

Příklad 1.

Bydlíme v n -patrovém mrakodrapu a máme k dispozici k vajec. Jen nás nudí byt pořád doma. V zájmu vědy chceme tak zjistit, z kolika pater můžeme vajíčko vyhodit na zem, aniž by se rozbilo.

Všechna vajíčka jsou stejná a pokud přežijí pád z patra p , pak jistě přežijí i pád z nižšího patra. Naopak, pokud se z patra p rozbije, tak ho čeká stejný osud i z vyššího patra.

Jak můžeme zjistit nejvyšší patro takové, že po pádu z něj vajíčko přežije? Uvažme:

- $n = 100$, jedno vejce
- $n = 100$, libovolně mnoho vajec
- $n = 100$, dvě vejce
- dvě vejce pro libovolné n

Definice. Největší společný dělitel dvou přirozených čísel x, y je největší číslo d takové, že x i y jsou dělitelné d .

Příklad 2.

Nalezněte algoritmus, který spočítá největší společný dělitel dvou čísel x, y . Jakou časovou složitost bude mít tento algoritmus vzhledem k hodnotě $\max(x, y)$?

Příklad 3.

Vymyslete algoritmus, kterým pro přirozené n, k spočítáme co nejrychleji n^k .

Příklad 4.

Mějme funkci $f(x, y)$, jejíž kód vypadá následovně:

```
if y == 0:
    return 0
else if even(y):
    return 2*f(x, y/2)
else:
    return 2*f(x, y/2) + x
```

Co funkce $f(x, y)$ dělá? Jakou má časovou složitost?

Příklad 5.

Mějme čísla a, b uložená jako seznamy n cifer v desítkové soustavě (počáteční cifry mohou být nuly). Navrhněte algoritmy, které spočítají

- a jako normální číslo
- $a + b$ znovu jako seznam cifer
- $a < b$, kde odpověď je 0 nebo 1
- $a \cdot b$ znovu jako seznam cifer