

Uvod u interaktivno dokazivanje teorema

Vežbe 07

Zadatak 1 *Alternirajuća suma neparnih prirodnih brojeva*

Pokazati da važi:

$$-1 + 3 - 5 + \dots + (-1)^n(2n - 1) = (-1)^n n.$$

Primitivnom rekurzijom definisati funkciju *alternirajuca-suma* :: *nat* ⇒ *int* koja računa alternirajuću sumu neparnih brojeva od 1 do $2n - 1$, tj. definisati funkciju koja računa levu stranu jednakosti.

primrec *alternirajuca-suma* :: *nat* ⇒ *int* **where**

alternirajuca-suma 0 = 0

| *alternirajuca-suma* (Suc n) = *alternirajuca-suma* n + (-1)[^](Suc n) * (2 * int (Suc n) - 1)

Proveriti vrednost funkcije *alternirajuca-suma* za proizvoljan prirodni broj.

value *alternirajuca-suma* 6

Dokazati sledeću lemu indukcijom koristeći metode za automatsko dokazivanje.

lemma *alternirajuca-suma* n = (-1)[^]n * int n

by (*induction* n) (*auto simp add: algebra-simps*)

Dokazati sledeću lemu indukcijom raspisivanjem detaljnog Isar dokaza.

lemma *alternirajuca-suma* n = (-1)[^]n * int n

proof (*induction* n)

case 0

then show ?case **by** *simp*

next

case (Suc n)

have *alternirajuca-suma* (Suc n) = *alternirajuca-suma* n + (-1)[^](Suc n) * (2 * int (Suc n) - 1)

by (*rule alternirajuca-suma.simps(2)*)

also have ... = (-1)[^]n * int n + (-1)[^](Suc n) * (2 * int (Suc n) - 1)

using *Suc* **by** *simp*

also have ... = 2 * (-1)[^](Suc n) * int (Suc n) - (-1)[^](Suc n) - (-1)[^](Suc n) * int n

by (*simp add: algebra-simps*)

also have ... = (-1)[^](Suc n) * int n + (-1)[^](Suc n)

by (*simp add: algebra-simps*)

also have ... = (-1)[^](Suc n) * int (Suc n)

by (*simp add: algebra-simps*)

finally show ?case .

qed

Zadatak 2 Množenje matrica

Pokazati da važi sledeća jednakost:

$$\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}^n = \begin{pmatrix} 1 & n \\ 0 & 1 \end{pmatrix}, n \in \mathbb{N}.$$

Definisati tip *mat2* koji predstavlja jednu 2×2 matricu prirodnih brojeva. Tip *mat2* definisati kao skraćenicu uređene četvorkе prirodnih brojeva. Uređena četvorkа odgovara 2×2 matrici kao

$$(a, b, c, d) \equiv \begin{pmatrix} a & b \\ c & d \end{pmatrix}.$$

type-synonym *mat2* = *nat* × *nat* × *nat* × *nat*

term (1, 1, 0, 1)::*mat2*

Definisati konstantu *eye* :: *mat2*, koja predstavlja jediničnu matricu.

definition *eye* :: *mat2* **where**

$$eye \equiv (1, 0, 0, 1)$$

Definisati funkciju *mat-mul* :: *mat2* ⇒ *mat2* ⇒ *mat2*, koja množi dve matrice.

fun *mat-mul* :: *mat2* ⇒ *mat2* ⇒ *mat2* **where**

$$\begin{aligned} mat-mul (a1, b1, c1, d1) (a2, b2, c2, d2) = \\ (a1*a2 + b1*c2, a1*b2 + b1*d2, \\ c1*a2 + d1*c2, c1*b2 + d1*d2) \end{aligned}$$

Definisati funkciju *mat-pow* :: *mat2* ⇒ *nat* ⇒ *mat2*, koja stepenuje matricu.

fun *mat-pow* :: *mat2* ⇒ *nat* ⇒ *mat2* **where**

$$mat-pow M 0 = eye$$

| *mat-pow* M (Suc n) = *mat-mul* M (*mat-pow* M n)

Dokazati sledeću lemu koristeći metode za automatsko dokazivanje.

lemma *mat-pow* (1, 1, 0, 1) n = (1, n, 0, 1)

by (*induction* n) (*auto simp add: eye-def*)

Dokazati sledeću lemu indukcijom raspisivanjem detaljnog Isar dokaza.

lemma *mat-pow* (1, 1, 0, 1) n = (1, n, 0, 1)

proof (*induction* n)

case 0

then show ?*case*

by (*simp add: eye-def*)

next

case (Suc n)

then show ?*case*

proof –

$$\begin{aligned} \text{have } mat-pow (1, 1, 0, 1) (Suc n) = mat-mul (1, 1, 0, 1) \\ (mat-pow (1, 1, 0, 1) n) \end{aligned}$$

by (*simp only: mat-pow.simps(2)*)

$$\text{also have } \dots = mat-mul (1, 1, 0, 1) (1, n, 0, 1)$$

by (*simp only: Suc*)

$$\text{also have } \dots = (1, n + 1, 0, 1) \text{ by } simp$$

$$\text{also have } \dots = (1, Suc n, 0, 1) \text{ by } simp$$

```

    finally show ?thesis .
  qed
qed

```

Zadatak 3 Deljivost

Pokazati sledeću lemu.

Savet: Obrisati *One-nat-def* i *algebra-simps* iz *simp*-a u finalnom koraku dokaza.

lemma

```

  fixes n::nat
  shows (6::nat) dvd n * (n + 1) * (2 * n + 1)
proof (induction n)
  case 0
  then show ?case
    by simp
next
  case (Suc n)
  then show ?case
  proof -
    note [simp] = algebra-simps
    have Suc n * (Suc n + 1) * (2 * Suc n + 1) = (n + 1) * (n + 2) * (2 * (n + 1) + 1)
  by simp
  also have ... = (n + 1) * (n + 2) * (2 * n + 3) by simp
  also have ... = n * (n + 1) * (2 * n + 3) + 2 * (n + 1) * (2 * n + 3) by simp
  also have ... = n * (n + 1) * (2 * n + 1) + 2 * n * (n + 1) + 2 * (n + 1) * (2 * n +
3) by simp
  also have ... = n * (n + 1) * (2 * n + 1) + 2 * (n + 1) * (3 * n + 3) by simp
  also have ... = n * (n + 1) * (2 * n + 1) + 6 * (n + 1) * (n + 1) by simp
  finally show ?thesis
    using Suc
    by (simp del: algebra-simps One-nat-def)
  qed
qed

```

Zadatak 4 Nejednakost

Pokazati da za svaki prirodan broj $n > 2$ važi $n^2 > 2 * n + 1$.

Savet: Iskoristiti *nat-induct-at-least* kao pravilo indukcije i lemu *power2-eq-square*.

thm *nat-induct-at-least*

thm *power2-eq-square*

lemma *n2-2n:*

```

  fixes n::nat
  assumes n ≥ 3
  shows n2 > 2 * n + 1
  using assms
proof (induction n rule: nat-induct-at-least)
  case base
  then show ?case by simp
next
  case (Suc n)

```

```
have  $2 * \text{Suc } n + 1 < 2 * (\text{Suc } n) + 2 * n$ 
  using  $\langle n \geq 3 \rangle$  by simp
also have  $\dots = 2 * n + 1 + 2 * n + 1$ 
  by simp
also have  $\dots < n^2 + 2 * n + 1$ 
  using Suc by simp
also have  $\dots = (\text{Suc } n)^2$ 
  by (simp add: power2-eq-square)
finally show ?case .
qed
```