

Řešení domácích úkolů - 3. týden

Pythagorejské trojice

Pro dané n vypište počet různých trojic přirozených čísel a, b, c takových, že

- $a, b, c \leq n$
- $a < b$
- a a b jsou nesoudělná
- $a^2 + b^2 = c^2$

Řešení

Analýza

Řešení hrubou silou:

Prohledáme prostor přípustných trojic a, b, c . U těchto řešení můžeme dosáhnout významné úspory, když vhodně zmenšíme dimenzi prohledávaného prostoru, a když pak zvolíme rozumně úzké intervaly přípustných hodnot.

Alternativní řešení:

Celá čísla m, n , $m > n$, definují unikátní Pythagorejskou trojici takto:

$$a = m^2 - n^2, \quad b = 2mn, \quad c = m^2 + n^2$$

Namísto prostoru a, b, c tedy můžeme prohledávat prostor m, n .

Vzorové řešení

V tomto řešení:

- i neboli a je nejmenší člen trojice a probíhá nezávisle od 1 do n . Přesnější by bylo zvolit interval od 2 do $0,71n$ - žádná Pythagorejská trojice neobsahuje číslo 1 a pokud má být i nejmenším členem trojice, musí být menší než $n/\sqrt{2}$.
- j neboli b musí být větší než i .
- k neboli c musí být úplným čtvercem a splňovat s i, j Pythagorovu větu. Pokud jsou i, j nesoudělná, je i, j, k Pythagorejská trojice.

Je rozumné uvažovat o pořadí testů, abychom vyloučili co nejvíc kandidátů co nejlevněji.

```
import math

n = int(input())
pocet = 0

for a in range(2, int(0.71*n)):
    for b in range(a+1, n):
        if math.gcd(a, b) != 1:
            continue
        c_2 = a**2 + b**2
```

```
c = math.isqrt(c_2)
if c <= n and c**2 == c_2:
    pocet += 1

print(pocet)
```

Alternativní řešení

Protože m, n generují unikátní trojici, v případě nalezení trojice s $a > b$ prostě a, b prohodíme, tedy trojici započteme.

```
max_n = int(input())

pocet = 0

for m in range(2, max_n):
    for n in range(1, m):
        a = m * m - n * n
        b = 2 * m * n
        c = m * m + n * n
        if a > 0 and c <= max_n and gcd(a, b) == 1:
            pocet += 1

print(pocet)
```

Obvyklé problémy v řešeních

Testy

Pro tuto úlohu má ReCodEx 3 testy s hodnotami $n = 10, 100, 300$. Příslušné počty Pythagorejských trojic jsou 1, 16 a 47.

Časté chyby:

- Vylučování řešení s $a > b$ u alternativního řešení.
- Přílišné "osekání" prohledávané oblasti. Je lepší si raději nechat bezpečnostní "okraj".