

Cifrele unui număr

```
cin>>x;
if(x==0)
    {prelucrează x dacă este cazul deoarece este caz particular}
while(x!=0)
    {prelucrează x%10;
    x=x/10;
    }
```

sau

```
cin>>x;
do
    { prelucrează x%10;
      x=x/10;
    }
while(x!=0);
```

Dacă trebuie să prelucrăm cifrele câte două, fiecare cu cea din stânga ei, numărul trebuie să aibă cel puțin două cifre:

```
if(x<10)
    {prelucrează cazul particular în care x are o cifră}
while(x>=10)
    {prelucrează x%10 și x/10%10 ;
    x=x/10;
    }
```

Dacă trebuie să prelucrăm fiecare cifră împreună cu cifra situată cu două poziții mai la stânga vom scrie while(x>=100).....și vom prelucra x%10 împreună cu (x/100)%10

Fie x un număr natural.

- *numărul cifrelor*

```
cin>>x;
nr=0;
if(x==0) nr=1;
while(x!=0)
    {nr++;
    x=x/10;
    }
cout<<nr;
```

- *numărul cifrelor pare*

```
cin>>x;
nr=0;
if(x==0) nr=1; // 0 este cifră pară
while(x!=0)
    {if(x%2==0)
        nr++; //un număr par are ultima cifră pară
    x=x/10;
    }
cout<<nr;
```

- *suma cifrelor*

```
cin>>x;
s=0;
while(x!=0)
    {s=s+x%10
    x=x/10;
    }
cout<<s;
```

- *suma cifrelor impare*

```
cin>>x;
s=0;
while(x!=0)
    {if(x%2!=0) s=s+x%10;
    x=x/10;
    }
cout<<s;
```

- *produsul cifrelor*

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

- *produsul cifrelor impare sau -1 dacă nu are cifre impare*

```
cin>>x;
p=1;nr=0;
while(x!=0)
    {if(x%2!=0)
        {p=p*(x%10);nr=1;}
    x=x/10;
    }
if(nr==0) p=-1;
cout<<p;
```

- *cifra maximă*

```
cin>>x;
maxi=0;
while(x!=0)
    {if(x%10>maxi)
        maxi=x%10;
    x=x/10;
    }
cout<<maxi;
```

- *cea mai mare cifră pară sau -1 dacă nu există*

```
cin>>x;
maxi=-1;
while(x!=0)
    { if(x%2==0)
      if(x%10>maxi)
        maxi=x%10;
      x=x/10;
    }
cout<<maxi;
```

- *cifra minimă*

```
cin>>x;
mini=9;
while(x!=0)
    { if(x%10<mini)
      mini=x%10;
      x=x/10;
    }
cout<<mini;
```

- *cea mai mică cifră pară sau -1 dacă nu există*

```
cin>>x;
mini=10;
while(x!=0)
    { if(x%2==0)
      if(x%10<mini)
        mini=x%10;
      x=x/10;
    }
if(mini==10) mini=-1;
cout<<mini;
```

- *cifra maximă și numărul de apariții*

```
cin>>x;
maxi=0; nr=0;
if(x==0) nr=1;
while(x!=0)
    { if(x%10>maxi)
      { maxi=x%10;nr=1;}
    else
      if(x%10==maxi) nr++;
      x=x/10;
    }
cout<<maxi<< ' ' <<nr;;
```

- *cele mai mari două cifre distincte dacă există sau mesajul "nu există"*

```
cin>>x;
max1=-1;max2=-1;
while(x!=0)
    {if(x%10>max1) {max2=max1; max1=x%10;}
    else if(x%10>max2 && x%10!=max1) max2=x;
    x=x/10;
    }
if(max1!=-1 && max2!=-1)cout<<max1<<' '<<max2;
else cout<<"nu exista";
```

- *suma cifrelor situate pe poziții pare(de rang par) dacă pozițiile se numerează de la dreapta la stânga pornind de la 0*

```
cin>>x;
s=0; poz=0;
while(x!=0)
    {if(poz%2==0) s=s+x%10;
    x=x/10;
    poz++;
    }
cout<<s;
```

sau

```
cin>>x;
s=0;
while(x!=0)
    {s=s+x%10;
    x=x/100;      //se elimină câte două cifre din număr
    }
cout<<s;
```

- *oglinditul(inversatul/răsturnatul)*

```
cin>>x;
o=0;
while(x!=0)
    {o=o*10+x%10;
    x=x/10;
    }
cout<<o;
```

- *test de palindrom*

```
cin>>x;
o=0; aux=x;
while(aux!=0)
    {o=o*10+aux%10;
    aux=aux/10;
    }
if(x==o) cout<<"este palindrom";
else cout<<"nu este palindrom";
```

- *test de număr dublu palindrom(este palindrom și are cifrele egale câte două (ex. 1133443311))*

```
cin>>x;
o=0; aux=x;
while(aux!=0)
    {o=o*10+aux%10;
    aux=aux/10;
    }
aux=x;
ok=1;
while(aux>=10)
{if(aux%10!=aux/10%10)
    ok=0;
aux=aux/100;
}
if(x==o &&ok==1&&x>=10)
cout<<"este dublu palindrom";
else cout<<"nu este dublu palindrom";
```

- *Formare de număr nou doar din anumite cifre/prin eliminarea cifrelor care nu interesează*
- *numărul format doar din cifrele pare în ordinea din numărul inițial(ex. 1234546 se formează 2446) sau -1 dacă nu are cifre pare*

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

- *test de număr rotund(trecut în baza 2 numărul de cifre 0 este egal cu numărul de cifre 1)*

```
cin>>x;
nr0=0;nr1=0;
if(x==0) nr0++;
while(x!=0)
    {if(x%2==0)
        nr0++;
        else nr1++;
        x=x/2;
    }
if(nr0==nr1) cout<<"este rotund";
else cout<<"nu este rotund";
```

- *trecerea numărului într-o bază $b < 10$*

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

- *trecerea numărului din baza $b < 10$ în baza 10*

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

- *verificare dacă numărul este scris în baza $b \leq 10$ (toate cifrele să fie mai mici decât b sau cifra maximă să fie mai mică decât b)*

```
cin>>x>>b;
ok=1;
while(x!=0)
{ if(x%10>=b) ok=0;
  x=x/10;
}
if(ok==1)cout<<"da";
else cout<<"nu";
```

- *verificare dacă are cifrele identice*

```
cin>>x;
ok=1;
while(x>=10)
{ if(x%10!=x/10%10) ok=0;
  x=x/10;
}
if(ok==1)cout<<"da";
else cout<<"nu";
```

- *verificare dacă are cifrele în ordine crescătoare de la dreapta la stânga*

```
cin>>x;
ok=1;
while(x>=10)
{ if(x%10>=x/10%10) ok=0;
  x=x/10;
}
if(ok==1)cout<<"da";
else cout<<"nu";
```

- *verificați dacă are aspect de deal(munte)(ex.1356742)*

```
cin>>x;
ok=1;
while(x>=10 && x%10<(x/10)%10)
    x=x/10;
if(x<10) ok=0; //are doar o panta
else
    { while(x>=10 && x%10>(x/10)%10)
        x=x/10;
      if(x>=10) ok=0; //mai are cifre care nu sunt in ordinea dorita
    }
if(ok==1)cout<<"da";
else cout<<"nu";
```

- *verificați dacă are aspect de vale(ex. 5314)*

```
cin>>x;
ok=1;
while(x>=10 && x%10>(x/10)%10)
    x=x/10;
if(x<=10) ok=0; //are doar o panta
else
    { while(x>=10 && x%10<(x/10)%10)
        x=x/10;
      if(x>=10) ok=0; //mai are cifre care nu sunt in ordinea dorita
    }
if(ok==1)cout<<"da";
else cout<<"nu";
```

- *verificați dacă are aspect de fierăstrău(ex.14141, trebuie să aibă cel puțin 3 cifre pentru a le putea compara)*

```
cin>>x;
ok=1;
while(x>=100)
    { if(x%10!=x/100%10) ok=0;
      x=x/10;
    }
if(ok==1)cout<<"da";
else cout<<"nu";
```

- *frecvența cifrelor*

```
cin>>x;
for(int c=0;c<=9;c++)
    fr[c]=0;
if(x==0) fr[0]=1;
while(x!=0)
    { fr[x%10]++;
      x=x/10;
    }
for(c=0;c<=9;c++)
    cout<<c<<" cu frecventa "<<fr[c];
```

- *cel mai mare număr care se poate forma cu cifrele numărului dat(vezi vector de frecvență)*

```
cin>>x;
for(int c=0;c<=9;c++)
    fr[c]=0;
if(x==0) fr[0]=1;
while(x!=0)
    { fr[x%10]++;
      x=x/10;
    }
y=0;
for(c=9;c>=0;c--)
    for(int i=1;i<=fr[c];i++)
        y=y*10+c;
cout<<y;
```

- *cel mai mare număr care se poate forma cu cifrele distincte ale numărului dat*

```
cin>>x;
for(int c=0;c<=9;c++)
    fr[c]=0;
if(x==0) fr[0]=1;
while(x!=0)
    { fr[x%10]++;
      x=x/10;
    }
y=0;
for(c=9;c>=0;c--)
    if(fr[c]!=0)
        y=y*10+c;
cout<<y;
```

- *cel mai mic număr care se poate forma cu cifrele numărului dat*

```
cin>>x;
for(int c=0;c<=9;c++)
    fr[c]=0;
if(x==0) fr[0]=1;
while(x!=0)
    { fr[x%10]++;
      x=x/10;
    }
y=0;
if(fr[0]!=0) //se cauta prima cifra diferita de 0
    { c=1;
      while(c<=9&&fr[c]==0) c++;
      if(c<=9)
          {y=c; fr[c]--;}
    }
for(c=0;c<=9;c++)
    for(int i=1;i<=fr[c];i++)
        y=y*10+c;
cout<<y;
```


- cel mai mic număr care se poate forma cu cifrele distincte ale numărului dat

```
cin>>x;
for(int c=0;c<=9;c++)
    fr[c]=0;
if(x==0) fr[0]=1;
while(x!=0)
    { fr[x%10]++;
      x=x/10;
    }
y=0;
if(fr[0]!=0) //se cauta prima cifra diferita de 0
    { c=1;
      while(c<=9&&fr[c]==0) c++;
      if(c<=9)
          { y=c; fr[c]=0; }
    }
for(c=0;c<=9;c++)
    if(fr[c]!=0)
        y=y*10+c;
cout<<y;
```

- dacă numărul are 9 cifre distincte nenule construiți și afișați, dacă există numărul imediat mai mare(ex. 326978541 se va obține 326981457) altfel afișați mesajul nu există

```
cin>>x;
for(int c=0;c<=9;c++)
    fr[c]=0;
while(x>=10 && x%10<x/10%10)
    { fr[x%10]++;
      x=x/10;
    }
if(x>=10)
    { u=x%10;
      c=(x/10)%10;
      fr[c]++;
      x=x/100;
      x=x*10+u;
      for(c=1;c<=9;c++)
          if(fr[c]!=0) x=x*10+c;
      cout<<x;
    }
else cout<<"nu exista";
```

//sau pentru că are fix 9 cifre if(x==987654321) cout<<"nu există";

- formați un număr nou prin rotirea circulară a cifrelor cu o poziție la stânga

```
cin>>x;
y=0; p=1;
while(x>=10)
    { y=y+(x%10)*p;
      p=p*10;
      x=x/10;
    }
```

```
y=y*10+x;  
cout<<y;
```

- *formați un număr nou prin rotirea circulară a cifrelor cu o poziție la dreapta*

```
cin>>x;  
u=x%10;x=x/10;  
aux=x;  
p=1;  
while(aux!=0)  
{ p=p*10;  
  aux=aux/10;  
}  
y=u*p+x;  
cout<<y;
```

- *cifra de control*

- cu două structuri repetitive

```
cin>>x;  
while(x>=10)  
{ s=0;  
  while(x!=0)  
  {  
    s=s+x%10  
    x=x/10;  
  }  
  x=s;  
}  
cout<<x;
```

- cu o singură structură repetitivă

```
cin>>x;  
s=0;  
while(x>=10)  
{ s=s+x%10  
  x=x/10;  
  if(x<10) { x=x+s;s=0;}  
}  
cout<<x;
```

- fără structuri repetitive

```
cin>>x;  
if(x==0) cifra=0;  
else  
  if(x%9==0) cifra=9;  
  else cifra=x%9;  
cout<<cifra;
```

- dacă se citesc $n \leq 9$ cifre, formați un număr doar din cifrele pare sau -1 dacă nu are cifre pare

```
cin>>n;
x=0;
for(int i=1;i<=n;i++)
{ cin>>c;
if(c%2==0)
{ x=x*10+c;
nr++;
}
}
if(nr==0) x=-1;
cout<<x;
```

Converșii între numere naturale și numere reale

Se dă un număr real x. Formați numere naturale din

- partea întreagă a lui x

#include<cmath> //ceil(x) partea întreaga prin adaus

```
cin>>x;
```

```
y=floor(x); //sau y=(int)x;
```

în variantele mai noi dacă x este real și y este întreg merge și direct y=x deoarece face conversia automat

- partea zecimală a lui x transformată în număr întreg(ex. 23,456 se formează 456)

```
cin>>x;
z=x-floor(x); //sau z=x-(int)x; //partea zecimală
while(z!=floor(z)) //se transforma in numar intreg
z=z*10;
```

- suma numerelor care reprezintă partea întreagă și partea zecimală(34.123 se formează și se adună 34 cu 123)

```
cin>>x;
y=floor(x);
z=x-floor(x);
while(z!=floor(z))
z=z*10;
cout<<y+z;
```

- Transformați un număr natural dat în numărul real subunitar corespunzător(ex. 453 în 0.453)

```
cin>>x;
float y=x;
while(y>=1)
y=y/10;
```

- Se dau două numere naturale, formați un număr nou lipind la sfârșitul numărului mai mare pe cel mai mic.

```
cin>>a>>b;
if(a<b){x=b;y=a;}
else {x=a;y=b;}
if(y==0)x=x*10;
aux=y;
while(aux!=0)
{ aux=aux/10;
x=x*10;
}
x=x+y;
cout<<x;
```

Afișare cu număr fix de zecimale a unui nr. real fără rotunjire(ex. 2 zecimale)

```
#include <iostream>
#include <iomanip>
using namespace std;
int main()
{ float x=3.346;
  x=x*100;
  x=(int)x;
  x=x/100;
  cout<<fixed<<setprecision(2)<<x;
  return 0;
}
```