Induction and Recursion

Induction

Induction

Everybody knows that to prove a sequence of statements

$$S_0, S_1, S_2, \ldots$$

Induction

Everybody knows that to prove a sequence of statements

$$S_0, S_1, S_2, \ldots$$

it suffices to prove only that

Induction

Everybody knows that to prove a sequence of statements

$$S_0, S_1, S_2, \ldots$$

it suffices to prove only that

(1) S_0 is true, and

Induction

Everybody knows that to prove a sequence of statements

$$S_0, S_1, S_2, \ldots$$

it suffices to prove only that

(1) S_0 is true, and

Induction

Everybody knows that to prove a sequence of statements

$$S_0, S_1, S_2, \ldots$$

it suffices to prove only that

(1) S_0 is true, and

(2) $(\forall n)(S_n \to S_{n+1})$ is true.

Induction

Everybody knows that to prove a sequence of statements

$$S_0, S_1, S_2, \ldots$$

it suffices to prove only that

(1) S_0 is true, and

(2) $(\forall n)(S_n \to S_{n+1})$ is true.

Induction

Everybody knows that to prove a sequence of statements

$$S_0, S_1, S_2, \ldots$$

it suffices to prove only that

(1) S_0 is true, and

(2) $(\forall n)(S_n \to S_{n+1})$ is true.

The validity of this method relies on the fact that $\mathbb{N}$ is the intersection of all inductive sets.

Induction

Everybody knows that to prove a sequence of statements

$$S_0, S_1, S_2, \ldots$$

it suffices to prove only that

(1) S_0 is true, and

(2) $(\forall n)(S_n \to S_{n+1})$ is true.

The validity of this method relies on the fact that $\mathbb{N}$ is the intersection of all inductive sets.

Justification for the method of proof by induction:

Induction

Everybody knows that to prove a sequence of statements

$$S_0, S_1, S_2, \ldots$$

it suffices to prove only that

(1) S_0 is true, and

(2) $(\forall n)(S_n \to S_{n+1})$ is true.

The validity of this method relies on the fact that $\mathbb{N}$ is the intersection of all inductive sets.

Justification for the method of proof by induction:

Find a formula $\varphi(x)$ so that $(\forall n)(\varphi(n) \leftrightarrow S_n)$.

Induction

Everybody knows that to prove a sequence of statements

$$S_0, S_1, S_2, \ldots$$

it suffices to prove only that

(1) S_0 is true, and

(2) $(\forall n)(S_n \to S_{n+1})$ is true.

The validity of this method relies on the fact that $\mathbb{N}$ is the intersection of all inductive sets.
Justification for the method of proof by induction:
Find a formula $\varphi(x)$ so that $(\forall n)(\varphi(n) \leftrightarrow S_n)$. Then observe that, if Items (1) and (2) from above are true, then

Induction

Everybody knows that to prove a sequence of statements

$$S_0, S_1, S_2, \dots$$

it suffices to prove only that

(1) S_0 is true, and

(2) $(\forall n)(S_n \to S_{n+1})$ is true.

The validity of this method relies on the fact that $\mathbb{N}$ is the intersection of all inductive sets.

Justification for the method of proof by induction:

Find a formula $\varphi(x)$ so that $(\forall n)(\varphi(n) \leftrightarrow S_n)$. Then observe that, if Items (1) and (2) from above are true, then

$$\{x \in \mathbb{N} \mid \varphi(x)\}$$

Induction

Everybody knows that to prove a sequence of statements

$$S_0, S_1, S_2, \ldots$$

it suffices to prove only that

(1) S_0 is true, and

(2) $(\forall n)(S_n \to S_{n+1})$ is true.

The validity of this method relies on the fact that $\mathbb{N}$ is the intersection of all inductive sets.

Justification for the method of proof by induction:

Find a formula $\varphi(x)$ so that $(\forall n)(\varphi(n) \leftrightarrow S_n)$. Then observe that, if Items (1) and (2) from above are true, then

$$\{x \in \mathbb{N} \mid \varphi(x)\}$$

is an inductive subset of $\mathbb{N}$.

Induction

Everybody knows that to prove a sequence of statements

$$S_0, S_1, S_2, \ldots$$

it suffices to prove only that

(1) S_0 is true, and

(2) $(\forall n)(S_n \to S_{n+1})$ is true.

The validity of this method relies on the fact that $\mathbb{N}$ is the intersection of all inductive sets.

Justification for the method of proof by induction:

Find a formula $\varphi(x)$ so that $(\forall n)(\varphi(n) \leftrightarrow S_n)$. Then observe that, if Items (1) and (2) from above are true, then

$$\{x \in \mathbb{N} \mid \varphi(x)\}$$

is an inductive subset of $\mathbb{N}$. Hence the displayed set is $\mathbb{N}$ itself.

Recursion Theorem

Recursion Theorem

The function $F(n) = n!$ is easy to define 'recusively'.

Recursion Theorem

The function $F(n) = n!$ is easy to define ‘recusively’. The function $F\colon \mathbb{N} \to \mathbb{N}\colon n \mapsto n!$ is defined by:

Recursion Theorem

The function $F(n) = n!$ is easy to define 'recusively'. The function $F\colon \mathbb{N} \to \mathbb{N}\colon n \mapsto n!$ is defined by:

1. $F(0) = 1$

Recursion Theorem

The function $F(n) = n!$ is easy to define 'recusively'. The function $F\colon \mathbb{N} \to \mathbb{N}\colon n \mapsto n!$ is defined by:

1. $F(0) = 1$

Recursion Theorem

The function $F(n) = n!$ is easy to define 'recusively'. The function $F\colon \mathbb{N} \to \mathbb{N}\colon n \mapsto n!$ is defined by:

1. $F(0) = 1$
2. $F(n+1) = (n+1) \cdot F(n)$.

Recursion Theorem

The function $F(n) = n!$ is easy to define 'recusively'. The function $F\colon \mathbb{N} \to \mathbb{N}\colon n \mapsto n!$ is defined by:

1. $F(0) = 1$
2. $F(n+1) = (n+1) \cdot F(n)$.

Recursion Theorem

The function $F(n) = n!$ is easy to define 'recusively'. The function $F\colon \mathbb{N} \to \mathbb{N}\colon n \mapsto n!$ is defined by:

1. $F(0) = 1$
2. $F(n+1) = (n+1) \cdot F(n)$.

This function is not easy to define any other way.

Recursion Theorem

The function $F(n) = n!$ is easy to define 'recusively'. The function $F\colon \mathbb{N} \to \mathbb{N}\colon n \mapsto n!$ is defined by:

1. $F(0) = 1$
2. $F(n+1) = (n+1) \cdot F(n)$.

This function is not easy to define any other way.

Recursion Theorem.

Recursion Theorem

The function $F(n) = n!$ is easy to define 'recusively'. The function $F \colon \mathbb{N} \to \mathbb{N} \colon n \mapsto n!$ is defined by:

1. $F(0) = 1$
2. $F(n+1) = (n+1) \cdot F(n)$.

This function is not easy to define any other way.

Recursion Theorem. For any set A, any $a_0 \in A$, and any function $G \colon A \times \mathbb{N} \to A$,

Recursion Theorem

The function $F(n) = n!$ is easy to define 'recusively'. The function $F\colon \mathbb{N} \to \mathbb{N}\colon n \mapsto n!$ is defined by:

1. $F(0) = 1$
2. $F(n+1) = (n+1) \cdot F(n)$.

This function is not easy to define any other way.

Recursion Theorem. For any set A, any $a_0 \in A$, and any function $G\colon A \times \mathbb{N} \to A$, there exists a unique function $F\colon \mathbb{N} \to A$ satisfying

1. $F(0) = a_0$

Recursion Theorem

The function $F(n) = n!$ is easy to define 'recusively'. The function $F\colon \mathbb{N} \to \mathbb{N}\colon n \mapsto n!$ is defined by:

1. $F(0) = 1$
2. $F(n+1) = (n+1) \cdot F(n)$.

This function is not easy to define any other way.

Recursion Theorem. For any set A, any $a_0 \in A$, and any function $G\colon A \times \mathbb{N} \to A$, there exists a unique function $F\colon \mathbb{N} \to A$ satisfying

1. $F(0) = a_0$
2. $F(S(n)) = G(F(n), n)$ for all $n \in \mathbb{N}$.

Idea of proof

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST.

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

- **(Stage 1.)** Define the set P of all 'partial computations'. A partial computation is a function $t_m\colon m \to A$ that satisfies the recursion on its domain. This stage relies on the Axiom of Comprehension.

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

- **(Stage 1.)** Define the set P of all 'partial computations'. A partial computation is a function $t_m \colon m \to A$ that satisfies the recursion on its domain. This stage relies on the Axiom of Comprehension.
- **(Stage 2.)** Form the union $F = \bigcup P$ of the set constructed in the previous step. This stage relies on the Axiom of Union.

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

- **(Stage 1.)** Define the set P of all 'partial computations'. A partial computation is a function $t_m \colon m \to A$ that satisfies the recursion on its domain. This stage relies on the Axiom of Comprehension.
- **(Stage 2.)** Form the union $F = \bigcup P$ of the set constructed in the previous step. This stage relies on the Axiom of Union.
- **(Stage 3.)** Verify the details:

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

- **(Stage 1.)** Define the set P of all 'partial computations'. A partial computation is a function $t_m \colon m \to A$ that satisfies the recursion on its domain. This stage relies on the Axiom of Comprehension.
- **(Stage 2.)** Form the union $F = \bigcup P$ of the set constructed in the previous step. This stage relies on the Axiom of Union.
- **(Stage 3.)** Verify the details:

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

- **(Stage 1.)** Define the set P of all 'partial computations'. A partial computation is a function $t_m : m \to A$ that satisfies the recursion on its domain. This stage relies on the Axiom of Comprehension.
- **(Stage 2.)** Form the union $F = \bigcup P$ of the set constructed in the previous step. This stage relies on the Axiom of Union.
- **(Stage 3.)** Verify the details:
    1. F satisfies the function rule.

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

- **(Stage 1.)** Define the set P of all 'partial computations'. A partial computation is a function $t_m : m \to A$ that satisfies the recursion on its domain. This stage relies on the Axiom of Comprehension.
- **(Stage 2.)** Form the union $F = \bigcup P$ of the set constructed in the previous step. This stage relies on the Axiom of Union.
- **(Stage 3.)** Verify the details:
  1. F satisfies the function rule.

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

- **(Stage 1.)** Define the set P of all 'partial computations'. A partial computation is a function $t_m\colon m \to A$ that satisfies the recursion on its domain. This stage relies on the Axiom of Comprehension.
- **(Stage 2.)** Form the union $F = \bigcup P$ of the set constructed in the previous step. This stage relies on the Axiom of Union.
- **(Stage 3.)** Verify the details:
    1. F satisfies the function rule.
    2. $\mathrm{dom}(F) = \mathbb{N}, \mathrm{im}(F) \subseteq A$.

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

- **(Stage 1.)** Define the set P of all 'partial computations'. A partial computation is a function $t_m : m \to A$ that satisfies the recursion on its domain. This stage relies on the Axiom of Comprehension.
- **(Stage 2.)** Form the union $F = \bigcup P$ of the set constructed in the previous step. This stage relies on the Axiom of Union.
- **(Stage 3.)** Verify the details:
  1. F satisfies the function rule.
  2. $\mathrm{dom}(F) = \mathbb{N}, \mathrm{im}(F) \subseteq A$.

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

- **(Stage 1.)** Define the set P of all 'partial computations'. A partial computation is a function $t_m : m \to A$ that satisfies the recursion on its domain. This stage relies on the Axiom of Comprehension.
- **(Stage 2.)** Form the union $F = \bigcup P$ of the set constructed in the previous step. This stage relies on the Axiom of Union.
- **(Stage 3.)** Verify the details:
  1. F satisfies the function rule.
  2. $\mathrm{dom}(F) = \mathbb{N}, \mathrm{im}(F) \subseteq A$.
  3. F satisfies the recursion.

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

- **(Stage 1.)** Define the set P of all 'partial computations'. A partial computation is a function $t_m\colon m \to A$ that satisfies the recursion on its domain. This stage relies on the Axiom of Comprehension.
- **(Stage 2.)** Form the union $F = \bigcup P$ of the set constructed in the previous step. This stage relies on the Axiom of Union.
- **(Stage 3.)** Verify the details:
  1. F satisfies the function rule.
  2. $\mathrm{dom}(F) = \mathbb{N}, \mathrm{im}(F) \subseteq A$.
  3. F satisfies the recursion.

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

- **(Stage 1.)** Define the set P of all 'partial computations'. A partial computation is a function $t_m : m \to A$ that satisfies the recursion on its domain. This stage relies on the Axiom of Comprehension.
- **(Stage 2.)** Form the union $F = \bigcup P$ of the set constructed in the previous step. This stage relies on the Axiom of Union.
- **(Stage 3.)** Verify the details:
    1. F satisfies the function rule.
    2. $\mathrm{dom}(F) = \mathbb{N}, \mathrm{im}(F) \subseteq A$.
    3. F satisfies the recursion.
    4. Any function $F' : \mathbb{N} \to A$ that satisfies the recursion must equal F.

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

- **(Stage 1.)** Define the set P of all 'partial computations'. A partial computation is a function $t_m \colon m \to A$ that satisfies the recursion on its domain. This stage relies on the Axiom of Comprehension.
- **(Stage 2.)** Form the union $F = \bigcup P$ of the set constructed in the previous step. This stage relies on the Axiom of Union.
- **(Stage 3.)** Verify the details:
  1. F satisfies the function rule.
  2. $\mathrm{dom}(F) = \mathbb{N}, \mathrm{im}(F) \subseteq A$.
  3. F satisfies the recursion.
  4. Any function $F' \colon \mathbb{N} \to A$ that satisfies the recursion must equal F.

Idea of proof

The proof of the Recursion Theorem appears on page 48 of Hrbacek and Jech and in a more general form on page 98 of Monk's NST. Here we sketch the idea:

- **(Stage 1.)** Define the set P of all 'partial computations'. A partial computation is a function $t_m\colon m \to A$ that satisfies the recursion on its domain. This stage relies on the Axiom of Comprehension.
- **(Stage 2.)** Form the union $F = \bigcup P$ of the set constructed in the previous step. This stage relies on the Axiom of Union.
- **(Stage 3.)** Verify the details:
  1. F satisfies the function rule.
  2. $\mathrm{dom}(F) = \mathbb{N}, \mathrm{im}(F) \subseteq A$.
  3. F satisfies the recursion.
  4. Any function $F'\colon \mathbb{N} \to A$ that satisfies the recursion must equal F.

All parts of Stage 3 can be proved by induction.