

- Bizonyítsa be, hogy ha f_1, f_2, g_1, g_2 pozitív függvények és $f_1(n) \in O(g_1(n))$ és $f_2(n) \in O(g_2(n))$, akkor
 - $f_1(n) + f_2(n) \in O(\max(g_1(n), g_2(n)))$,
 - $f_1(n)f_2(n) \in O(g_1(n)g_2(n))$.
- Az alábbi függvények közül melyikre igaz, hogy $O(n^2)$?
 $f_1(n) = 11n^2 + 100000$ $f_2(n) = 8n^2 \log_2 n$ $f_3(n) = 1,5n + 3\sqrt{n}$
- Jelölje egy algoritmus maximális lépésszámát az n méretű bemeneteken $L(n)$. Tegyük fel, hogy az algoritmus szerkezetéből következik, hogy $L(n) \leq 2L(\frac{n}{2}) + n$ ha $n \geq 2$ és $L(1) = 1$. (Tegyük fel, hogy $n = 2^k$, azaz n 2-hatvány.)
 Láss be, hogy ekkor $L(n) \in O(n \log n)$. Igaz-e, hogy $L(n) \in O(n^2)$?
- Az alábbi függvényeket rendezze nagyságrend szerint nem csökkenő sorozatba: ha f_i után közvetlenül f_j következik a sorban, akkor $f_i(n) \in O(f_j(n))$ teljesüljön!
 $f_1(n) = 8n^3$ $f_2(n) = 5\sqrt{n} + 1000n$ $f_3(n) = 2^{(\log_2 n)^2}$ $f_4(n) = 1514n^2 \log_2 n$
- Az alábbi pszeudokódban egy $*$ kiírása számít lépésnek. Mutassa meg, hogy a kód lépésszáma $O(n^3)$.

```

for i = 0 to n-1:
    for j = i+1 to n:
        print j darab *
```
- Adjon O becslést a következő függvényekre:
 $(n^2 + 8)(n + 1)$ $(n \log n + n^2)(n^3 + 2)$ $(n! + 2^n)(n^3 + \log(n^2 + 1))$ $(2^n + n^2)(n^3 + 3^n)$
- Jelölje egy algoritmus maximális lépésszámát az n méretű bemeneteken $L(n)$. Adjunk felső becslést az $L(n)$ nagyságrendjére, ha tudjuk, hogy $L(1) = 2$ és $n > 1$ esetén
 - $L(n) = L(n - 1) + 3$
 - $L(n) = L(n - 1) + 5$
 - $L(n) = L(n - 1) + 3n$
 - $L(n) = 2L(n - 1) + 3$
 - $L(n) = L(\lceil n/2 \rceil) + 3$
 - $L(n) = L(\lceil n/2 \rceil) + n^k$
 - $L(n) = 2L(\lceil n/2 \rceil) + 3$
 - $L(n) = 4L(\lceil n/2 \rceil) + 3$
 Az (e)-(h) esetben elegendő 2 hatványra meggondolni.
 Mi változik, ha egyenlőség helyett $\leq$ vagy $\geq$ áll?
- Mely $a, b > 1$ egész számokra teljesülnek az alábbiak?
 $n^a \in O(n^b)$ $2^{an} \in O(2^{bn})$ $\log_a n \in O(\log_b n)$
- Tekintsük az $f_1(n) = 1,5n!$ és $f_2(n) = 200(n - 1)!$ függvényeket. Melyik igaz és melyik nem az alábbiak közül?
 $f_1 \in O(f_2)$ $f_2 \in O(f_1)$ $f_1 \in \Omega(f_2)$ $f_2 \in \Omega(f_1)$ $f_1 \in \Theta(f_2)$ $f_2 \in \Theta(f_1)$
- Az egyszerű algoritmussal, illetve a gyorskereséssel állapítsa meg, hogy az $S = ABBABACABCBAC$ szövegben az $M = ABABC$ minta hányszor fordul elő! Hány összehasonlítást használtak az algoritmusok?
- Az $n > 2$ hosszú csupa 0-ból álló szöveghez adjon meg olyan m hosszú mintát, melyen az egyszerű algoritmus m -től függetlenül $O(n)$ összehasonlítást használ!
- Igazolja, hogy az egyszerű algoritmus várható futási ideje $O(n)$, ha a szöveg és a minta is véletlen 0/1 sorozat (a bitek egymástól függetlenek, mindegyik $\frac{1}{2} - \frac{1}{2}$ valószínűséggel 0 vagy 1).
 Mi a helyzet, ha csak a minta véletlen?