

Oszthatóság, prímszámok

(feladatsor)

Diszkrét modellek alkalmazásai

1. Írjunk függvényt, amely adott $a, b \in \mathbb{N}^+$ számokra eldönti, hogy a osztója-e b -nek. Oldjuk meg legalább háromféleképpen. A beépített `a.divides(b)` függvényt csak ellenőrzésre használjuk: ellenőrizzük a megoldást minden a és b egész számra 1 és 100 között.
2. Írjunk függvényt, amely visszaadja egy szám pozitív osztóit egy listában. Ne használjuk a beépített `divisors()` tagfüggvényt, csak ellenőrzésre.
3. Ellenőrizzük az 1-től 100-ig lévő számokra, hogy az oszthatóság tranzitív.
4. Írjunk függvényt, amely eldönti, hogy egy adott $n \in \mathbb{N}$ szám prímszám-e. Ne használjuk a beépített prímszámos függvényeket. Ellenőrizzük a megoldást az `is_prime()` beépített függvénnyel 0-tól 1000-ig.
5. Írjuk ki 1-től 100-ig az összes prímszámot – legalább háromféleképpen.
6. Írjuk ki az első 100 prímszámot – legalább háromféleképpen.
7. Hány prímszám van 1 000 000-ig? Legalább háromféleképpen számítsuk ki.
8. Hozzunk létre egy listát, amely az 1000 alatti ikerprímpárokat tartalmazza:
[(3, 5), (5, 7), (11, 13), ...].
9. Implementáljuk Eratoszthenész szitáját: írjunk függvényt, amely visszaadja adott n -ig az összes prímszámot egy listában. (Ne használjuk a beépített prímszámos függvényeket.)
10. Írjunk függvényt, amely kiszámítja $\tau(n)$ -et
a) egyszerű számlálással; b) a prímtényezős felbontás segítségével.
11. Írjunk függvényt, amely kiszámítja $\sigma(n)$ -et
a) egyszerű összeadással; b) a prímtényezős felbontás segítségével.
12. Keressük meg az összes tökéletes számot 10000-ig.