

Mám následující kód v Lean4.

Můžeš mi ho prosím doplnit o komentáře a vysvětlit, jak celý důkaz funguje?



Tady je váš Lean 4 kód s komentáři a vysvětlením, jak důkaz funguje:

Lean



```
import Mathlib

-- Důkaz tvrzení, že n-té Fibonacciho číslo je menší než 2 na n-tou.
-- Tvrzení: pro každé přirozené číslo n platí fib n < 2^n
theorem fib_lt_two_pow (n : ℕ) : Nat.fib n < 2 ^ n := by
  -- Důkaz se provádí pomocí silné indukce (strong_induction_on) nad n.
  -- Silná indukce nám umožňuje předpokládat, že tvrzení platí pro všech
  na k < n.
  induction n using Nat.strong_induction_on with
  | h n ih => -- 'h' je název pro indukční krok, 'ih' je indukční hypotéza
    -- Rozebereme případy pro n: 0, 1 a n+2.
    match n with
    | 0 => -- Základní případ: n = 0
      -- Chceme dokázat: fib 0 < 2^0
      -- Víme, že fib 0 = 0 a 2^0 = 1. Tedy 0 < 1, což platí.
      simp [Nat.fib, pow_zero]
    | 1 => -- Základní případ: n = 1
      -- Chceme dokázat: fib 1 < 2^1
      -- Víme, že fib 1 = 1 a 2^1 = 2. Tedy 1 < 2, což platí.
      simp [Nat.fib, pow_one]
    | n + 2 => -- Indukční krok pro n >= 2 (tedy n + 2, kde n je libovolné přirozené číslo)
      -- Použijeme rekurentní definici Fibonacciho posloupnosti: fib (n+2) = fib (n+1) + fib n
      rw [Nat.fib_add_two]
      -- Nyní chceme dokázat: fib (n+1) + fib n < 2^(n+2)

      -- Indukční hypotéza (ih) nám říká, že pro každé k < n+2 platí fib k < 2^k.
      -- Aplikujeme indukční hypotézu pro n (víme, že n < n+2 pro n >= 0)
```

```

    have h_n_simple : Nat.fib n < 2 ^ n := ih n (by simp) -- 'by simp'
    potvrzuje, že  $n < n+2$ 

    -- Aplikujeme indukční hypotézu pro  $n+1$  (víme, že  $n+1 < n+2$  pro  $n \geq 0$ )
    have h_n_plus_1_simple : Nat.fib (n + 1) < 2 * 2 ^ n := by
      -- Důkaz této nerovnosti krok za krokem:
      calc
        -- Z indukční hypotézy pro  $(n+1)$  víme:  $\text{fib } (n+1) < 2^{(n+1)}$ 
        Nat.fib (n + 1) < 2 ^ (n + 1) := ih (n + 1) (by simp)
        -- A víme, že  $2^{(n+1)} = 2 * 2^n$  (použitím pravidla  $a^{(x+y)} = a^x * a^y$ )
        _ = 2 * 2 ^ n := by rw [Nat.pow_succ']

    -- Pomocná rovnost pro pravou stranu dokazované nerovnosti:  $2^{(n+2)} = 4 * 2^n$ 
    have two_pow_n_plus_2_simple : 2 ^ (n + 2) = 4 * 2 ^ n := by
      -- Důkaz této rovnosti krok za krokem:
      calc
        --  $2^{(n+2)} = 2^n * 2^2$  (použitím pravidla  $a^{(x+y)} = a^x * a^y$ )
        2 ^ (n + 2) = 2 ^ n * 2 ^ 2 := by rw [Nat.pow_add]
        --  $2^2$  je 4
        _ = 2 ^ n * 4 := by norm_num
        -- Zaměníme pořadí (komutativita násobení)
        _ = 4 * 2 ^ n := by rw [Nat.mul_comm]

    -- Zjednodušení výrazů v cílové nerovnosti.
    -- 'simp only' je použit pro selektivní zjednodušení, aby nedošlo
    k přílišné automatické redukci.
    simp only [Nat.fib_add_two, Nat.pow_succ', Nat.pow_add, Nat.pow_two]

    -- Nyní máme dokázat:  $(\text{Nat.fib } (n + 1) + \text{Nat.fib } n) < 4 * 2^n$ 
    -- Použijeme linarith, což je automatický řešitel pro lineární nerovnosti.
    -- Linarith použije nerovnosti h_n_simple a h_n_plus_1_simple k prokázání cíle.
    -- Máme:
    -- 1.  $\text{fib } n < 2^n$ 
    -- 2.  $\text{fib } (n+1) < 2 * 2^n$ 
    -- Sečtením levých stran:  $\text{fib } n + \text{fib } (n+1)$ 
    -- Sečtením pravých stran:  $2^n + 2 * 2^n = 3 * 2^n$ 
    -- Tedy:  $\text{fib } n + \text{fib } (n+1) < 3 * 2^n$ 
    -- Vzhledem k tomu, že  $3 * 2^n < 4 * 2^n$  (protože  $3 < 4$ ),
    -- linarith snadno odvodí požadované  $\text{fib } (n+1) + \text{fib } n < 4 * 2^n$ 
    linarith [h_n_simple, h_n_plus_1_simple]

    -- Komentovaná definice Fibonacciho posloupnosti (pouze pro referenci, n

```

```
eni součástí důkazu)
-- def fib : ℕ → ℕ
-- | 0 => 0      -- fib 0 je 0
-- | 1 => 1      -- fib 1 je 1
-- | n + 2 => fib (n + 1) + fib n -- fib (n+2) je součet předchozích dv
ou Fibonacciho čísel
```

Jak celý důkaz funguje?

Tento důkaz tvrdí, že **každé Fibonacciho číslo $fib(n)$ je menší než 2^n** . Používá se pro to **silná indukce** (strong induction).

Co je silná indukce?

Na rozdíl od standardní indukce, kde pro důkaz tvrzení pro n předpokládáme platnost tvrzení pouze pro $n - 1$, silná indukce nám umožňuje předpokládat, že tvrzení platí pro všechna přirozená čísla $k < n$. To je obzvláště užitečné u rekurentních definic, jako je Fibonacciho posloupnost, kde hodnota $fib(n + 2)$ závisí na $fib(n + 1)$ i $fib(n)$.

Struktura důkazu:

1. Základní případy (Base Cases):

- $n = 0$: Musíme ukázat, že $fib(0) < 2^0$.
 - $fib(0) = 0$ a $2^0 = 1$.
 - $0 < 1$, což platí.
- $n = 1$: Musíme ukázat, že $fib(1) < 2^1$.
 - $fib(1) = 1$ a $2^1 = 2$.
 - $1 < 2$, což platí.

Leanův příkaz `simp` automaticky ověří tyto jednoduché aritmetické pravdy pomocí definic `Nat.fib`, `pow_zero` a `pow_one`.

2. Indukční krok (Inductive Step):

- Chceme dokázat, že tvrzení platí pro libovolné $n + 2$ (kde $n \geq 0$, takže $n + 2$ pokrývá všechna $n \geq 2$), za předpokladu, že platí pro všechna $k < n + 2$.
- **Indukční hypotéza (IH):** Předpokládáme, že pro všechna $k < n + 2$ platí $fib(k) < 2^k$.
- **Cíl:** Dokázat, že $fib(n + 2) < 2^{n + 2}$.
- **Použití definice Fibonacciho čísel:**
Nejprve použijeme definici Fibonacciho čísel: $fib(n + 2) = fib(n + 1) + fib(n)$.

Tuto změnu provede `rw [Nat.fib_add_two]`.

Nyní je cílem dokázat $\text{fib}(n+1) + \text{fib}(n) < 2^{n+2}$.

- **Aplikace indukční hypotézy:**

Protože $n < n+2$ a $n+1 < n+2$ (pro n

≥ 0), můžeme aplikovat indukční hypotézu na $\text{fib}(n)$ a $\text{fib}(n+1)$:

- `h_n_simple` : $\text{fib}(n) < 2^n$ (protože $n < n+2$)
- `h_n_plus_1_simple` : $\text{fib}(n+1) < 2^{n+1}$ (protože $n+1 < n+2$)

- **Přetvoření nerovnosti pro $\text{fib}(n+1)$:**

Dále si uvědomíme, že 2^{n+1} lze přepsat jako $2 \cdot 2^n$. To nám dává:

$\text{fib}(n+1) < 2 \cdot 2^n$.

- **Přetvoření pravé strany cíle:**

Podobně si upravíme pravou stranu cílové nerovnosti 2^{n+2} . Víme, že $2^{n+2} = 2^n \cdot 2^2 = 2^n$

$\cdot 4 = 4$

$\cdot 2^n$.

- **Kombinace nerovností pomocí `linarith`:**

Nyní máme tyto fakta:

1. $\text{fib}(n) < 2^n$
2. $\text{fib}(n+1) < 2 \cdot 2^n$
3. Cíl je ukázat: $\text{fib}(n+1) + \text{fib}(n) < 4 \cdot 2^n$

Když sečteme levé strany nerovností (1) a (2), dostaneme:

$\text{fib}(n) + \text{fib}(n+1) < 2^n + 2 \cdot 2^n$

$\text{fib}(n) + \text{fib}(n+1) < 3 \cdot 2^n$

Protože víme, že $3 \cdot 2^n < 4 \cdot 2^n$ (jelikož $3 < 4$ a 2^n je kladné), můžeme uzavřít, že:

$\text{fib}(n) + \text{fib}(n+1) < 4 \cdot 2^n$.

Toto je přesně to, co jsme chtěli dokázat. `linarith` je taktika v Leanu, která automaticky řeší lineární nerovnosti, a je schopna odvodit tuto závěrečnou nerovnost z poskytnutých dílčích nerovností.

Celý důkaz je tedy systematickým rozбором případů a použitím silné indukce k postupnému dokázání, že Fibonacciho čísla rostou pomaleji než mocniny dvojky.



Nechte Gemini napsat dokument nebo kód, který budete moct upravit

