Informatică I2
Laboratorul 13
Algoritmi elementari

Botescu Mihai
mihai.botescu00@e-uvt.ro
Universitatea de Vest din Timișoara
Facultatea de Matematică și Informatică

April 28, 2021

Contents

1 Număr prim

1.1 Noțiuni

Definiție. Un număr natural $p > 1$ este se numește **prim** dacă:

- $p|ab \implies p|a$ sau $p|b$;
- Un număr natural $p > 1$ este se numește indecompozabil (sau ireductibil) dacă: $d|p \implies d = 1$ sau $d = p$

1.2 Observații

- Pentru orice număr natural $p > 1$, p este prim dacă și numai dacă este indecompozabil.
- Cei doi divizori ai unui număr indecompozabil (prim) sunt 1 și însuși numărul.
- Conform definiției, numerele 0 și 1 **nu sunt prime**!
- Un număr natural mai mare decât 1 care nu este prim se numește compus sau decompozabil sau reductibil.

1.3 Verificarea primalității

Pentru a stabili dacă un număr p este prim:

1. Numărăm divizorii săi. Dacă sunt 2 divizori, p este prim.
2. determinăm suma divizorilor. Dacă suma este $p + 1$, numărul este prim.
3. căutăm divizori ai săi diferiți de 1 și de el însuși. Dacă nu găsim, numărul este prim.
4. Cum verificăm algoritmic dacă un număr natural n este prim?
5. presupunem că numărul este prim;
6. verificăm cazurile particulare; dacă n este 0 sau 1, schimbăm presupunerea
7. căutăm un divizor în intervalul $[2, \sqrt{n}]$, parcurgând numerele din interval
8. dacă îl găsim, schimbăm presupunerea

   Observație: Deoarece divizorii unui număr n sunt în pereche, dacă nu găsim divizor în intervalul $[2, \sqrt{n}]$, nu vom găsi nici în intervalul $[\sqrt{n}, n]$.

2 Divizorii unui număr

O problemă frecvent întâlnită este determinarea divizorilor unui număr dat. În practică se pot cere diverse operații cu aceștia: afișarea, însumarea, numărarea, etc.

2.1 Algoritmul naiv

O primă metodă de determinare a divizorilor constă în **a observa că toți divizorii lui n sunt între 1 și n, inclusiv**. Putem **parcurge numerele din acest interval și verifica dacă sunt într-adevăr divizori** ai lui n, caz în care sunt luați în considerare.
O primă soluție este să **observăm că pentru orice** n, de la $\frac{n}{2}$ la n **nu mai sunt divizori**. Putem astfel să înjumătățim interval în care căutăm divizorii. Astfel înjumătățim și timpul de execuție, dar nu este o îmbunătățire suficientă.

2.2 Un algoritm mai eficient

Soluția acceptabilă este să observăm că **divizorii oricărui număr n sunt în pereche**: dacă d este divizor al lui n, atunci și $\frac{n}{d}$ este divizor al lui n. De exemplu, pentru $n = 75$.

- 1 este divizor 75, atunci și $75/1 = 75$ este divizor al lui 75;
- 2 nu este divizor al lui 75
- 3 este divizor 75, atunci și $75/3 = 25$ este divizor al lui 75;
- 4 nu este divizor al lui 75
- 5 este divizor 75, atunci și $75/5 = 15$ este divizor al lui 75;
- 6 nu este divizor al lui 75
- 7 nu este divizor al lui 75
- 8 nu este divizor al lui 75
- 9 nu este divizor al lui 75. Mai mult, $9 \cdot 9 > 75$, alți divizori nu vom mai găsi.

  Divizorii lui 75 sunt: 1 75 3 25 5 15. Constatăm astfel că pentru a determina divizorii lui n este suficient să parcurgem numerele de la 1 la $\sqrt{n}$.

Un caz special îl constituie pătratele perfecte. În cazul lor trebuie evitată analizarea de două ori a lui $\sqrt{n}$, care este divizor al lui n. Pentru 36 avem divizorii:

- 1 în pereche cu 36

- 2 în pereche cu 18
- 3 în pereche cu 12
- 4 în pereche cu 9
- 5 nu este divizor al lui 36
- 6 în pereche cu 6. 6 **trebuie luat o singură dată!**
- $7 \cdot 7 > 36$, **ne oprim!**

2.3 Observații

- Numerele care **nu sunt pătrate perfecte** au **număr par de divizori**.
- Singurele numere cu **număr impar de divizori** sunt **pătratele perfecte**.
- Cel mai mic **divizor propriu** al unui număr natural (diferit de 1 și de numărul însuși) este **număr prim**.

```cpp
#include <iostream>
int main()
{
        int n;
        std::cin >> n;
        for(int d =1 ; d * d <= n ; d ++ )
        if(n % d == 0)
        {
                std::cout << d << " ";
                if(d * d < n) // daca d != sqrt(n)
                std::cout << n / d << " ";
        }
        return 0;
}
```

3 Cel mai mare divizor comun. Cel mai mic multiplu comun.

3.1 Definiții

D1. Fie a și b două numere naturale. Un număr natural d se numește **cel mai mare divizor comun** (pe scurt **cmmdc**) al lui a și b dacă îndeplinește condițiile:

- $d|a$ și $d|b$;
- dacă $c|a$ și $c|b$, atunci $c|d$.
- Cel mai mare divizor comun al numerelor a și b se notează (a, b) sau $gcd(a, b)$ – greatest common divisor.

Dacă (a, b)=1, spunem că **a și b sunt prime între ele** sau **relativ prime**, sau că a **este prim** cu b.

3.2 Proprietăți

Cel mai mare divizor comun al numerelor naturale are proprietățile:

- $((a, b), c) = (a,(b, c))\forall a, b, c \in \mathbb{N}$
- $(a, b) = 1,(a, c) = 1 \implies (a, bc) = 1$
- $a|bc,(a, b) = 1 \implies a|c$
- $a|c, b|c,(a, b) = 1 \implies ab|c$

3.3 Determinarea cmmdc

Cel mai mare divizor comun al două numere naturale n și m poate fi determinat folosind descompunerea în factori primi a celor două numere. Această metodă este mai dificil de implementat. Există o metodă mai simplu de implementat într-un program, numită algoritmul lui Euclid.
Sunt două variante ale algoritmului lui Euclid: cu scăderi și cu împărțiri.

3.4 Algoritmul lui Euclid cu scăderi

Algoritmul lui Euclid cu scăderi se bazează pe ideea că **cele mai mare divizor a două numere divide și diferența acestora**. Algoritmul este:

- Cât timp numerele sunt diferite, se scade numărul mai mic din numărul mai mare.
- Când numerele devin egale, valoare comună este cel mai mare divizor comun al valorilor inițiale.

- Algoritmul nu poate fi aplicat dacă unul dintre numere este 0. De ce?
  Exemplu:

  Fie $n = 32$ și $m = 24$.

  - Numerele nu sunt egale, scădem numărul mai mic din numărul mai mare, $n = n - m = 32 - 24 = 8$.
  - Acum $n = 8$ și $m = 24$.
  - Numerele nu sunt egale, scădem numărul mai mic din numărul mai mare, $m = m - n = 24 - 8 = 16$.
  - Acum $n = 8și m = 16$.
  - Numerele nu sunt egale, scădem numărul mai mic din numărul mai mare, $m = m - n = 16 - 8 = 8$.
  - Acum $n = 8$ și $m = 8$.
  - Numerele sunt egale. Valoarea comună, 8, este cel mai mare divizor comun al valorilor inițiale, 32 și 24.

3.5 Algoritmul lui Euclid cu împărțiri

Algoritmul lui Euclid cu împărțiri se bazează pe ideea că cel mai mare divizor a două numere divide și restul împărțirii acestora, conform teoremei împărțirii cu rest. Algoritmul este:

- Cât timp m != 0:
- Determinăm restul împărțirii lui n la m.
- În continuare n devine m, iar m devine restul calculat.
- Valoarea actuală a lui n este cel mai mare divizor comun a valorilor inițiale.

Exemplu:

- Fie n=32 și m=24.
- m != 0:
- Calculăm r = n % m = 8
- n devine m, iar m devine r.
- Acum n=24 și m=8.
- m != 0:
- Calculăm r = n % m = 0
- n devine m, iar m devine r.
- Acum n=8 și m=0.
- m este 0. Valoarea actuală a lui n = 8 este cel mai mare divizor comun al valorilor inițiale, 32 și 24.

3.6 Cel mai mic multiplu comun

Definiție. Fie a și b două numere naturale. Se numește **cel mai mic multiplu comun** (pe scurt **cmmmc**) al lui a și b cel mai mic număr natural nenul cu proprietatea că se divide atât cu a cât și cu b.
Obs. Cel mai mic multiplu comun al numerelor a și b se notează $[a, b]$ sau $lcm(a, b)$ – least common multiple.

3.7 Determinarea cmmmc

Pentru a determina cel mai mic multiplu comun se pot folosi mai multe metode:

- Determinarea cmmmc folosind cmmdc

  Observație: Produsul a două numere naturale nenule este egal cu produsul dintre cel mai mare divizor comun al lor și cel mai mic multiplu comun al lor.

  $a \cdot b = (a, b) \cdot [a, b]$ $a \cdot b = gcd(a, b) \cdot lcm(a, b)$

- Determinarea cmmmc folosind un algoritm de tip Euclid Fie a și b valorile date. Vom construi valorile m și n, astfel:

```
initial n    a, m    b;
cat timp m != n:
dac  m < n, atunci m cre te cu valoarea lui a: n    n + a
dac  m > n, atunci n cre te cu valoarea lui b: m    m + b
```

  valoarea finală, comună, a lui n și m este cel mai mic multiplu comun pentru a și b

  Observație: Algoritmul poate fi aplicat similar pentru trei sau mai multe numere!

  Aplicații ale cmmmc

  Deja știți că pentru a aduna două (sau mai multe) fracții trebuie să le aducem la același numitor, iar cel mai mic numitor comun a două fracții este egal cu cel mai mic multiplu comun al numitorilor.

  Urmează alte două aplicații mai practice ale **CMMMC**.

  Problema roților dințate

  Se dă un angrenaj format din două roți dințate conectate. Prima are n dinți, a doua are m dinți. Între centrele celor două roți este trasată o linie colorată. Rotile încep să se miște. După câte rotații ale primei roți linia colorată va uni din nou centrele roților?

  Răspunsul se determină folosind cel mai mic multiplu comun: $[n, m]$. Mai precis, prima roată va face $[n, m]n$ rotații iar a doua va face $[n, m]m$ rotații.

Alinierea planetelor

Considerăm trei planete care se rotesc în jurul soarelui. Ele fac rotație completă în a, b, respectiv c ani, numere naturale. Dacă la un moment dat planetele sunt aliniate (între ele și cu soarele), după cât timp vor din nou aliniate?

Răspunsul este $[a, b, c]$ ani. În acest ele vor face $[a, b, c]a$, $[a, b, c]b$, respectiv $[a, b, c]c$ rotații complete în jurul soarelui.