

Probleme ATESTAT Informatică 2021-2022
***Subiecte și rezolvări PROGRAMARE* ISJ Galați**

Subiectul nr 1

Se citesc de la tastatură mai multe numere naturale până când se întâlnește valoarea 0. Să se afișeze pe ecran câte dintre aceste numere sunt egale cu suma cuburilor cifrelor lor.

Exemplu: 153 10 1 34 5 0 se afișează 2 (sunt 2 numere: 153 1)

```
#include <iostream>
using namespace std;
int s,x,k;
int suma_cuburilor_cifrelor(int x)
{
    int s=0;
    while(x)
    {
        int u=x%10;
```

```
        s+=u*u*u;
        x/=10;
    }
    return s;
}
int main()
{
    cin>>x;
    while(x)
    {
        if(x==suma_cuburilor_cifrelor(x))
            k++;
        cin>>x;
    }
    cout<<k;
    return 0;
}
```

Subiectul nr 2

Se citește un număr de la tastatură de cel mult 9 cifre. Să se afișeze la ecran numărul obținut prin micșorarea fiecărei cifre nenule.

Exemplu: Dacă se citește numărul 1048 se va afișa 37

```
#include<iostream>
using namespace std;
int x, y, p;
int main()
{
```

```
    cin>>x;
    p=1;
    while(x!=0)
    {
        if(x%10!=0)
            y=y+(x%10-1)*p;
        else y=y+(x%10)*p;
        p=p*10;
        x=x/10;
    }
    cout<<y;
    return 0;
}
```

Subiectul nr 3

Se citește de la tastatură un text de maxim 50 caractere litere mici ale alfabetului englez. Să se afișeze textul obținut prin ștergerea primelor k vocale (k citit de la tastatură). Dacă în text nu există k vocale se vor șterge toate vocalele.

Exemplu: Dacă textul citit este *voi lua nota maxima la bacalaureat* și k=11 se va afișa *v l nt mxm l bcalaureat*

```
#include <iostream>
#include<cstring>
using namespace std;
char s[51], voc[]="aeiou";
int i, k;
int main()
{
    cin.get(s, 51);
```

```
    cin>>k;
    for(i=0;i<strlen(s) && k!=0;i++)
    if(strchr(voc,s[i])!=0)
    {
        strcpy(s+i, s+i+1);
        i--;
        k--;
    }
    cout<<s;
    return 0;
}
```



Subiectul nr 4

Se citesc de la tastatură trei numere naturale a, b și c cu maxim 9 cifre fiecare. Să se verifice dacă numerele a, b și c au exact aceiași factori în descompunerea lor în factori primi (exponenții lor putând să difere). Se va afișa pe ecran "DA" sau "NU".

Exemplu:

dacă a=300, b=1500, c=30 se afișează DA

dacă a=300, b=700, c=140 se afișează NU

```
#include <iostream>
using namespace std;
int fa[100],fb[100],fc
[100],a,b,c,nra,nrb,nrc;
void descompunere(int x,int v
[100],int &nr)
{
    int b=2,e=0;
    while(x!=1)
    {
        while(x%b==0)
```

```
        {
            x/=b; e++;
        }
        if(e)
            v[++nr]=b;
        b++; e=0;
    }
}

int main()
{
    cin>>a>>b>>c;
    descompunere(a,fa,nra);
    descompunere(b,fb,nrb);
    descompunere(c,fc,nrc);
    int ok=1;
    for(int i=1; i<=max(max
(nra,nrb),nrc); i++)
        if(!(fa[i]==fb[i]&&fa[i]
==fc[i])) ok=0;
    if(ok==1) cout<<"DA";
    else      cout<<"NU";
    return 0;
}
```

Subiectul nr 5

Prin asocierea fiecărei litere mici din alfabetul englez cu un număr egal cu poziția literei în alfabet, se poate codifica orice secvență de litere mici cu o secvență de numere. Se citește din fișierul text cod.txt un număr n ($0 < n < 30000$) și de pe rândul următor, o secvență de n litere mici. Scrieți un program care va codifica secvența de caractere citită și va scrie pe ecran numerele asociate, separate prin spații.

Exemplu: pentru fișierul cod.txt: 4; info

se vor afișa pe ecran, în ordine, numerele: 9 14 6 15

```
#include <iostream>
#include <fstream>
#include <cstring>
```

```
using namespace std;
ifstream f("cod.txt");
char s[201];
int n;
int main()
{
    f>>n;
    f.get();
    f.get(s,201);
    for(int i=0; i<strlen(s); i++)
        cout<<int(s[i]-'a'+1)<<"
";
    return 0;
}
```

Subiectul nr 6

Se citesc de la tastatură un număr n ($n \leq 100$) și apoi n elemente ale unui tablou unidimensional v. Să se afișeze pe ecran cea mai lungă secvență de elemente citite consecutiv cu proprietatea că fiecare componentă a succesiunii are același număr de cifre distincte în scrierea sa.

Exemplu: n=9 și elementele 55, 8, 37, 949, 26, 2213, 157, 38, 2 se afișează 37, 949, 26.

```
#include <iostream>
using namespace std;
int n,v[101],fr[10],fr2
[101],poz,poz2,poz1f,poz2f,maxi;
int main()
{
    cin>>n;
    for(int i=1; i<=n; i++)
    {
        cin>>v[i];
        int aux=v[i];
        for(int j=0; j<=9; j++) fr[j]
=0;
        while(aux)
        {
            fr[aux%10]++;
            aux/=10;
        }
    }
    for(int j=0; j<=9; j++) fr2[j]=1;
    for(int i=1; i<=n; i++)
    {
        if(fr2[i]==fr2[i+1])
        {
            poz=i;
            i++;
            while(fr2[i]==fr2[i+1])
            {
                poz2=i;
                if(poz2-poz+1>maxi)
                {
                    maxi=poz2-poz+1;
                    poz1f=poz;
                    poz2f=poz2;
                }
            }
        }
        for(int i=poz1f;
i<=poz2f; i++)
            cout<<v[i]<<" ";
        return 0;
    }
}
```

```
    }
    for(int j=0; j<=9; j++)
        if(fr[j]!=0) fr2[j]=1;
    }
    for(int i=1; i<=n; i++)
    {
        if(fr2[i]==fr2[i+1])
        {
            poz=i;
            i++;
            while(fr2[i]==fr2[i+1])
            {
                poz2=i;
                if(poz2-poz+1>maxi)
                {
                    maxi=poz2-poz+1;
                    poz1f=poz;
                    poz2f=poz2;
                }
            }
        }
        for(int i=poz1f;
i<=poz2f; i++)
            cout<<v[i]<<" ";
        return 0;
    }
}
```



Subiectul nr 7

Se citește un număr $n < 100$. Să se afișeze primele n numere cu următoarea proprietate: pătratul răsturnatului este egal cu răsturnatul pătratului.

Exemplu: dacă $n=12$ se va afișa 1 2 3 11 12 13 21 22 31 101 102 103.

```
#include <iostream>
using namespace std;
int n,x,k,inv, invp, pinv,
aux;
int main()
{
    cin>>n;
    x=1;
    while(k<n)
    {
        aux=x;
        inv=0;
        while(aux!=0)
        {
```

```
            inv=inv*10+aux%10;
            aux=aux/10;
        }
        pinv=inv*inv;
        invp=0;
        while(pinv!=0)
        {
            invp=invp*10+pinv%
10;
            pinv=pinv/10;
        }
        if(invp==x*x)
        {
            cout<<x<<" ";
            k++;
        }
        x++;
    }
    return 0;
}
```

Subiectul nr 8

Se citește din fișierul *date.in* o matrice A cu m linii și n coloane ($m \leq 20, n \leq 20$).

Să se construiască și să se afișeze în fișierul *date.out* matricea B obținută prin rotirea cu 90° în sens trigonometric a matricei A.

Exemplu: dacă $m=3$ și $n=5$ și matricea A este

$$\begin{pmatrix} 7 & 4 & 2 & 3 & 7 \\ 9 & 6 & 5 & 1 & 6 \\ 4 & 7 & 2 & 5 & 9 \end{pmatrix}$$

$$\begin{pmatrix} 7 & 6 & 9 \\ 3 & 1 & 5 \\ 2 & 5 & 2 \\ 4 & 6 & 7 \\ 7 & 9 & 4 \end{pmatrix}$$

matricea B obținută va fi

```
#include <fstream>
using namespace std;
ifstream f("date.in");
```

```
ofstream g("date.out");
int A[21][21], B[21][21], m, n, i, j;
int main()
{
    f>>m>>n;
    for(i=1; i<=m; i++)
        for(j=1; j<=n; j++)
            f>>A[i][j];
    for(i=1; i<=n; i++)
        for(j=1; j<=m; j++)
            B[i][j]=A[j][n+1-i];
    for(i=1; i<=n; i++)
    {
        for(j=1; j<=m; j++)
            g<<B[i][j]<<" ";
        g<<endl;
    }
    f.close();
    g.close();
    return 0;
}
```



Subiectul nr 9

Se citește un număr $n < 100$. Să se afișeze primele n numere prime palindrom.

Un număr este palindrom dacă citit de la stânga la dreapta sau de la dreapta la stânga este același.

Exemplu: dacă $n=12$ se va afișa 2, 3, 5, 7, 11, 101, 131, 151, 181, 191, 313, 353.

```
#include <iostream>
using namespace std;
int n, x, k, ok, d, aux, inv;
int main()
{
    cin >> n;
    x = 2;
    while (k < n)
    {
        ok = 1;
```

```
for (d = 2; d <= x / 2; d++)
    if (x % d == 0)
        ok = 0;
aux = x;
inv = 0;
while (aux != 0)
{
    inv = inv * 10 + aux % 10;
    aux = aux / 10;
}
if (ok == 1 && inv == x)
{
    k++;
    cout << x << " ";
}
x++;
}
return 0;
}
```

Subiectul nr 10

Se citește o matrice pătratică A de dimensiune n ($n \leq 20$).

Să se afișeze suma elementelor prime de pe diagonala secundară.

Exemplu: dacă matricea A este $n=5$ și

$$\begin{pmatrix} 2741 & 1199 & 27 & 13 & 5 \\ 22 & 6 & 7008 & 1 & 1999 \\ 45 & 46 & 202 & 201 & 33 \\ 1024 & 29 & 111 & 469 & 199 \\ 96 & 213 & 21 & 4096 & 16 \end{pmatrix}$$

se va

afișa 34

```
#include <iostream>
#include <cmath>
using namespace std;
int A[21][21], i, j, s, n, ok, d;
int main()
{
    cin >> n;
```

```
for (i = 1; i <= n; i++)
    for (j = 1; j <= n; j++)
    {
        cin >> A[i][j];
        if (i + j == n + 1)
        {
            ok = 1;
            for (d = 2; d <= sqrt(A[i][j]); d++)
            {
                if (A[i][j] % d == 0)
                    ok = 0;
                if (A[i][j] == 0 || A[i][j] == 1)
                    ok = 0;
            }
            if (ok == 1)
                s = s + A[i][j];
        }
    }
    cout << s;
    return 0;
}
```

Subiectul nr 11

Se citește un număr $n < 25$. Să se afișeze primele n numere cu următoarea proprietate: suma divizorilor fiecărui număr este un număr prim.

Exemplu: dacă $n=12$ se va afișa 1 2 4 9 16 25 64 289 729 1681 2401 3481.

```
#include <iostream>
using namespace std;
int n, x, k, suma, ok, d;
int main()
{
    cin >> n;
    x = 1;
    while (k < n)
    {
        suma = 0;
        for (d = 1; d <= x; d++)
```

```
if (x % d == 0)
    suma = suma + d;
ok = 1;
for (d = 2; d <= suma / 2; d++)
    if (suma % d == 0)
        ok = 0;
if (ok == 1)
{
    cout << x << " ";
    k++;
}
x++;
}
return 0;
}
```



Subiectul nr 12

Se citește un număr n ($n \leq 100$) și apoi n elemente ale unui tablou unidimensional v .

Să se elimine din tablou elementele care au suma cifrelor număr prim. Să se afișeze pe ecran tabloul rezultat.

Exemplu: dacă $n=12$ și elementele 34 1109 56 875 285 531 1190 11 98 287 444 41 se va afișa 875 285 531 444.

```
#include <iostream>
using namespace std;
int n, v[101], i, j, s, aux, ok, d;
int main()
{
    cin>>n;
    for(i=1; i<=n; i++)
        cin>>v[i];
    for(i=1; i<=n; i++)
    {
        s=0;
        aux=v[i];
```

```
while(aux!=0)
{
    s=s+aux%10;
    aux=aux/10;
}
ok=1;
for(d=2; d<=s/2; d++)
    if(s%d==0)
        ok=0;
if(s==1) ok=0;
if(ok==1)
{
    for(j=i; j<n; j++)
        v[j]=v[j+1];
        n--;
        i--;
}
for(i=1; i<=n; i++)
```

Subiectul nr 13

Se citește un număr n ($n \leq 100$) și apoi n elemente ale unui tablou unidimensional v .

Să se ordoneze descrescător după suma cifrelor elementele tabloului v și apoi să se afișeze tabloul ordonat.

Exemplu: dacă $n=12$ și elementele 34 1109 56 879 285 531 1190 17 98 287 444 41 se va afișa 879 98 287 285 1190 1109 56 531 441 17 34 41.

```
#include <iostream>
using namespace std;
int n, v[101], auxi, auxj, aux, i, j, si, sj;
int main()
{
    cin>>n;
    for(i=1; i<=n; i++)
        cin>>v[i];
```

```
for(i=1; i<n; i++)
    for(j=i+1; j<=n; j++)
    {
        auxi=v[i];
        auxj=v[j];
        si=0;
        sj=0;
        while(auxi!=0)
        {
            si=si+auxi%10;
            auxi=auxi/10;
        }
        while(auxj!=0)
        {
            sj=sj+auxj%10;
            auxj=auxj/10;
        }
        if(si<sj)
```

Subiectul nr 14

Se citește de la tastatură un număr natural nenul n , cu maximum 8 cifre.

Să se construiască în memorie și să se afișeze apoi pe ecran o matrice pătratică, având numărul de linii și de coloane egal cu numărul de cifre ale numărului dat, completată cu cifrele numărului citit. Astfel, elementele de pe ultima coloană a matricei vor fi toate egale cu cifra unităților numărului dat, elementele de pe penultima coloană a matricei vor fi toate egale cu cifra zecilor numărului dat, și așa mai departe, ca în exemplu.

Exemplu:

Dacă se citește 28463,

matricea construită va fi următoarea:

```
2 8 4 6 3
2 8 4 6 3
2 8 4 6 3
```

```
2 8 4 6 3
2 8 4 6 3
```

```
#include <iostream>
using namespace std;
int a[9][9];
int main()
{
    int n, c=0;
    cin>>n;
    int aux=n;
    while(aux)
    {
        aux/=10;
        c++;
    }
    for(int i=1; i<=c; i++)
    {
```



Subiectul nr 15

Se citesc din fișierul fib.in un număr natural n (n<=100) și apoi un șir format din n numere ordonate crescător.

Scrieți programul care determină și afișează în fișierul fib.out numărul de termeni ai șirului dat care au proprietatea că aparțin șirului lui Fibonacci.

Exemplu: pentru n=6 și numerele 2, 5, 10, 13, 13, 25

se afișează valoarea 4, deoarece termenii 2, 5, 13 și 13 fac parte din șirul lui Fibonacci.

```
#include <fstream>
using namespace std;
ifstream f("fib.in");
ofstream g("fib.out");
int n,a,b,c,nr;
int main()
{
    f>>n;
    a=0; b=1; c=1;
```

```
while (f>>n)
{
    if (n==a||n==c)
        nr++;
    else if ((n>c))
    {
        while (n>c)
        {
            a=b;
            b=c;
            c=a+b;
        }
        if (n==c)
            nr++;
    }
}
g<<nr;
return 0;
}
```

Subiectul nr 16

Se citesc două numere a și b (a, b cu maxim 9 cifre). Să se afișeze toate numerele cu proprietatea că au numărul de cifre pare egal cu numărul de cifre impare.

Exemplu: dacă a=94 și b=1014

se va afișa 94 96 98 1001 1003 1005 1007 1009 1010 1012 1014.

```
#include <iostream>
using namespace std;
int a, b, i, aux, nrp, nri;
int main()
{
    cin>>a>>b;
    for(i=a; i<=b; i++)
    {
```

```
        aux=i;
        nrp=0;
        nri=0;
        while(aux!=0)
        {
            if(aux%2==0)
                nrp++;
            else nri++;
            aux=aux/10;
        }
        if(nrp==nri)
            cout<<i<<" ";
    }
    return 0;
}
```

Subiectul nr 17

Se citește un număr de la tastatură de cel mult 9 cifre. Să se afișeze la ecran numărul obținut prin dublarea fiecărei cifre mai mică decât 5.

Exemplu:

dacă se citește numărul 275043 se va afișa 475086.

```
#include<iostream>
using namespace std;
int x, y, p;
int main()
```

```
{
    cin>>x;
    p=1;
    while(x!=0)
    {
        if(x%10<5)
            y=y+(x%10)*2*p;
        else y=y+(x%10)*p;
        p=p*10;
        x=x/10;
    }
    cout<<y;
    return 0;
}
```



Subiectul nr 18

Se citesc de la tastatură un număr n ($n \leq 100$) și apoi n elemente ale unui tablou unidimensional v . Să se insereze după fiecare număr prim exact două valori de 0 și să se afișeze pe ecran vectorul astfel modificat.

Exemplu: Pentru $n=7$ și elementele 55, 8, 37, 94, 101, 29, 15

se afișează 55, 8, 37, 0, 0, 94, 101, 0, 0, 29, 0, 0, 15

```
#include <iostream>
using namespace std;
int n, v[101];
bool ok;
int main()
{
    cin >> n;
    for (int i = 0; i < n; i++)
        cin >> v[i];
```

```
for (int i = 0; i < n; i++)
{ok = true;
    if (v[i] == 0 || v[i] == 1)
        ok = false;
    for (int d = 2; d <= v[i] / 2; d++)
        if (v[i] % d == 0)
            ok = false;
    if (ok)
    {for (int j = n + 1; j > i + 2; j--)
        v[j] = v[j - 2];
        v[i + 1] = v[i + 2] = 0;
        n += 2;
        i += 2;
    }
    for (int i = 0; i < n; i++)
        cout << v[i] << " ";
    return 0;
}
```

Subiectul nr 19

Se citesc de la tastatură două numere naturale a și b ($a \leq 1000, b \leq 1000$).

Să se afișeze pe ecran toate numerele din intervalul $[a, b]$ pentru care suma divizorilor lor este un număr prim.

Se va realiza o funcție care calculează și returnează suma divizorilor unui număr natural și o altă funcție care verifică dacă un număr este prim.

Exemplu: pentru $a=10, b=70$, se vor afișa 16, 25, 64

```
#include <iostream>
using namespace std;
int sumaDiv(int n)
{
    int s=0;
    for(int d=1; d<=n; d++)
        if(n%d==0)
            s=s+d;
    return s;
```

```
bool estePrim(int n)
{
    bool ok = true;
    if (n == 0 || n == 1)
        ok = false;
    for (int d = 2; d <= n / 2; d++)
        if (n % d == 0)
            ok = false;
    return ok;
}
int main()
{
    int a, b;
    cin >> a >> b;
    for(int i=a; i<=b; i++)
        if(estePrim(sumaDiv(i)))
            cout << i << " ";
    return 0;
}
```

Subiectul nr 20

Se citește de la tastatură o matrice A cu m linii și n coloane ($m \leq 20, n \leq 20$).

Să se verifice dacă există linii care au toate elementele palindromuri. Se va afișa indicele liniei/liniilor.

Exemplu:

Pentru $m=4, n=5$

și matricea

12	1	9	1	5
121	1	0	33	141
18	1	91	1	5
88	828	11	8	4

Se afișează 2, 4

```
#include <iostream>
using namespace std;
int n, i, a[21][21], j, k, m, ok, aux, p;
int main()
{
```

```
    cin >> m >> n;
    for(i=1; i<=m; i++)
        for(j=1; j<=n; j++)
            cin >> a[i][j];
    for(i=1; i<=m; i++)
    {
        ok=1;
        for(j=1; j<=n; j++)
        {
            aux=a[i][j];
            p=0;
            while(aux)
            {
                p=p*10+aux%10;
                aux/=10;
            }
            if(p!=a[i][j])
                ok=0;
        }
        if(ok==1) cout << i << " ";
    }
    return 0;
}
```



Subiectul nr 21

Se citește de la tastatură o matrice A cu m linii și n coloane ($m \leq 20, n \leq 20$) cu elemente cifre ale sistemului zecimal (0, 1, ..., 9). Se cere să se afișeze pe ecran perechile de linii complementare (adică suma elementelor aflate pe aceeași coloană trebuie să fie 9)

Exemplu:

Pentru $m=4, n=5$ și matricea A

```
1 1 9 1 5
1 1 0 3 1
1 1 9 1 5
8 8 0 8 4
```

Se afișează (1,4), (3,4)

```
#include <iostream>
using namespace std;
```

```
int n,i,a[21][21],j,k,m;
int main()
{
    cin>>m>>n;
    for(i=1; i<=m; i++)
        for(j=1; j<=n; j++)
            cin>>a[i][j];
    for(i=1; i<=m; i++)
        for(j=i+1; j<=m; j++)
        {
            int ok=1;
            for(k=1; k<=n; k++)
                if(a[i][k]+a[j][k]!=9)
                    ok=0;
            if(ok==1)
                cout<<"("<<i<<" "<<j<<"), ";
        }
    return 0;
}
```

Subiectul nr 22

Pe prima linie a fișierului text sir.in se găsește un cuvânt de exact 9 litere mici ale alfabetului englez. Pe a doua linie a fișierului se găsește o cifră nenulă k. Se cere să se afișeze pe ecran, de k ori, cel de al k-lea caracter de pe prima linie a fișierului sir.in.

Exemplu: dacă din fișier se citește

algorithmi

5

se va afișa pe ecran: rrrrr

```
#include <iostream>
#include <fstream>
using namespace std;
char s[10];
int k,i;
ifstream f ("sir.in");
int main()
{
    f>>s>>k;
    for(i=1; i<=k; i++)
        cout<<s[k-1];
    return 0;
}
```

Subiectul nr 23

Se citește de la tastatură un text de maxim 50 caractere litere mici ale alfabetului englez. Să se afișeze textul obținut prin dublarea primelor k vocale (k citit de la tastatură). Dacă în text nu există k vocale se vor dubla toate vocalele.

Exemplu:

dacă textul citit este *voi lua nota maxima la bacalaureat* și $k=11$ se va afișa *vooi luuaa nootaa maaxiimaa laa baacalaureat*.

```
#include <iostream>
#include <cstring>
using namespace std;
char s[101], voc[]="aeiou";
int i, k, j;
int main()
```

```
{
    cin.get(s, 51);
    cin>>k;
    for(i=0; i<strlen(s) && k!=0; i++)
        if(strchr(voc,s[i])!=0)
        {
            for(j=strlen(s); j>i; j--)
                s[j]=s[j-1];
            i++;
            k--;
        }
    cout<<s;
    return 0;
}
```



Subiectul nr 24

Se citesc de la tastatură două numere naturale a și b ($a \leq 1000, b \leq 1000$). Să se afișeze pe ecran toate numerele din intervalul [a,b] pentru care suma factorialilor cifrelor lor este mai mică sau egală cu 100.

Se va realiza o funcție care calculează și returnează factorialul unui număr natural și o altă funcție care returnează suma factorialilor cifrelor unui număr.

Exemplu:

pentru a=24, b=40, se vor afișa 24, 30, 31, 32, 33, 34, 40

```
#include <iostream>
using namespace std;
int fact(int c)
{
    int p=1;
    while(c>1)
    {
        p*=c;
        c--;
    }
}
```

```
}
return p;
}
int suma(int n)
{
    int s=0;
    while(n)
    {
        s+=fact(n%10); n/=10;
    }
    return s;
}
int main()
{
    int a, b;
    cin>>a>>b;
    for(int i=a; i<=b; i++)
        if(suma(i)<=100)
            cout<<i<<" ";
    return 0; }
```

Subiectul nr 25

Se citește o matrice pătratică A de dimensiune n ($n \leq 20$).

Să se afișeze la ecran matricea B formată din suma cifrelor fiecărui element din matricea A.

Exemplu: dacă matricea A este $n=5$ și

$$\begin{pmatrix} 2741 & 1199 & 27 & 13 & 5 \\ 22 & 6 & 7008 & 14 & 1999 \\ 45 & 46 & 202 & 201 & 33 \\ 1024 & 29 & 111 & 469 & 199 \\ 96 & 213 & 21 & 4096 & 16 \end{pmatrix}$$

se va afișa

$$\begin{pmatrix} 14 & 20 & 9 & 4 & 5 \\ 4 & 6 & 15 & 5 & 28 \\ 9 & 10 & 4 & 3 & 6 \\ 7 & 11 & 3 & 19 & 19 \\ 15 & 6 & 3 & 19 & 7 \end{pmatrix}$$

```
#include<iostream>
using namespace std;
int A[21][21],B[21][21], i, j, x, s, n;
int main()
{
    cin>>n;
    for(i=1; i<=n; i++)
        for(j=1; j<=n; j++)
        {
            cin>>A[i][j];
            s=0;
            x=A[i][j];
            while(x!=0)
            {
                s=s+x%10;
                x=x/10;
            }
            B[i][j]=s;
        }
    for(i=1; i<=n; i++)
    {
        for(j=1; j<=n; j++)
            cout<<B[i][j]<<" ";
        cout<<endl;
    }
}
```

Subiectul nr 26

Se citește o matrice pătratică A de dimensiune n ($n \leq 20$).

Să se afișeze la ecran matricea B formată din cifrele de control ale fiecărui element din matricea A.

Exemplu: dacă matricea A este $n=5$ și

$$\begin{pmatrix} 2741 & 1199 & 27 & 13 & 5 \\ 22 & 6 & 7008 & 14 & 1999 \\ 45 & 46 & 202 & 201 & 33 \\ 1024 & 29 & 111 & 469 & 199 \\ 96 & 213 & 21 & 4096 & 16 \end{pmatrix}$$

se va afișa

$$\begin{pmatrix} 5 & 2 & 9 & 4 & 5 \\ 4 & 6 & 6 & 5 & 1 \\ 9 & 1 & 4 & 3 & 6 \\ 7 & 2 & 3 & 1 & 1 \\ 6 & 6 & 3 & 1 & 7 \end{pmatrix}$$

```
#include<iostream>
using namespace std;
int A[21][21],B[21][21], i, j, x, s, n;
int main()
{
    cin>>n;
    for(i=1; i<=n; i++)
        for(j=1; j<=n; j++)
        {
            cin>>A[i][j];
            x=A[i][j];
            s=0;
            while(x>9)
            {
                while(x!=0)
                {
                    s=s+x%10; x=x/10;
                }
                x=s; s=0;
            }
            B[i][j]=x;
        }
    for(i=1; i<=n; i++)
    {
        for(j=1; j<=n; j++)
            cout<<B[i][j]<<" ";
        cout<<endl;
    }
}
```



Subiectul nr 27

Se citesc de la tastatură două șiruri de caractere cu maximum 250 de caractere fiecare (litere mici ale alfabetului englez), cu caracterele în ordine alfabetică, introduse de la tastatură. Se cere să se afișeze pe ecran un al treilea șir, format din toate caracterele primelor două, așezate în ordine alfabetică.

Exemplu: dacă se citesc `sir1="ampstz"` și `sir2="bfgostx"` se va afișa `abfgmopssttxz`.

```
#include <iostream>
#include <cstring>
using namespace std;
char s1[251], s2[251], s3[251], i, j;
int n1, n2, n3;
int main()
{
    cin >> s1 >> s2;
    n1 = strlen(s1); n2 = strlen(s2);
    while(i < n1 && j < n2)
```

```
{
    if(s1[i] < s2[j])
    {
        s3[n3] = s1[i];
        n3++; i++;
    }
    else
    {
        s3[n3] = s2[j];
        n3++; j++;
    }
}
while(i < n1)
{
    s3[n3] = s1[i];
    n3++; i++; }
while(j < n2)
{
    s3[n3] = s2[j];
    n3++; j++;
}
cout << s3;
return 0;
}
```

Subiectul nr 28

Se citește din fișierul *date.in* o matrice A cu m linii și n coloane ($m \leq 20$, $n \leq 20$). Să se construiască și să se afișeze în fișierul *date.out* matricea B obținută prin rotirea cu 90° în sensul acelor de ceasornic a matricei A.

Exemplu: dacă $m=3$ și $n=5$ și matricea A este

$$\begin{pmatrix} 7 & 4 & 2 & 3 & 7 \\ 9 & 6 & 5 & 1 & 6 \\ 4 & 7 & 2 & 5 & 9 \end{pmatrix}$$

matricea B obținută va fi

$$\begin{pmatrix} 4 & 9 & 7 \\ 7 & 6 & 4 \\ 2 & 5 & 2 \\ 5 & 1 & 3 \\ 9 & 6 & 7 \end{pmatrix}.$$

```
#include <fstream>
using namespace std;
ifstream f("date.in");
ofstream g("date.out");
int A[21][21], B[21][21], m, n, i, j;
int main()
{
    f >> m >> n;
    for(i=1; i<=m; i++)
        for(j=1; j<=n; j++)
            f >> A[i][j];
    for(i=1; i<=n; i++)
        for(j=1; j<=m; j++)
            B[i][j] = A[m+1-j][i];
    for(i=1; i<=n; i++)
    {
        for(j=1; j<=m; j++)
            g << B[i][j] << " ";
        g << endl;
    }
    return 0;
}
```

Subiectul nr 29

Se citește un număr de la tastatură de cel mult 9 cifre. Să se afișeze pe ecran cifrele care nu apar în numărul n.

Exemplu: $n=29225$ se afișează 0, 1, 3, 4, 6, 7, 8.

```
#include <iostream>
using namespace std;
int n, fr[11];
int main()
{
```

```
    cin >> n;
    while(n)
    {
        fr[n%10]++;
        n/=10;
    }
    for(int i=0; i<10; i++)
    {
        if(!fr[i])
            cout << i << " ";
    }
    return 0;
}
```



Subiectul nr 30

Se citesc de la tastatură un număr n ($n \leq 100$) și apoi n elemente ale unui tablou unidimensional v . Să se insereze înaintea fiecărui element pătrat perfect rădăcina sa pătrată și să se afișeze pe ecran vectorul astfel modificat.

Exemplu: $n=5$ și elementele 49, 64, 37, 94, 100 se afișează 7, 49, 8, 64, 37, 94, 10, 100

```
#include <iostream>
#include <cmath>
using namespace std;
int n,i,v[101],j;
int main()
{
    cin>>n;
    for(i=1; i<=n; i++)
```

```
        cin>>v[i];
    for(i=1; i<=n; i++)
    {
        if(sqrt(v[i])==int(sqrt(v
[i])))
        {
            n++;
            int p=i;
            for(j=n; j>p; j--)
                v[j]=v[j-1];
            v[p]=sqrt(v[i]);
            i++;
        }
    }
    for(i=1; i<=n; i++)
        cout<<v[i]<<" ";
    return 0;
}
```

Subiectul nr 31

Se citesc de la tastatură un număr n ($n \leq 100$) și apoi n elemente ale unui tablou unidimensional v . Să se afișeze pe ecran perechile de numere citite consecutiv cu proprietatea că cel de-al doilea număr al perechii reprezintă pătratul numărului de apariții ale cifrei 1 în primul.

Exemplu: $n=7$ și numerele 121, 4, 89, 7, 0, 111, 9 se afișează (121,4) (7,0) (111,9)

```
#include <iostream>
using namespace std;
int n,p,u,aux,c;
int main()
```

```
{
    cin>>n;    cin>>p;
    for(int i=2; i<=n; i++)
    {
        cin>>u;
        aux=p;
        c=0;
        while(aux)
        {
            if(aux%10==1) c++;
            aux/=10;
        }
        if(c*c==u) cout<<"("<<p<<","<<u<<") ";
        p=u;
    }
    return 0;
}
```

Subiectul nr 32

Se citesc de la tastatură două numere naturale a și b ($a \leq 1000, b \leq 1000$). Să se afișeze pe ecran toate numerele din intervalul $[a,b]$ care sunt prime cu suma cifrelor lor. Se va realiza o funcție care calculează și returnează suma cifrelor unui număr natural și o altă funcție care returnează c.m.m.d.c. a două numere naturale.

Exemplu: pentru $a=50, b=60$, se vor afișa 52, 53, 56, 58, 59.

```
#include <iostream>
using namespace std;
int a,b,i;
int suma(int x)
{
    int s=0;
    while(x)
    {
```

```
        s+=x%10;
        x/=10;
    }
    return s;
}
int cmmdc(int a,int b)
{
    if(a<b)
    {
        int aux=a;
        a=b;
        b=aux;
    }
    while(b!=0)
    {
        int c=a%b;
        a=b;
        b=c;
    }
    return a;
}
```



Subiectul nr 33

Se citește de la tastatură un text format din cuvinte separate între ele prin câte un spațiu. Fiecare cuvânt are cel mult 40 de caractere, doar litere mici ale alfabetului englez. Textul are cel mult 200 de caractere. Să se afișeze pe ecran, pe aceeași linie separate printr-un spațiu, doar cuvintele din textul citit care conțin cel mult trei vocale. Se consideră vocale: a, e, i, o, u.

Exemplu: dacă textul este: pentru examenul de atestat se folosesc tablouri

se afișează pe ecran: pentru de atestat se folosesc

```
#include <iostream>
#include <cstring>
using namespace std;
char s[201], *p, voc[]="aeiou";
int v;
int main()
{
```

```
    cin.get(s,201);
    p=strtok(s, " ");
    while(p)
    {
        v=0;
        bool ok=true;
        for(int i=0; i<strlen(p)
        &&ok==true; i++)
            if(strchr(voc,p[i])!=0)
            {
                v++;
                if(v>3) ok=false;
            }
        if(ok==true) cout<<p<<" ";
        p=strtok(NULL, " ");
    }
    return 0;
}
```

Subiectul nr 34

Se citește din fișierul **date.in** un număr natural **n** cu cel mult 10 cifre. Să se formeze un nou număr obținut prin eliminarea cifrei maxime din numărul inițial (dacă cifra maximă apare de mai multe ori, se vor elimina toate aparițiile) și să se afișeze pe ecran noul număr.

Exemplu: dacă $n=296929013$ se afișează 262013

dacă $n=333$ se afișează 0

```
#include <iostream>
#include <fstream>
using namespace std;
ifstream f("date.in");
int x;
int cifra_max(int x)
{
    int c=0;
    while(x)
    {
        if(x%10>c) c=x%10;
        x/=10;
    }
}
```

```

    }
    return c;
}
int sterge_cifra_max(int x,int c)
{
    int nr=0,p=1;
    while(x)
    {
        if(x%10!=c)
        {
            nr=nr+p*(x%10);
            p*=10;
        }
        x/=10;
    }
    x=nr;
    return x; }
int main()
{
    f>>x;
    int c=cifra_max(x);
    cout<<sterge_cifra_max(x,c);
    return 0;
}
```

Subiectul nr 35

Se citesc un număr n ($n \leq 100$) și apoi n elemente ale unui tablou unidimensional v . Să se elimine din tablou elementele care au cifra de control egală cu o valoare a . Să se afișeze pe ecran tabloul rezultat.

Exemplu: dacă $n=12$ și elementele 34 1109 56 875 285 531 1190 11 98 287 444 41 și $a=2$ se va afișa 34 285 531 98 287 444 41.

```
#include <iostream>
using namespace std;
int n, v[101], a, i, j, cc;
int main()
{
    cin>>n;
    for(i=1; i<=n; i++)
        cin>>v[i];
    cin>>a;
```

```
    for(i=1; i<=n; i++)
    {
        if(v[i]%9==0)
            cc=9;
        else cc=v[i]%9;
        if(cc==a)
        {
            for(j=i; j<n; j++)
                v[j]=v[j+1];
            n--;
            i--;
        }
    }
    for(i=1; i<=n; i++)
        cout<<v[i]<<" ";
    return 0;
}
```



Subiectul nr 36

Se citește un număr de la tastatură de cel mult 9 cifre. Să se afișeze pe ecran cel mai mare număr natural strict mai mic decât n care are toate cifrele egale între ele.

Exemplu: $n=356$ se afișează 333

```
#include <iostream>
using namespace std;
int n;
int main()
{
    cin>>n;
    int pc=n, nrc=1;
    while (pc>9)
    {
        nrc++; pc/=10;
    }
    int aux=n, c=0;
    while (aux)
```

```
{
    if (pc>aux%10)
    {
        pc--; aux=0;
    }
    else if (pc==aux%10) c++;
    aux/=10;
}
if (c==nrc) pc--;
if (pc==0)
{
    pc=9; nrc--;
}
if (nrc==0) cout<<"0";
while (nrc)
{
    cout<<pc;
    nrc--;
}
return 0;
}
```

Subiectul nr 37

Se citește un număr $n < 100$. Să se afișeze primele n numere cu următoarea proprietate: sunt fie puteri ale lui 2, fie puteri ale lui 3.

Exemplu: dacă $n=12$ se va afișa 1 2 3 4 8 9 16 27 32 64 81 128.

```
#include <iostream>
using namespace std;
int n, x, k, aux1, aux;
int main()
{
    cin>>n;
    x=1;
    while (k<n)
    {
```

```
        aux=x;
        while (aux%2==0)
            aux=aux/2;
        aux1=x;
        while (aux1%3==0)
            aux1=aux1/3;
        if (aux==1 || aux1==1)
        {
            cout<<x<<" ";
            k++;
        }
        x++;
    }
    return 0;
}
```



Subiectul nr 39

Se citește un număr $n < 100$. Să se afișeze primele n numere cu următoarea proprietate: la trecerea în baza 2 numărul valorilor de 1 este dublul valorilor de 0.

Exemplu: dacă $n=20$ se va afișa 2 4 5 6 8 9 10 11 12 13 14 16 17 18 19 20 21 22 23 24.

```
#include <iostream>
using namespace std;
int n,x,k,aux, unu, zero;
int main()
{   cin>>n;
    x=1;
    while(k<n)
    {   aux=x; unu=0; zero=0;
        while(aux!=0)
```

```
        {   if(aux%2==1) unu++;
            else zero++;
            aux=aux/2;
        }
        if(unu==2*zero)
        {   cout<<x<<" ";
            k++; }
        x++;
    }
    return 0; }
```

Subiectul nr 40

Se citesc un număr n ($n \leq 100$) și apoi n elemente ale unui tablou unidimensional v . Să se ordoneze crescător elementele impare, iar elementele pare trebuie să rămâne pe pozițiile inițiale. Să se afișeze tabloul ordonat.

Exemplu: dacă $n=12$ și elementele 34 1109 56 879 285 531 1190 17 98 287 444 41 se va afișa 34 17 56 41 285 287 1190 531 98 879 444 1109.

```
#include <iostream>
using namespace std;
int n, v[101], aux, i, j;
int main()
{   cin>>n;
    for(i=1; i<=n; i++) cin>>v[i];
```

```
    for(i=1; i<n; i++)
        for(j=i+1; j<=n; j++)
            if(v[i]%2==1
            && v[j]%2==1 && v[i]>v[j])
            {   aux=v[i]; v[i]=v[j]; v[j]=aux; }
    for(i=1; i<=n; i++)
        cout<<v[i]<<" ";
    return 0;
}
```

