran astfel:

- pe prima linie, numărul superprim şi indicele de linie sau un mesaj corespunzător, în cazul în care nu există nici un număr superprim
- pe linia a doua elementele cerute la punctul b separate printr-un spaţiu.

Exemplu:

Fişierul **date.in**:

```
4   3
33  65  105
44  32  22
62  76  41
31   1  12
```

Se afişează pe ecran:

```
31  4
65   32   22
```

Subiectul nr. 8

Pe prima linie a fişierului **date.in** se găsesc numere naturale nenule. Se cere:

a. pentru fiecare număr impar din fişier să se afişeze cel mai mare număr care se poate forma din cifrele lui.
b. să se verifice dacă numerele din fişierul de intrare reprezintă o progresie aritmetică; în caz afirmativ să se afişeze primul element şi raţia, altfel să se afişeze un mesaj corespunzător.

Datele de ieşire se vor scrie pe ecran, astfel:

- pe prima linie, elementele determinate la punctul a
- pe a doua linie, raspunsul cerintei b.

Exemplu:

Fişierul **date.in**:

```
31   38   45   52   59   66
```

Se afişează pe ecran:

```
31   38   54   52   95   66
primul termen este: 31 si raţia: 7
```

Subiectul nr. 9

Fie un tablou bidimensional (matrice) cu **n** linii şi **n** coloane. Se cere:

a. să se afişeze transpusa matricei;
b. să se afişeze câte perechi de numere prime între ele sunt pe diagonala secundară a matricei iniţiale.

Datele de intrare se vor citi din fişierul **date.in** astfel:

- pe prima linie se găseşte **n**
- pe următoarele **n** linii se găsesc câte **n** numere ce reprezintă elementele matricei.

Exemplu:

Fişierul de intrare:

```
4
 1   2   3   4
 5   6   7   8
 9  10  11  12
13  14  15  16
```

Se afişează:

a)
```
1    5    9   13
2    6   10   14
3    7   11   15
4    8   12   16
```

b)
```
5
```

Subiectul nr. 10

Fie un tablou bidimensional cu **n** linii şi **m** coloane. Se cere:

a. să se determine elementul maxim din matrice;
b. să se afişeze câte cifre distincte are acesta.

Datele de intrare se vor citi din fişierul **date.in** astfel:

- pe prima linie se găsesc **n** şi **m** separate printr-un spaţiu;
- pe următoarele **n** linii se găsesc câte **m** numere ce reprezintă elementele matricei.

Datele de ieşire se vor scrie pe ecran astfel:

- pe prima linie elementul maxim;
- pe linia următoare numărul de cifre distincte ale numărului maxim.

Exemplu:

Fişierul **date.in**:

```
3    5
 1   2   32   4   5
61  14  282   9  10
11  12   13  14  15
```

Dacă se citeşte de la tastatură **x=2**, atunci pe ecran se afişează

Se afişează pe ecran:

a.) **282**
b.) **2**

Subiectul nr. 11

În fişierul **date.in** se găsesc, pe o singură linie, prenumele unor elevi separate prin câte un spaţiu.

a. Să se ordoneze alfabetic (fără să se facă distincţie între litere mari şi mici) şi să se afişeze şirul de nume astfel obţinut.
b. Să se afişeze numarul de fete din fişier, ştiind că numai prenumele fetelor se termină cu litera „a”.

Datele de ieşire se vor scrie pe ecran, astfel:

- pe prima linie, prenumele ordonate
- pe a doua linie, numărul de fete.

Exemplu:

Fişierul **date.in**:

```
IOANA Marius Ciprian maria Silvia Gabriel Vlad
```

Se afişează pe ecran:
```
Ciprian Gabriel IOANA maria Marius Silvia Vlad
3
```

Subiectul nr. 12

În fişierul **date.in** se găsesc, pe o singură linie, numere strict pozitive separate prin câte un spaţiu.

a. Să se afişeze câte numere sunt în fişier;
b. Să se ordoneze crescător elementele fişierului, în funcţie de numărul de cifre pare ale fiecărui număr.

Datele de ieşire se vor scrie pe ecran, astfel:
- pe o linie, numărul determinat la punctul a
- pe linia următoare, numerele ordonate, separate prin câte un spaţiu.

Exemplu:
Fişierul date.in:
```
16    6    832    25    11    514
```
Se afişează pe ecran:
```
6
11   16    6    25    514    832
```

Subiectul nr. 13

În fişierul **date.in** se găsesc, pe o linie, numere naturale de cel mult 9 cifre, separate prin câte un spaţiu. Se cere:

a. să se determine numerele care reprezintă un factorial, adică sunt de forma k!; pentru fiecare astfel de număr se va afişa valoarea lui k
b. să se scrie, în ordine descrescătoare după prima cifră, toate numerele din fişier.

Datele de ieşire se vor scrie pe ecran, astfel:
- pe prima linie, numerele k determinate, separate prin spaţii;
- pe a doua linie numerele sortate descrescător după prima cifră, separate prin spaţii.

Exemplu:
Fişierul **date.in**:
```
233    24    210    6    17    720
```
Se afişează:
```
4    3    6
720   6   233   24   210   17
```

Subiectul nr. 14

În fişierul **date.in** se găsesc, pe o linie, numere naturale de cel mult 9 cifre, separate prin câte un spaţiu. Se cere:

a. să se determine şi să se afişeze numerele care reprezintă termeni ai şirului Fibonacci
b. să se afişeze, în ordine crescătoare după ultima cifră, toate numerele din fişier.

Datele de ieşire se vor scrie pe ecran, astfel:
- pe prima linie, numerele Fibonacci determinate, separate prin spaţii;
- pe a doua linie numerele sortate crescător după ultima cifră, separate prin spaţii.

Exemplu:
Fişierul **date.in**:
```
233    24    210    8    17    55
```
Se afişează:
```
233    8    55
210    233    24   55   17    8
```

Subiectul nr. 15

În fişierul **date.in** se găsesc pe o singură linie numere întregi nenule separate prin spaţii.

a. să se afişeze media aritmetică a valorilor din fişier;
b. să se creeze un vector care va conţine toate elementele, în ordinea în care apar în fişier; se vor afişa elementele listei, pornind de la cea mai mică valoare.

Datele de ieşire se vor scrie pe ecran, astfel:
- pe prima linie, media aritmetica, cu doua zecimale
- pe linia a doua, elementele listei, separate prin spaţii

Exemplu:
Fişierul **date.in**:
```
-6   18   26   -56   71   -24   13   -27   14   15
```
Pe ecran se vor afişa valorile:
a) 4.40
b) **-56 71 -24 13 -27 14 15 -6 18 26**

Subiectul nr. 16

Se dă un tablou bidimensional cu **n** linii şi **m** coloane şi componente numere reale.

a. Să se creeze fişierul date.out care să conţină elementele de pe marginile tabloului.
b. Să se verifice dacă matricea conţine cel puţin un punct de tip *şa.* Un punct este de tip *şa* dacă îndeplineşte următoarele doua condiţii: - este cea mai mare valoare de pe linia sa
- este cea mai mică valoare de pe coloana sa.

Datele de intrare se vor citi din fişierul **date.in** astfel:
- pe prima linie se găsesc **n** şi **m**
- pe următoarele **n** linii câte **m** numere ce vor reprezenta elementele matricei.

Datele de ieşire se vor scrie pe ecran, astfel:
- mesajul „Nu exista punct şa” sau „Punct şa” (linie, coloana).

Exemplu:
Fişierul **date.in**:
```
3  4
0.1   1.2   2.7   3.5
7.2   9.0   6.1   8.0
23.5  0.75  18.6  9.9
```
Fişierul `date.out` va conţine:
```
0.1  1.2  2.7  3.5  8.0  9.9  18.6  0.75  23.5  7.2
```
Şi se va afişa pe ecran:
```
Punct sa (1,4)
```

Subiectul nr. 17

În fişierul **date.in** se găsesc, pe aceeaşi linie, numere întregi separate prin spaţii.

a. Să se afişeze elementele din fişier situate pe poziţii pare (al doilea, al patrulea etc.).
b. Să se construiască o listă simplu înlănţuită care să conţină toate elementele şirului, în ordine inversă celei în care apar în fişier.

Datele de ieşire se vor scrie pe ecran, astfel:
- pe o linie, elementele, din fişier, situate pe poziţiile pare
- pe linia a doua, elementele listei.

Exemplu:

Fişierul **date.in**:

```
18   20   17   34   15   29   16   14   15   34
```

Se afişează pe ecran:

```
20   34   29   14   34
34   15   14   16   29   15   34   17   20   18
```

Subiectul nr. 18

Se citesc de la tastatură numere naturale până la introducerea numărului 0.

a. Să se memoreze într-un vector, toate numerele citite care au minim 3 cifre, în ordine inversă citirii;
b. Să se elimine din fiecare număr cifra sau cifrele din mijloc după cum numărul are un număr impar sau par de cifre.

Datele de ieşire se vor scrie în fişierul **date.out**, astfel:

- pe prima linie, elementele vectorului construit la cerinţa a
- pe linia a doua, elementele vectorului obţinut prin modificările realizate la cerinţa b.

Exemplu:

Pentru **1572 2351 1 519 73402 36 0** introduse de la tastatură

Fişierul de ieşire **date.out** va conţine:

```
73402   519   2351   1572
7302    59    21     12
```

Subiectul nr. 19

În fişierul **date.in** se găsesc cuvinte separate prin câte un spaţiu, cuvinte care sunt scrise cu litere mari şi mici. Se cere:

a. Să se afişeze cuvintele din fişier, în ordinea crescătoare a lungimii fiecărui cuvânt.
b. Să se memoreze fiecare cuvânt din fişier transformat în litere mari precum şi numărul de vocale ce apar în cuvântul respectiv; se vor lua în calcul doar vocalele *a e i o* şi *u*;

Datele de ieşire se vor scrie pe ecran astfel:

- pe un rând cuvintele în ordinea cerută la punctul a, separate prin spaţii
- pe rândul al doilea cuvintele şi numărul de vocale în formatul **(cuvânt, nr_de_vocale)**;

Exemplu:

Pentru fişierul **date.in**:

```
InformatICa   eLEV   sCOalA   CARTE
```

se vor afişa pe ecran liniile:

```
eLEV   CARTE   sCOalA   InformatICa
(INFORMATICA,5)   (ELEV,2)   (SCOALA,3)   (CARTE,2)
```

Subiectul nr. 20

Se consideră **n** intervale date. Se cere:

a. Să se determine intersecţia lor;
b. Să se afle câte intervale disjuncte sunt în raport cu intervalul [a,b], a şi b se citesc de la tastatură.

Datele de intrare se vor citi din fişierul **date.in** astfel:

- pe prima linie se găseşte **n**
- pe următoarele **n** linii, câte 2 numere naturale ce vor reprezenta limita inferioară şi respectiv cea superioară a fiecărui interval.

Datele de ieşire se vor scrie pe ecran astfel:

- pe prima linie două numere separate prin spaţiu ce vor reprezenta intersecţia intervalelor sau un mesaj, în cazul în care se obţine mulţimea vidă.
- pe linia a doua, numărul de intervale disjuncte obţinute prin reuniune

Exemplu:

Se citesc valorile a=2, b=9;
Fişierul **date.in**:

```
5
2   5
3   7
10   13
4   9
12   18
```

Se afişează pe ecran:

```
Intervalele nu se intersectează
2
```

Subiectul nr. 21

În fişierul **date.in** se găsesc, pe o singură linie, numere naturale nenule separate prin spaţii.

a. Să se memoreze într-un vector numerele din fişier şi, pentru fiecare, numărul de cifre impare ce apar în scrierea sa;
b. Să se afişeze cel mai mare dintre numerele care au un număr minim de cifre impare.

Datele de ieşire se vor scrie pe ecran astfel:

- pe un rând componentele vectorului în formatul **(nr, nr_de_cifre_impare)**;
- pe rândul al doilea numărul determinat conform cerinţei de la punctul b.

Exemplu:
Fişierul **date.in**:

```
72946    14    339   180   1027
```

Se afişează pe ecran:

```
(72946,2)    (14,1)    (339,3)   (180,1)    (1027,2)
180
```

Subiectul nr. 22

Un graf neorientat cu **n** noduri şi **m** muchii este descris în fişierul **graf.in** cu următoarea structură :

- pe prima linie valorile pentru **n** şi **m** separate printr-un spaţiu ;
- pe următoarele **m** linii, perechi de numere **i j** reprezentând noduri ce formează o muchie.

Să se realizeze un program care să răspundă următoarelor cerinţe:

a. Să se afişeze matricea de adiacenţă asociată grafului descris în fişierul **graf.in.**
b. Să se afişeze nodurile grafului în ordinea descrescătoare a gradelor.

Subiectul nr. 23

Un graf neorientat cu **n** noduri este descris în fişierul **graf.in** cu următoarea structură:

- pe prima linie, valoarea lui **n**;
- pe următoarele **n** linii, câte **n** valori 0 şi 1 reprezentând matricea de adiacenţă a grafului.

Să se realizeze un program care să raspundă următoarelor cerinte :

a. Să se afişeze numărul de muchii ale grafului descris în fişierul **graf.in** şi apoi perechile de numere reprezentând muchiile acestuia, în ordine lexicografică**.**
b. Să se afişeze numărul de noduri terminale.

Subiectul nr. 24

Un graf neorientat cu **n** noduri este descris în fişierul **graf.in** cu următoarea structură:

- pe prima linie valoarea lui **n**;
- pe următoarele linii, până la sfârşit de fişier, perechi de numere **i j** reprezentând noduri ce formează o muchie.

Să se realizeze un program care să răspundă următoarelor cerinţe:

a. Să se afişeze, de la **1** la **n**, gradele nodurilor grafului descris în fişierul **graf.in**.
b. Să se determine dacă două noduri **x** şi **y** sunt legate printr-un lanţ.

Subiectul nr. 25

Un graf orientat cu **n** noduri şi **m** arce este descris în fişierul **graf.in** cu următoarea structură:
- pe prima linie, valorile pentru **n** şi **m** separate printr-un spaţiu;
- pe următoarele **m** linii, perechi de valori **i j** reprezentând noduri ce formează un arc.

Să se realizeze un program care să raspundă următoarelor cerinţe:
a. Să se afişeze pe ecran perechile de noduri **i j** din graful descris în fişierul **graf.in** cu proprietatea că există arc atât de la **i** la **j**, cât şi de la **j** la **i**.
b. Să se afişeze muchiile incidente în vârfurile de grad maxim.