

Ciurului lui Eratostene

Introducere

Ciurul lui Eratostene reprezintă o metodă de determinare a tuturor numerelor prime mai mici sau egale cu o valoare dată, n . În practică, valoarea lui n trebuie să fie una rezonabilă, ea depinzând de restricțiile de timp și memorie aplicate problemei.

Ciurului Eratostene se aplică în felul următor – în continuare lucrăm cu $n=30$:

- se scriu toate numerele naturale de la 1 la $n=30$:
- se taie din lista numărul 1, care nu este prim
- se identifică următorul număr netăiat, 2. Acest este prim, iar din listă se vor tăia toți multipli săi, mai mari decât acesta.
- se identifică următorul număr netăiat, 3. Acest este prim, iar din listă se vor tăia toți multipli săi, mai mari decât acesta. Unii dintre ei au fost deja tăiați, deoarece sunt și multipli ai lui 2.
- se identifică următorul număr netăiat, 5. Acest este prim, iar din listă se vor tăia toți multipli săi, mai mari decât acesta. Unii dintre ei au fost deja tăiați, deoarece sunt și multipli ai lui 2 sau 3.
- se identifică următorul număr netăiat, 7. Acesta este prim, dar totodată $7 \cdot 7 > 30$ căutarea numerelor prime s-a încheiat. Toate numerele care nu au fost tăiate sunt prime.

Algoritm

Pentru identificarea numerelor prime într-un algoritm/program, vom folosi un vector caracteristic, $v[]$, cu următoarea semnificație:

$v[x] = \{0 \text{ dacă } x \text{ prim}, 1 \text{ dacă } x \text{ nu este prim}\}$

Vom porni de la un vector cu toate elementele nule, apoi vom marca pe rând cu 1 toate numerele care sunt multipli ai unor numere mai mici. Un algoritm pseudocod care realizează acest operații este:

```
pentru i ← 0, n execută
    V[i] ← 0
sf_pentru
V[0] ← 1
V[1] ← 1
pentru i ← 2, sqrt(n) execută
    dacă V[i] = 0 atunci
        pentru j ← 2, [n/i] execută
            V[i*j] ← 1
        sf_pentru
    sf_dacă
sf_pentru
```