**Divizibilitate**

1. Numarul / suma divizorilor unui nr nat

div improprii: 1, x

div propria: orice d ai x%d = 0 si d = 1, d=x

Teorema: d este divizor pentru x => x/d divisor lui x

x/2 = >in intervalul [x/2+1, x-1] nu exista niciun divisor

divizorii pot fi doar pana la sqrt(x)

Nr divizorilor

int n, d, nr;

cin>>n;

caz special n = 1;

nr = 2;

for(d = 2; d\*d<n; d++){

if(n%d == 0){

prelucrare d, n/d => nr+=2;

if(d\*d==n)

prelucrare d=> n++;

1. Verificare daca x este nr prim

x = prim ⬄ are exact 2 div => 1 nu e prim

nr = 0;

caz n = 1;

for(d=2; d\*d < n; d++)

if(n%d == 0) nr +=2

if(d\*d == n) nr++

if(nr == 0) n prim

bool ok = true;

if(n<2 || (n%2 == 0 && n != 2)) ok false

for(int d = 3; d\*d <= n && ok; d+=2)

if(n%d == 0) ok = false

if(ok) nr prim

1. Cel mai mare divizor comun

Algoritmul lui Euclid

|  |  |  |
| --- | --- | --- |
| a | b | Rest = a%b |
| 882 | 2100 | 882 |
| 2100 | 882 | 336 |
| 882 | 336 | 210 |
| 336 | 210 | 126 |
| 210 | 126 | 84 |
| 126 | 84 | 42 |
| 84 | 42 | 0 |

r = a%b

while(r != 0){

a = b

b = r

r = a%b

}

cmmdc = b

Obs:

* a si b se modifica => copiii ale val initiale
* caz special a = 0 sau b = 0

a = 0 && b = 0 -> enuntul problemei

a = 0 || b = 0 -> cmmdc a + b

cmmmc = a/cmmdc(a, b)\*b

1. Descompunerea in factori primi

cin >> n;

d = 2;

while(n != 0){

e = 0

while(n%d == 0){

n/=d

e++

}

if(e != 0) prelucrare

d++

}

d = 2;

while(n != 1 && d\*d <= n){

e = 0

while(n%d == 0){

e++, n/=d;

}

if(e != 0) prelucrare

d++

}

if(n!=1)prelucrare n //n ramas este sigur nr prim

Rezultat important

n = f1e1 \* f2e2 \* … \* fkek

nrdivizori = (e1 + 1)\*(e2 +1)\* … \*(ek + 1)