

Algoritmi elementari

(a,b)=cmmdc(a,b)

$$[a,b]=\text{cmmmc}(a,b)=\frac{a*b}{\text{cmmdc}(a,b)}$$

cmmdc(a,b)-prin scăderi succesive

```
while(a!=b)
  if(a>b)
    a=a-b;
  else
    b=b-a;
cmmdc=a;
```

cmmdc(a,b)-prin împărțiri succesive

```
r=a%b;
while(r!=0)
{
  a=b;
  b=r;
  r=a%b;
}
cmmdc=b;
```

- Se citesc 2 numere naturale nenule a și b. Se cere să se afișeze cmmdc-ul și cmmmc-ul celor 2 numere.
- Se citesc n numere naturale nenule, cu cel mult 4 cifre fiecare unde $2 \leq n \leq 1000$. Se cere:
 - Cmmdc-ul celor n numere
 - Cmmmc-ul celor mai mari 2 numere
 - Cmmdc-ul celor mai mici 2 numere pare (dacă există)
 - Să se verifice dacă există perechi de numere citite consecutiv, prime între ele
 - Să se afișeze fracția subunitară ireductibilă care se poate forma din numărul minim și numărul maxim din șir. Exp: $n=5, a=\{3,2,5,8,4\} \Rightarrow \frac{1}{4}$
- Se citesc numere naturale nenule până la întâlnirea unui număr par. Se cere:
 - Tripletele (a,b,c) de numere citite consecutiv cu proprietatea că $\text{cmmmc}(a,b,c)=a$
 - Numărul tripletele (a,b,c) de numere citite consecutiv cu proprietatea că $b=\text{cmmdc}(a,c)$
 - Numărul perechilor (a,b) de numere citite consecutiv cu proprietatea că $(a,b)=\text{primul număr citit}$
- Se citesc de la tastatură două numere naturale n și k ($2 \leq k \leq n$). Să se afișeze toate perechile de numere naturale mai mici decât n al căror c.m.m.d.c. este k.
- Să se afișeze toate numerele naturale, mai mici decât un număr natural n, care sunt prime cu n.
Ex: $n=12 \Rightarrow \{1, 5, 7, 11\}$

Cifrele unui număr natural

Numărul cifrelor unui număr natural n

```
if(n==0) nrc=1;
else{
  nrc=0;
  while(n>0){ nrc++; n=n/10; }
}
```

Suma cifrelor numărului natural n

```
s=0;
while(n>0)
{
  s=s+n%10;
  n=n/10;
}
```

Determinarea primei cifre a numărului natural n

```
c=n;
while(c>9)
{
  c=c/10;
}
```

Numărul cifrelor unui număr natural n

```
if(n==0) nrc=1;
else nrc=log10(n)+1;
```

Oglinditul numărului n

```
o=0;
while(n>0)
{
  o=o*10+n%10;
  n=n/10;
}
```

Determinarea cifrei maxime

```
cmax=0;
while(n>0)
{
  if(cmax<n%10) cmax=n%10;
  n=n/10;
}
```

- Se citeste un număr natural cu cel puțin două cifre și cel mult 9 cifre. Să se calculeze cel mai mic multiplu par al numărului obținut din prima și ultima cifră a numărului citit.

- Ex: pentru 3261 se obține 62
2. Se citește un număr natural n cu cel mult 9 cifre. Să se determine cea mai mare cifră a lui n și de câte ori apare ea în scrierea numărului n .
Ex: pentru $n=5371277$ se va afișa 7 3
 3. Se citește un număr natural n cu maxim 9 cifre. Să se verifice dacă el are toate cifrele ordonate strict descrescător de la cifra cea mai semnificativă spre cifra unităților.
Ex: Pentru 54321 se va afișa DA, iar pentru 543234 se va afișa NU.
 4. Se citesc de la tastatură cele n elemente ale unui șir de numere întregi. Să se afișeze toate perechile de elemente ale șirului (consecutive) cu proprietatea că ambele elementele ale perechii au aceeași sumă a cifrelor.
 5. Se citesc 2 numere naturale n și k cu maxim 7 cifre fiecare. Să se calculeze ultima cifră a numărului n^k .
 6. Se citește un număr natural n cu cel puțin 3 cifre. Să se afișeze numărul obținut din n prin eliminarea primei și ultimei cifre.
Ex: pentru $n=24255$ se va afișa 425
 7. Se citesc două numere naturale a și b , fiecare având cifrele distincte. Determinați câte cifre comune au cele 2 numere.
Ex: $a=23416$ și $b=345987$ au 2 cifre comune (3 și 4)
 8. Se citește un număr natural n . Să se elimine din numărul n cifra sau cifrele din mijloc și să se afișeze numărul rezultat.
 9. Se citește un număr natural n . Să se afișeze toate numerele mai mici sau egale cu n cu proprietatea că sunt egale cu suma pătratelor cifrelor lor.
 10. Se citește un număr natural n și o cifră c . Calculați de câte ori apare cifra c în scrierea numerelor naturale de la 1 la n .
 11. Determinați cifra de control a unui număr natural n (suma cifrelor respectivului număr, până când se ajunge la rang de unitate).
Ex: $1824 \rightarrow 1+8+2+4=15 \rightarrow 1+5=6$

Determinarea pătratelor perfecte din intervalul natural $[a,b]$

```
for(n=a;n<=b;n++)
    if(sqrt(n)==(int)sqrt(n))
        cout<<n<<' ';
```

Verificarea primalității numărului n natural

```
if(n<2) prim=0;
else if(n==2) prim=1;
else if(n%2==0) prim=0;
else{ prim=1;
    for(d=3;d*d<=n;d=d+2)
        if(n%d==0) prim=0;
    }
```

```
if(n<2) prim=0;
else { prim=1;
    for(d=2;d<=sqrt(n);d++)
        if(n%d==0) prim=0;
    }
```

1. Să se afișeze toate numerele prime din intervalul natural $[a,b]$.
2. Să se determine cel mai mic număr prim mai mare decât un număr natural n citit de la tastatură.
3. Să se afișeze primele n numere prime, n număr natural citit de la tastatură.
4. Un număr se numește **aproape prim** dacă poate fi scris ca produs de 2 numere prime distincte. Se citesc n numere naturale cu maxim 4 cifre fiecare. Să se determine câte dintre ele sunt aproape prime.
5. Să se afișeze toate numerele naturale cu 4 cifre $(\overline{abcd})$ cu proprietatea că $(\overline{acd})$ este număr prim, iar $(\overline{bad})$ este pătrat perfect.
6. Să se afișeze toate numerele ≤ 1000 cu proprietatea că au exact 3 divizori.
7. Să se afișeze toate numerele prime ≤ 1000 din șirul 1,4,7,10,...
8. Să se afișeze toate modalitățile distincte de descompunere a unui număr par în sumă de 2 numere prime.
Ex: $10=3+7$, $10=5+5$