



Subprogram care verifică dacă un nr n dat este prim

```
int prim(int n)
{if(n==0 || n==1) return 0;
  for(int i=2;i<=sqrt(n);i++)
    if(n%i==0) return 0;
  return 1;
}
```



Subprogram inversul (rasturnatul) numarului dat n

```
int rast(int n)
{
    r=0;
    while(n!=0)
    {
        c=n%10;
        r=r*10+c;
        n=n/10;
    }
    return r;
}
```

**Subprogram schimba cifra c1 cu c2 pentru un nr dat n**

```
int schimba(int &n, int c1, int c2)
{ int r=0;
  while(n!=0)
  { if(n%10==c1)
    r=r*10+c2;
    else r=r*10+n%10;
    n=n/10;
  }
  while(r!=0)
  { n=n*10+r%10;
    r=r/10;
  }
  return n;
}
```



Subprogram cmmdc

cmmdc=cel mai mic divizor comun

Daca **cmmdc(a,b)=1** spunem ca a si b sunt **prime** intre ele.

```
int cmmdc(int x, int y)
{ while(x!=y)
  if(x>y) x=x-y;
  else y=y-x;
  return x;
}
```

Exemplu:

$(8;12)=(8;4)=(4;4)=4$



Subprogram cmmdc

```
int cmmdc (int a, int b)
{
    int i,c;
    for(i=1;i<=a;i++)
        if ((a%i==0)&&(b%i==0)) c=i;
    return c;
}
```

i	c
1	1
2	2
3	2
4	4
5	4
6	4
7	4
8	4



Subprogram cmmdc

Algoritmul lui Euclid constă în împărțiri successive până când se obține restul zero. După prima împărțire, **b** devine deîmpărțit ($a=b$) iar **r** împărțitor ($b=r$) ș.a.m.d. Cmmdc este ultimul rest nenul, adică **b**.

($a > b$)

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

Exemplu: **a=49** **b=12**

a	b	r
49	14	7
14	7	0

**Subprogram cmmmc**

cmmmc=cel mai mic multiplu comun (cel mai mic număr la care se imparte și a și b)

```
int cmmmc (int a, int b)
{
    int k=0;
    do {
        k++;
    } while ((a*k)%b!=0);
    return (a*k);
}
```

Exemplu: **a=4 b=6**

a	b	k	a*k
4	6	1	4
		2	8
		3	12