

# Computer Algebra Proofs for Combinatorial Inequalities and Identities

Manuel Kauers

RISC-Linz, Austria

# Typical Combinatorial Identities

# Typical Combinatorial Identities

- ▶ Identities involving sums and products

$$\sum_{k=1}^n \sum_{i=1}^k \frac{1}{i} = -n + (n+1) \sum_{k=1}^n \frac{1}{k} \quad (n \geq 1)$$

# Typical Combinatorial Identities

- ▶ Identities involving sums and products

$$\sum_{k=1}^n \sum_{i=1}^k \frac{1}{i} = -n + (n+1) \sum_{k=1}^n \frac{1}{k} \quad (n \geq 1)$$

- ▶ Identities about the Fibonacci numbers

$$\sum_{k=1}^n \frac{(-1)^k}{F_k F_{k+1}} = -\frac{F_n}{F_{n+1}} \quad (n \geq 1)$$

# Typical Combinatorial Identities

- ▶ Identities involving sums and products

$$\sum_{k=1}^n \sum_{i=1}^k \frac{1}{i} = -n + (n+1) \sum_{k=1}^n \frac{1}{k} \quad (n \geq 1)$$

- ▶ Identities about the Fibonacci numbers

$$\sum_{k=1}^n \frac{(-1)^k}{F_k F_{k+1}} = -\frac{F_n}{F_{n+1}} \quad (n \geq 1)$$

- ▶ Identities with orthogonal polynomials, double exponential sequences, ...

# Typical Combinatorial Identities

- ▶ Identities involving sums and products

$$\sum_{k=1}^n \sum_{i=1}^k \frac{1}{i} = -n + (n+1) \sum_{k=1}^n \frac{1}{k} \quad (n \geq 1)$$

- ▶ Identities about the Fibonacci numbers

$$\sum_{k=1}^n \frac{(-1)^k}{F_k F_{k+1}} = -\frac{F_n}{F_{n+1}} \quad (n \geq 1)$$

- ▶ Identities with orthogonal polynomials, double exponential sequences, ...
- ▶ Routines are desired which not only *prove* but also *find* such identities.

# Typical Combinatorial Inequalities

# Typical Combinatorial Inequalities

- ▶ Inequalities involving sums and products

$$\prod_{k=0}^n \frac{3k+4}{3k+2} > 1 + \frac{2}{3} \sum_{k=1}^{n+1} \frac{1}{k} \quad (n \geq 1)$$



## Typical Combinatorial Inequalities

- ▶ Inequalities involving sums and products

$$\prod_{k=0}^n \frac{3k+4}{3k+2} > 1 + \frac{2}{3} \sum_{k=1}^{n+1} \frac{1}{k} \quad (n \geq 1)$$

- ▶ Inequalities about the Fibonacci numbers

$$\sum_{k=1}^n \frac{(2F_{k+1} - F_k)^2}{F_k} \geq \frac{(3F_{n+1} + F_n - 3)^2}{F_{n+2} - 1} \quad (n \geq 2)$$

## Typical Combinatorial Inequalities

- ▶ Inequalities involving sums and products

$$\prod_{k=0}^n \frac{3k+4}{3k+2} > 1 + \frac{2}{3} \sum_{k=1}^{n+1} \frac{1}{k} \quad (n \geq 1)$$

- ▶ Inequalities about the Fibonacci numbers

$$\sum_{k=1}^n \frac{(2F_{k+1} - F_k)^2}{F_k} \geq \frac{(3F_{n+1} + F_n - 3)^2}{F_{n+2} - 1} \quad (n \geq 2)$$

- ▶ Inequalities like

$$\left( \sum_{k=1}^n \sqrt{k} \right)^2 \stackrel{?}{\geq} \left( \sum_{k=1}^n \sqrt[3]{k} \right)^3 \quad (n \geq 1)$$

## Typical Combinatorial Inequalities

- ▶ Inequalities involving sums and products

$$\prod_{k=0}^n \frac{3k+4}{3k+2} > 1 + \frac{2}{3} \sum_{k=1}^{n+1} \frac{1}{k} \quad (n \geq 1)$$

- ▶ Inequalities about the Fibonacci numbers

$$\sum_{k=1}^n \frac{(2F_{k+1} - F_k)^2}{F_k} \geq \frac{(3F_{n+1} + F_n - 3)^2}{F_{n+2} - 1} \quad (n \geq 2)$$

- ▶ Inequalities like

$$\left( \sum_{k=1}^n \sqrt{k} \right)^2 \stackrel{?}{\geq} \left( \sum_{k=1}^n \sqrt[3]{k} \right)^3 \quad (n \geq 1)$$

- ▶ “Combinatorial” here just means that the inequality depends on a discrete parameter  $n$ . Inequalities like  $\sin x < x$  ( $x \geq 0$ ) are out of scope.

# *Proving Combinatorial Identities*

# Known Algorithms for Proving Identities

# Known Algorithms for Proving Identities

- ▶ Summation Algorithms

# Known Algorithms for Proving Identities

- ▶ Summation Algorithms
  - ▶ Gosper's algorithm

# Known Algorithms for Proving Identities

- ▶ Summation Algorithms
  - ▶ Gosper's algorithm
  - ▶ Zeilberger's algorithm



# Known Algorithms for Proving Identities

- ▶ Summation Algorithms
  - ▶ Gosper's algorithm
  - ▶ Zeilberger's algorithm
  - ▶ Sister Celine's algorithm

# Known Algorithms for Proving Identities

- ▶ Summation Algorithms
  - ▶ Gosper's algorithm
  - ▶ Zeilberger's algorithm
  - ▶ Sister Celine's algorithm
  - ▶ Karr's algorithm

# Known Algorithms for Proving Identities

- ▶ Summation Algorithms
  - ▶ Gosper's algorithm
  - ▶ Zeilberger's algorithm
  - ▶ Sister Celine's algorithm
  - ▶ Karr's algorithm
  - ▶ ...variations and generalizations of those ...

# Known Algorithms for Proving Identities

- ▶ Summation Algorithms
  - ▶ Gosper's algorithm
  - ▶ Zeilberger's algorithm
  - ▶ Sister Celine's algorithm
  - ▶ Karr's algorithm
  - ▶ ... variations and generalizations of those ...
- ▶ Generating Function Algorithms (remember Paule's talk)

# Known Algorithms for Proving Identities

- ▶ Summation Algorithms
  - ▶ Gosper's algorithm
  - ▶ Zeilberger's algorithm
  - ▶ Sister Celine's algorithm
  - ▶ Karr's algorithm
  - ▶ ... variations and generalizations of those ...
- ▶ Generating Function Algorithms (remember Paule's talk)
- ▶ *Today*: An algorithm for proving identities, which is applicable to a much larger input class.

## Proof by Induction: Outline

- ▶ Note: Proving an identity  $A = B$  amounts to testing zero equivalence of  $A - B$ .

## Proof by Induction: Outline

- ▶ Note: Proving an identity  $A = B$  amounts to testing zero equivalence of  $A - B$ .
- ▶ Task: Given a sequence  $(f_n)$ , prove that

$$\forall n \geq 0 : f_n = 0.$$

## Proof by Induction: Outline

- ▶ Note: Proving an identity  $A = B$  amounts to testing zero equivalence of  $A - B$ .
- ▶ Task: Given a sequence  $(f_n)$ , prove that

$$\forall n \geq 0 : f_n = 0.$$

- ▶ Idea: Find an  $N \geq 0$  such that

$$(\forall n \geq 0 : f_n = 0) \iff (f_0 = f_1 = \dots = f_{N-1} = 0).$$

Then zero equivalence of  $(f_n)$  can be decided by just evaluating the sequence at the first  $N$  points.



## Proof by Induction: Outline

- ▶ Note: Proving an identity  $A = B$  amounts to testing zero equivalence of  $A - B$ .
- ▶ Task: Given a sequence  $(f_n)$ , prove that

$$\forall n \geq 0 : f_n = 0.$$

- ▶ Idea: Find an  $N \geq 0$  such that

$$(\forall n \geq 0 : f_n = 0) \iff (f_0 = f_1 = \dots = f_{N-1} = 0).$$

Then zero equivalence of  $(f_n)$  can be decided by just evaluating the sequence at the first  $N$  points.

- ▶ Clearly: Every  $N \geq 0$  with

$$\forall n \geq 0 : (f_n = f_{n+1} = \dots = f_{n+N-1} = 0 \implies f_{n+N} = 0)$$

does the job.

## Proof by Induction: Outline

- ▶ Note: Proving an identity  $A = B$  amounts to testing zero equivalence of  $A - B$ .

- ▶ Task: Given a sequence  $(f_n)$ , prove that

$$\forall n \geq 0 : f_n = 0.$$

- ▶ Idea: Find an  $N \geq 0$  such that

$$(\forall n \geq 0 : f_n = 0) \iff (f_0 = f_1 = \dots = f_{N-1} = 0).$$

Then zero equivalence of  $(f_n)$  can be decided by just evaluating the sequence at the first  $N$  points.

- ▶ Clearly: Every  $N \geq 0$  with

$$\forall n \geq 0 : (f_n = f_{n+1} = \dots = f_{n+N-1} = 0 \implies f_{n+N} = 0)$$

does the job.

- ▶ Proof: If  $N$  has this property and  $f_0 = \dots = f_{N-1} = 0$  then  $f \equiv 0$  by induction. If not  $f_0 = \dots = f_{N-1} = 0$ , then  $f \not\equiv 0$  anyway.

# Proof by Induction: Use Knowledge

## Proof by Induction: Use Knowledge

- ▶ Method: try for  $N = 1, 2, 3, \dots$  whether

$$\forall n \geq 0 : (f_n = f_{n+1} = \dots = f_{n+N} = 0 \implies f_{n+N+1} = 0)$$

Stop as soon as this is the case and output the corresponding  $N$ .

## Proof by Induction: Use Knowledge

- ▶ Method: try for  $N = 1, 2, 3, \dots$  whether

$$\forall n \geq 0 : (f_n = f_{n+1} = \dots = f_{n+N} = 0 \implies f_{n+N+1} = 0)$$

Stop as soon as this is the case and output the corresponding  $N$ .

- ▶ Sufficient: if

$$\forall x_0, \dots, x_N : x_0 = x_1 = \dots = x_{N-1} = 0 \implies x_N = 0$$

then  $N$  certainly qualifies.

## Proof by Induction: Use Knowledge

- ▶ Method: try for  $N = 1, 2, 3, \dots$  whether

$$\forall n \geq 0 : (f_n = f_{n+1} = \dots = f_{n+N} = 0 \implies f_{n+N+1} = 0)$$

Stop as soon as this is the case and output the corresponding  $N$ .

- ▶ Sufficient: if

$$\forall x_0, \dots, x_N : x_0 = x_1 = \dots = x_{N-1} = 0 \implies x_N = 0$$

then  $N$  certainly qualifies.

- ▶ But: This can hardly be true for any  $N$ , if  $x_0, \dots, x_N$  are independent.

## Proof by Induction: Use Knowledge

- ▶ Method: try for  $N = 1, 2, 3, \dots$  whether

$$\forall n \geq 0 : (f_n = f_{n+1} = \dots = f_{n+N} = 0 \implies f_{n+N+1} = 0)$$

Stop as soon as this is the case and output the corresponding  $N$ .

- ▶ Sufficient: if

$$\forall x_0, \dots, x_N : x_0 = x_1 = \dots = x_{N-1} = 0 \implies x_N = 0$$

then  $N$  certainly qualifies.

- ▶ But: This can hardly be true for any  $N$ , if  $x_0, \dots, x_N$  are independent.
- ▶ Here, we need not assume that  $x_0, \dots, x_N$  be independent! If  $(f_n)$  is defined via recurrence equations, then these equations give rise to known polynomial relations

$$p_1(x_0, \dots, x_N) = \dots = p_m(x_0, \dots, x_N) = 0$$

# Proof by Induction: Use Knowledge and Computer Algebra



# Proof by Induction: Use Knowledge and Computer Algebra

- ▶ Thus we may deliver an  $N$  with

$$\begin{aligned} \forall x_0, \dots, x_{N+1} : (p_1 = \dots = p_m = x_0 = x_1 = \dots = x_N = 0) \\ \implies x_{N+1} = 0 \end{aligned}$$

# Proof by Induction: Use Knowledge and Computer Algebra

- ▶ Thus we may deliver an  $N$  with

$$\begin{aligned}\forall x_0, \dots, x_{N+1} : (p_1 = \dots = p_m = x_0 = x_1 = \dots = x_N = 0) \\ \implies x_{N+1} = 0\end{aligned}$$

- ▶ In other words, an  $N$  with

$$x_{N+1} \in \text{Rad}\langle p_1, \dots, p_m, x_0, \dots, x_N \rangle.$$

# Proof by Induction: Use Knowledge and Computer Algebra

- ▶ Thus we may deliver an  $N$  with

$$\begin{aligned}\forall x_0, \dots, x_{N+1} : (p_1 = \dots = p_m = x_0 = x_1 = \dots = x_N = 0) \\ \implies x_{N+1} = 0\end{aligned}$$

- ▶ In other words, an  $N$  with

$$x_{N+1} \in \text{Rad}\langle p_1, \dots, p_m, x_0, \dots, x_N \rangle.$$

- ▶ This can be decided using *Gröbner Bases*.  $\square$

## Example: Cassini's Identity

- ▶ Let's prove  $F_{n+1}^2 - F_n F_{n+2} = (-1)^n$ .

## Example: Cassini's Identity

- ▶ Let's prove  $F_{n+1}^2 - F_n F_{n+2} = (-1)^n$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (F_{n+i}) \quad y_i \sim ((-1)^{n+i}) \quad (i = 0, 1, 2, 3, \dots)$$

## Example: Cassini's Identity

- ▶ Let's prove  $F_{n+1}^2 - F_n F_{n+2} = (-1)^n$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (F_{n+i}) \quad y_i \sim ((-1)^{n+i}) \quad (i = 0, 1, 2, 3, \dots)$$

- ▶ Then we know

$$x_2 = x_1 + x_0, \quad x_3 = x_2 + x_1, \quad x_4 = x_3 + x_2, \quad \dots$$

$$y_1 = -y_0, \quad y_2 = -y_1, \quad y_3 = -y_2, \quad \dots$$

## Example: Cassini's Identity

- ▶ Let's prove  $F_{n+1}^2 - F_n F_{n+2} = (-1)^n$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (F_{n+i}) \quad y_i \sim ((-1)^{n+i}) \quad (i = 0, 1, 2, 3, \dots)$$

- ▶ Then we know

$$x_2 = x_1 + x_0, \quad x_3 = x_2 + x_1, \quad x_4 = x_3 + x_2, \quad \dots$$

$$y_1 = -y_0, \quad y_2 = -y_1, \quad y_3 = -y_2, \quad \dots$$

- ▶ First iteration ( $N = 0$ ):

$$x_1^2 - x_0 x_2 - y_1 \stackrel{?}{\in} \text{Rad}\langle x_2 - x_1 - x_0, y_1 + y_0, y_2 + y_1 \rangle$$

## Example: Cassini's Identity

- ▶ Let's prove  $F_{n+1}^2 - F_n F_{n+2} = (-1)^n$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (F_{n+i}) \quad y_i \sim ((-1)^{n+i}) \quad (i = 0, 1, 2, 3, \dots)$$

- ▶ Then we know

$$x_2 = x_1 + x_0, \quad x_3 = x_2 + x_1, \quad x_4 = x_3 + x_2, \quad \dots$$

$$y_1 = -y_0, \quad y_2 = -y_1, \quad y_3 = -y_2, \quad \dots$$

- ▶ First iteration ( $N = 0$ ):

$$x_1^2 - x_0 x_2 - y_1 \stackrel{?}{\in} \text{Rad}\langle x_2 - x_1 - x_0, y_1 + y_0, y_2 + y_1 \rangle \quad \text{false.}$$



## Example: Cassini's Identity

- ▶ Let's prove  $F_{n+1}^2 - F_n F_{n+2} = (-1)^n$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (F_{n+i}) \quad y_i \sim ((-1)^{n+i}) \quad (i = 0, 1, 2, 3, \dots)$$

- ▶ Then we know

$$x_2 = x_1 + x_0, \quad x_3 = x_2 + x_1, \quad x_4 = x_3 + x_2, \quad \dots$$

$$y_1 = -y_0, \quad y_2 = -y_1, \quad y_3 = -y_2, \quad \dots$$

- ▶ First iteration ( $N = 0$ ):

$$x_1^2 - x_0 x_2 - y_1 \stackrel{?}{\in} \text{Rad}\langle x_2 - x_1 - x_0, y_1 + y_0, y_2 + y_1 \rangle \quad \text{false.}$$

- ▶ Second iteration ( $N = 1$ ):

$$x_2^2 - x_1 x_3 - y_2 \stackrel{?}{\in} \text{Rad}\langle x_1^2 - x_0 x_2 - y_1, x_2 - x_1 - x_0, \\ x_3 - x_2 - x_1, y_1 + y_0, y_2 + y_1 \rangle$$

## Example: Cassini's Identity

- ▶ Let's prove  $F_{n+1}^2 - F_n F_{n+2} = (-1)^n$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (F_{n+i}) \quad y_i \sim ((-1)^{n+i}) \quad (i = 0, 1, 2, 3, \dots)$$

- ▶ Then we know

$$x_2 = x_1 + x_0, \quad x_3 = x_2 + x_1, \quad x_4 = x_3 + x_2, \quad \dots$$

$$y_1 = -y_0, \quad y_2 = -y_1, \quad y_3 = -y_2, \quad \dots$$

- ▶ First iteration ( $N = 0$ ):

$$x_1^2 - x_0 x_2 - y_1 \stackrel{?}{\in} \text{Rad}\langle x_2 - x_1 - x_0, y_1 + y_0, y_2 + y_1 \rangle \quad \text{false.}$$

- ▶ Second iteration ( $N = 1$ ):

$$x_2^2 - x_1 x_3 - y_2 \stackrel{?}{\in} \text{Rad}\langle x_1^2 - x_0 x_2 - y_1, x_2 - x_1 - x_0, \\ x_3 - x_2 - x_1, y_1 + y_0, y_2 + y_1 \rangle \quad \text{true.}$$

## Example: Cassini's Identity

- ▶ Let's prove  $F_{n+1}^2 - F_n F_{n+2} = (-1)^n$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (F_{n+i}) \quad y_i \sim ((-1)^{n+i}) \quad (i = 0, 1, 2, 3, \dots)$$

- ▶ Then we know

$$x_2 = x_1 + x_0, \quad x_3 = x_2 + x_1, \quad x_4 = x_3 + x_2, \quad \dots$$

$$y_1 = -y_0, \quad y_2 = -y_1, \quad y_3 = -y_2, \quad \dots$$

- ▶ First iteration ( $N = 0$ ):

$$x_1^2 - x_0 x_2 - y_1 \stackrel{?}{\in} \text{Rad}\langle x_2 - x_1 - x_0, y_1 + y_0, y_2 + y_1 \rangle \quad \text{false.}$$

- ▶ Second iteration ( $N = 1$ ):

$$x_2^2 - x_1 x_3 - y_2 \stackrel{?}{\in} \text{Rad}\langle x_1^2 - x_0 x_2 - y_1, x_2 - x_1 - x_0, \\ x_3 - x_2 - x_1, y_1 + y_0, y_2 + y_1 \rangle \quad \text{true.}$$

- ▶ The proof is completed by checking the claim for  $n = 0$ .

# When is the Method Applicable?

## When is the Method Applicable?

- ▶ The sequence  $(f_n)$  should be defined by a system of recurrences.

## When is the Method Applicable?

- ▶ The sequence  $(f_n)$  should be defined by a system of recurrences.
- ▶ The value  $f_n$  may depend rationally upon a history  $f_{n-1}, f_{n-2}, \dots, f_{n-r}$  of fixed length, and possibly on values of other sequences  $g_n, h_n, \dots$  which are themselves defined in the same way.

## When is the Method Applicable?

- ▶ The sequence  $(f_n)$  should be defined by a system of recurrences.
- ▶ The value  $f_n$  may depend rationally upon a history  $f_{n-1}, f_{n-2}, \dots, f_{n-r}$  of fixed length, and possibly on values of other sequences  $g_n, h_n, \dots$  which are themselves defined in the same way.
- ▶ Recurrence scheme:

$$g_{n+2} = \text{rat}_g(g_n, g_{n+1})$$

---

$g_n$	$g_{n+1}$	$g_{n+2}$	$g_{n+3}$	$g_{n+4}$	$\dots$
$h_n$	$h_{n+1}$	$h_{n+2}$	$h_{n+3}$	$h_{n+4}$	$\dots$
$f_n$	$f_{n+1}$	$f_{n+2}$	$f_{n+3}$	$f_{n+4}$	$\dots$

---

## When is the Method Applicable?

- ▶ The sequence  $(f_n)$  should be defined by a system of recurrences.
- ▶ The value  $f_n$  may depend rationally upon a history  $f_{n-1}, f_{n-2}, \dots, f_{n-r}$  of fixed length, and possibly on values of other sequences  $g_n, h_n, \dots$  which are themselves defined in the same way.
- ▶ Recurrence scheme:

$$g_{n+2} = \text{rat}_g(g_n, g_{n+1})$$

---

$g_n$	$g_{n+1}$	$g_{n+2}$	$g_{n+3}$	$g_{n+4}$	$\dots$
$h_n$	$h_{n+1}$	$h_{n+2}$	$h_{n+3}$	$h_{n+4}$	$\dots$
$f_n$	$f_{n+1}$	$f_{n+2}$	$f_{n+3}$	$f_{n+4}$	$\dots$

---



## When is the Method Applicable?

- ▶ The sequence  $(f_n)$  should be defined by a system of recurrences.
- ▶ The value  $f_n$  may depend rationally upon a history  $f_{n-1}, f_{n-2}, \dots, f_{n-r}$  of fixed length, and possibly on values of other sequences  $g_n, h_n, \dots$  which are themselves defined in the same way.
- ▶ Recurrence scheme:

$$g_{n+2} = \text{rat}_g(g_n, g_{n+1})$$

---

$g_n$	$g_{n+1}$	$g_{n+2}$	$g_{n+3}$	$g_{n+4}$	$\dots$
$h_n$	$h_{n+1}$	$h_{n+2}$	$h_{n+3}$	$h_{n+4}$	$\dots$
$f_n$	$f_{n+1}$	$f_{n+2}$	$f_{n+3}$	$f_{n+4}$	$\dots$

---

## When is the Method Applicable?

- ▶ The sequence  $(f_n)$  should be defined by a system of recurrences.
- ▶ The value  $f_n$  may depend rationally upon a history  $f_{n-1}, f_{n-2}, \dots, f_{n-r}$  of fixed length, and possibly on values of other sequences  $g_n, h_n, \dots$  which are themselves defined in the same way.
- ▶ Recurrence scheme:

$$h_{n+2} = \text{rat}_h(g_n, g_{n+1}, g_{n+2}, h_n, h_{n+1})$$

---

$g_n$	$g_{n+1}$	$g_{n+2}$	$g_{n+3}$	$g_{n+4}$	$\dots$
$h_n$	$h_{n+1}$	$h_{n+2}$	$h_{n+3}$	$h_{n+4}$	$\dots$
$f_n$	$f_{n+1}$	$f_{n+2}$	$f_{n+3}$	$f_{n+4}$	$\dots$

---

## When is the Method Applicable?

- ▶ The sequence  $(f_n)$  should be defined by a system of recurrences.
- ▶ The value  $f_n$  may depend rationally upon a history  $f_{n-1}, f_{n-2}, \dots, f_{n-r}$  of fixed length, and possibly on values of other sequences  $g_n, h_n, \dots$  which are themselves defined in the same way.
- ▶ Recurrence scheme:

$$h_{n+2} = \text{rat}_h(g_n, g_{n+1}, g_{n+2}, h_n, h_{n+1})$$

---

$g_n$	$g_{n+1}$	$g_{n+2}$	$g_{n+3}$	$g_{n+4}$	$\dots$
$h_n$	$h_{n+1}$	$h_{n+2}$	$h_{n+3}$	$h_{n+4}$	$\dots$
$f_n$	$f_{n+1}$	$f_{n+2}$	$f_{n+3}$	$f_{n+4}$	$\dots$

---

## When is the Method Applicable?

- ▶ The sequence  $(f_n)$  should be defined by a system of recurrences.
- ▶ The value  $f_n$  may depend rationally upon a history  $f_{n-1}, f_{n-2}, \dots, f_{n-r}$  of fixed length, and possibly on values of other sequences  $g_n, h_n, \dots$  which are themselves defined in the same way.
- ▶ Recurrence scheme:

$$h_{n+2} = \text{rat}_h(g_n, g_{n+1}, g_{n+2}, h_n, h_{n+1})$$

---

$g_n$	$g_{n+1}$	$g_{n+2}$	$g_{n+3}$	$g_{n+4}$	$\dots$
$h_n$	$h_{n+1}$	$h_{n+2}$	$h_{n+3}$	$h_{n+4}$	$\dots$
$f_n$	$f_{n+1}$	$f_{n+2}$	$f_{n+3}$	$f_{n+4}$	$\dots$

---

## When is the Method Applicable?

- ▶ The sequence  $(f_n)$  should be defined by a system of recurrences.
- ▶ The value  $f_n$  may depend rationally upon a history  $f_{n-1}, f_{n-2}, \dots, f_{n-r}$  of fixed length, and possibly on values of other sequences  $g_n, h_n, \dots$  which are themselves defined in the same way.
- ▶ Recurrence scheme:

$$f_{n+1} = \text{rat}_f(g_n, g_{n+1}, h_n, h_{n+1}, f_n)$$

---

$g_n$	$g_{n+1}$	$g_{n+2}$	$g_{n+3}$	$g_{n+4}$	$\dots$
$h_n$	$h_{n+1}$	$h_{n+2}$	$h_{n+3}$	$h_{n+4}$	$\dots$
$f_n$	$f_{n+1}$	$f_{n+2}$	$f_{n+3}$	$f_{n+4}$	$\dots$

---

## When is the Method Applicable?

- ▶ The sequence  $(f_n)$  should be defined by a system of recurrences.
- ▶ The value  $f_n$  may depend rationally upon a history  $f_{n-1}, f_{n-2}, \dots, f_{n-r}$  of fixed length, and possibly on values of other sequences  $g_n, h_n, \dots$  which are themselves defined in the same way.
- ▶ Recurrence scheme:

$$f_{n+1} = \text{rat}_f(g_n, g_{n+1}, h_n, h_{n+1}, f_n)$$

---

$g_n$	$g_{n+1}$	$g_{n+2}$	$g_{n+3}$	$g_{n+4}$	$\dots$
$h_n$	$h_{n+1}$	$h_{n+2}$	$h_{n+3}$	$h_{n+4}$	$\dots$
$f_n$	$f_{n+1}$	$f_{n+2}$	$f_{n+3}$	$f_{n+4}$	$\dots$

---

## When is the Method Applicable?

- ▶ The sequence  $(f_n)$  should be defined by a system of recurrences.
- ▶ The value  $f_n$  may depend rationally upon a history  $f_{n-1}, f_{n-2}, \dots, f_{n-r}$  of fixed length, and possibly on values of other sequences  $g_n, h_n, \dots$  which are themselves defined in the same way.
- ▶ Recurrence scheme:

$$f_{n+1} = \text{rat}_f(g_n, g_{n+1}, h_n, h_{n+1}, f_n)$$

---

$g_n$	$g_{n+1}$	$g_{n+2}$	$g_{n+3}$	$g_{n+4}$	$\dots$
$h_n$	$h_{n+1}$	$h_{n+2}$	$h_{n+3}$	$h_{n+4}$	$\dots$
$f_n$	$f_{n+1}$	$f_{n+2}$	$f_{n+3}$	$f_{n+4}$	$\dots$

---

## When is the Method Applicable?

- ▶ The sequence  $(f_n)$  should be defined by a system of recurrences.
- ▶ The value  $f_n$  may depend rationally upon a history  $f_{n-1}, f_{n-2}, \dots, f_{n-r}$  of fixed length, and possibly on values of other sequences  $g_n, h_n, \dots$  which are themselves defined in the same way.
- ▶ Recurrence scheme:

$$f_{n+1} = \text{rat}_f(g_n, g_{n+1}, h_n, h_{n+1}, f_n)$$

---

$g_n$	$g_{n+1}$	$g_{n+2}$	$g_{n+3}$	$g_{n+4}$	$\dots$
$h_n$	$h_{n+1}$	$h_{n+2}$	$h_{n+3}$	$h_{n+4}$	$\dots$
$f_n$	$f_{n+1}$	$f_{n+2}$	$f_{n+3}$	$f_{n+4}$	$\dots$

---



# When is the Method Applicable?

## When is the Method Applicable?

- ▶ This class of sequences is really big!

## When is the Method Applicable?

- ▶ This class of sequences is really big!
- ▶ It contains many special sequences such as

$$n, \quad F_n, \quad F_{2^n}, \quad F_{F_n}, \quad P_n(x), \quad L_n^\alpha(x), \quad C_n^m(x), \quad \dots$$

## When is the Method Applicable?

- ▶ This class of sequences is really big!
- ▶ It contains many special sequences such as

$$n, \quad F_n, \quad F_{2^n}, \quad F_{F_n}, \quad P_n(x), \quad L_n^\alpha(x), \quad C_n^m(x), \quad \dots$$

- ▶ It satisfies important closure properties such as

$$+, \cdot, -, /, \quad \Sigma, \quad \Pi, \quad K, \quad \text{affine transforms}$$

## When is the Method Applicable?

- ▶ This class of sequences is really big!
- ▶ It contains many special sequences such as

$$n, \quad F_n, \quad F_{2^n}, \quad F_{F_n}, \quad P_n(x), \quad L_n^\alpha(x), \quad C_n^m(x), \quad \dots$$

- ▶ It satisfies important closure properties such as

$$+, \cdot, -, /, \quad \Sigma, \quad \Pi, \quad K, \quad \text{affine transforms}$$

- ▶ *Theorem.* For all sequences from this class, the algorithm described before terminates (i.e., a value  $N$  is always found).

## When is the Method Applicable?

- ▶ This class of sequences is really big!
- ▶ It contains many special sequences such as

$$n, F_n, F_{2^n}, F_{F_n}, P_n(x), L_n^\alpha(x), C_n^m(x), \dots$$

- ▶ It satisfies important closure properties such as

$$+, \cdot, -, /, \Sigma, \Pi, K, \text{ affine transforms}$$

- ▶ *Theorem.* For all sequences from this class, the algorithm described before terminates (i.e., a value  $N$  is always found).
- ▶ In particular: *Zero equivalence is decidable for this class.*

# Example Gallery

## Example Gallery

$$\blacktriangleright \sum_{k=1}^n \sum_{i=1}^k \frac{1}{i} = -n + (n+1) \sum_{k=1}^n \frac{1}{k}$$



## Example Gallery

$$\begin{aligned} \blacktriangleright \sum_{k=1}^n \sum_{i=1}^k \frac{1}{i} &= -n + (n+1) \sum_{k=1}^n \frac{1}{k} \\ \blacktriangleright \sum_{k=1}^n \frac{(-1)^k}{F_k F_{k+1}} &= -\frac{F_n}{F_{n+1}} \end{aligned}$$

## Example Gallery

- ▶ 
$$\sum_{k=1}^n \sum_{i=1}^k \frac{1}{i} = -n + (n+1) \sum_{k=1}^n \frac{1}{k}$$
- ▶ 
$$\sum_{k=1}^n \frac{(-1)^k}{F_k F_{k+1}} = -\frac{F_n}{F_{n+1}}$$
- ▶ 
$$\sum_{k=0}^n \frac{1}{U_{2^k-1}(x)} = \frac{2xU_{2^n}(x) - U_{2^n-1}(x)}{U_{2^n-1}(x)}$$

## Example Gallery

- ▶ 
$$\sum_{k=1}^n \sum_{i=1}^k \frac{1}{i} = -n + (n+1) \sum_{k=1}^n \frac{1}{k}$$
- ▶ 
$$\sum_{k=1}^n \frac{(-1)^k}{F_k F_{k+1}} = -\frac{F_n}{F_{n+1}}$$
- ▶ 
$$\sum_{k=0}^n \frac{1}{U_{2^k-1}(x)} = \frac{2xU_{2^n}(x) - U_{2^n-1}(x)}{U_{2^n-1}(x)}$$
- ▶ 
$$\frac{2}{2 + \frac{3}{3 + \frac{4}{\dots + \frac{n}{n}}}}$$

$$= \sum_{k=0}^n \frac{(-1)^k}{k!}$$

## Example Gallery

$$\blacktriangleright \sum_{k=1}^n \sum_{i=1}^k \frac{1}{i} = -n + (n+1) \sum_{k=1}^n \frac{1}{k}$$

$$\blacktriangleright \sum_{k=1}^n \frac{(-1)^k}{F_k F_{k+1}} = -\frac{F_n}{F_{n+1}}$$

$$\blacktriangleright \sum_{k=0}^n \frac{1}{U_{2^k-1}(x)} = \frac{2xU_{2^n}(x) - U_{2^n-1}(x)}{U_{2^n-1}(x)}$$

$$\blacktriangleright \frac{2}{2 + \frac{3}{3 + \frac{4}{\dots + \frac{n}{n}}}}$$

▶ ...

# *Proving Combinatorial Inequalities*

# Known Algorithms for Proving Inequalities

# Known Algorithms for Proving Inequalities

- ▶ None.

# Known Algorithms for Proving Inequalities

- ▶ None.
  
- ▶ *Today*: A method for proving inequalities, which succeeds for a great many instances.



## Proof by Induction: Outline

- ▶ Note: Proving an inequality  $A > B$  amounts to testing positivity of  $A - B$ .

## Proof by Induction: Outline

- ▶ Note: Proving an inequality  $A > B$  amounts to testing positivity of  $A - B$ .
- ▶ Task: Given a sequence  $(f_n)$ , prove that

$$\forall n \geq 0 : f_n > 0.$$

## Proof by Induction: Outline

- ▶ Note: Proving an inequality  $A > B$  amounts to testing positivity of  $A - B$ .
- ▶ Task: Given a sequence  $(f_n)$ , prove that

$$\forall n \geq 0 : f_n > 0.$$

- ▶ Idea: Find an  $N \geq 0$  such that

$$(\forall n \geq 0 : f_n > 0) \iff (f_0 > 0 \wedge f_1 > 0 \wedge \cdots \wedge f_{N-1} > 0).$$

## Proof by Induction: Outline

- ▶ Note: Proving an inequality  $A > B$  amounts to testing positivity of  $A - B$ .
- ▶ Task: Given a sequence  $(f_n)$ , prove that

$$\forall n \geq 0 : f_n > 0.$$

- ▶ Idea: Find an  $N \geq 0$  such that

$$(\forall n \geq 0 : f_n > 0) \iff (f_0 > 0 \wedge f_1 > 0 \wedge \cdots \wedge f_{N-1} > 0).$$

- ▶ Clearly: Every  $N \geq 0$  with

$$\forall n \geq 0 : (f_n > 0 \wedge \cdots \wedge f_{n+N-1} > 0 \implies f_{n+N} > 0)$$

does the job.

## Proof by Induction: Outline

- ▶ Note: Proving an inequality  $A > B$  amounts to testing positivity of  $A - B$ .
- ▶ Task: Given a sequence  $(f_n)$ , prove that

$$\forall n \geq 0 : f_n > 0.$$

- ▶ Idea: Find an  $N \geq 0$  such that

$$(\forall n \geq 0 : f_n > 0) \iff (f_0 > 0 \wedge f_1 > 0 \wedge \dots \wedge f_{N-1} > 0).$$

- ▶ Clearly: Every  $N \geq 0$  with

$$\forall n \geq 0 : (f_n > 0 \wedge \dots \wedge f_{n+N-1} > 0 \implies f_{n+N} > 0)$$

does the job.

- ▶ Proof: If  $N$  has this property and  $f_0 > 0, \dots, f_{N-1} > 0$  then  $f > 0$  by induction. If not  $f_0 > 0, \dots, f_{N-1} > 0$ , then  $f \not> 0$  anyway.

# Proof by Induction: Use Knowledge

## Proof by Induction: Use Knowledge

- ▶ Method: try for  $N = 1, 2, 3, \dots$  whether

$$\forall n \geq 0 : (f_n > 0 \wedge \dots \wedge f_{n+N} > 0 \implies f_{n+N+1} > 0)$$

Stop as soon as this is the case and output the corresponding  $N$ .

## Proof by Induction: Use Knowledge

- ▶ Method: try for  $N = 1, 2, 3, \dots$  whether

$$\forall n \geq 0 : (f_n > 0 \wedge \dots \wedge f_{n+N} > 0 \implies f_{n+N+1} > 0)$$

Stop as soon as this is the case and output the corresponding  $N$ .

- ▶ Sufficient: if

$$\forall x_0, \dots, x_N : x_0 > 0 \wedge x_1 > 0 \wedge \dots \wedge x_{N-1} > 0 \implies x_N > 0$$

then  $N$  certainly qualifies.



## Proof by Induction: Use Knowledge

- ▶ Method: try for  $N = 1, 2, 3, \dots$  whether

$$\forall n \geq 0 : (f_n > 0 \wedge \dots \wedge f_{n+N} > 0 \implies f_{n+N+1} > 0)$$

Stop as soon as this is the case and output the corresponding  $N$ .

- ▶ Sufficient: if

$$\forall x_0, \dots, x_N : x_0 > 0 \wedge x_1 > 0 \wedge \dots \wedge x_{N-1} > 0 \implies x_N > 0$$

then  $N$  certainly qualifies.

- ▶ Again, this will be false if  $x_0, \dots, x_N$  are independent variables.

## Proof by Induction: Use Knowledge

- ▶ Method: try for  $N = 1, 2, 3, \dots$  whether

$$\forall n \geq 0 : (f_n > 0 \wedge \dots \wedge f_{n+N} > 0 \implies f_{n+N+1} > 0)$$

Stop as soon as this is the case and output the corresponding  $N$ .

- ▶ Sufficient: if

$$\forall x_0, \dots, x_N : x_0 > 0 \wedge x_1 > 0 \wedge \dots \wedge x_{N-1} > 0 \implies x_N > 0$$

then  $N$  certainly qualifies.

- ▶ Again, this will be false if  $x_0, \dots, x_N$  are independent variables.
- ▶ Again, we assume knowledge (e.g., defining recurrences) about  $(f_n)$  to be given, and extend the hypothesis accordingly.

## Proof by Induction: Use Knowledge

- ▶ Method: try for  $N = 1, 2, 3, \dots$  whether

$$\forall n \geq 0 : (f_n > 0 \wedge \dots \wedge f_{n+N} > 0 \implies f_{n+N+1} > 0)$$

Stop as soon as this is the case and output the corresponding  $N$ .

- ▶ Sufficient: if

$$\forall x_0, \dots, x_N : x_0 > 0 \wedge x_1 > 0 \wedge \dots \wedge x_{N-1} > 0 \implies x_N > 0$$

then  $N$  certainly qualifies.

- ▶ Again, this will be false if  $x_0, \dots, x_N$  are independent variables.
- ▶ Again, we assume knowledge (e.g., defining recurrences) about  $(f_n)$  to be given, and extend the hypothesis accordingly.
- ▶ This knowledge may be anything that gives rise to polynomial (in)equalities for the  $x_i$ .

# Proof by Induction: Use Knowledge and Computer Algebra

# Proof by Induction: Use Knowledge and Computer Algebra

- ▶ Thus we may deliver an  $N$  with

$$\begin{aligned}\forall x_0, \dots, x_{N+1} : (p_1 \leq 0, \dots, p_m \leq 0, x_0 > 0, \dots, x_N > 0) \\ \implies x_{N+1} > 0\end{aligned}$$

for certain explicit polynomials  $p_1, \dots, p_m$ .

# Proof by Induction: Use Knowledge and Computer Algebra

- ▶ Thus we may deliver an  $N$  with

$$\begin{aligned}\forall x_0, \dots, x_{N+1} : (p_1 \leq 0, \dots, p_m \leq 0, x_0 > 0, \dots, x_N > 0) \\ \implies x_{N+1} > 0\end{aligned}$$

for certain explicit polynomials  $p_1, \dots, p_m$ .

- ▶ This can be decided using *Cylindrical Algebraic Decomposition*.  $\square$

# Proof by Induction: Use Knowledge and Computer Algebra

- ▶ Thus we may deliver an  $N$  with

$$\begin{aligned} \forall x_0, \dots, x_{N+1} : (p_1 \leq 0, \dots, p_m \leq 0, x_0 > 0, \dots, x_N > 0) \\ \implies x_{N+1} > 0 \end{aligned}$$

for certain explicit polynomials  $p_1, \dots, p_m$ .

- ▶ This can be decided using *Cylindrical Algebraic Decomposition*.  $\square$
- ▶ The method can be applied to the same class of sequences as the identity prover explained before.

## Example: Bernoulli's Inequality

- ▶ Let's prove  $(z + 1)^n \geq 1 + nz$  for  $z \geq -1, n \geq 0$ .



## Example: Bernoulli's Inequality

- ▶ Let's prove  $(z + 1)^n \geq 1 + nz$  for  $z \geq -1$ ,  $n \geq 0$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (z + 1)^{n+i} \quad y_i \sim n + i \quad (i = 0, 1, 2, 3, \dots)$$

## Example: Bernoulli's Inequality

- ▶ Let's prove  $(z + 1)^n \geq 1 + nz$  for  $z \geq -1$ ,  $n \geq 0$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (z + 1)^{n+i} \quad y_i \sim n + i \quad (i = 0, 1, 2, 3, \dots)$$

- ▶ Then we know

$$\begin{aligned} x_1 &= (z + 1)x_0, & x_2 &= (z + 1)x_1, & x_3 &= (z + 1)x_2, & \dots \\ y_1 &= y_0 + 1, & y_2 &= y_1 + 1, & y_3 &= y_2 + 1, & \dots \end{aligned}$$

## Example: Bernoulli's Inequality

- ▶ Let's prove  $(z + 1)^n \geq 1 + nz$  for  $z \geq -1$ ,  $n \geq 0$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (z + 1)^{n+i} \quad y_i \sim n + i \quad (i = 0, 1, 2, 3, \dots)$$

- ▶ Then we know

$$x_1 = (z + 1)x_0, \quad x_2 = (z + 1)x_1, \quad x_3 = (z + 1)x_2, \dots$$

$$y_1 = y_0 + 1, \quad y_2 = y_1 + 1, \quad y_3 = y_2 + 1, \quad \dots$$

- ▶ First iteration ( $N = 0$ ):

$$\forall x_0, y_0, z : z \geq -1 \implies x_0 \geq 1 + y_0 z$$

## Example: Bernoulli's Inequality

- ▶ Let's prove  $(z + 1)^n \geq 1 + nz$  for  $z \geq -1, n \geq 0$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (z + 1)^{n+i} \quad y_i \sim n + i \quad (i = 0, 1, 2, 3, \dots)$$

- ▶ Then we know

$$x_1 = (z + 1)x_0, \quad x_2 = (z + 1)x_1, \quad x_3 = (z + 1)x_2, \dots$$

$$y_1 = y_0 + 1, \quad y_2 = y_1 + 1, \quad y_3 = y_2 + 1, \quad \dots$$

- ▶ First iteration ( $N = 0$ ):

$$\forall x_0, y_0, z : z \geq -1 \implies x_0 \geq 1 + y_0 z \quad \text{false.}$$

## Example: Bernoulli's Inequality

- ▶ Let's prove  $(z + 1)^n \geq 1 + nz$  for  $z \geq -1$ ,  $n \geq 0$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (z + 1)^{n+i} \quad y_i \sim n + i \quad (i = 0, 1, 2, 3, \dots)$$

- ▶ Then we know

$$x_1 = (z + 1)x_0, \quad x_2 = (z + 1)x_1, \quad x_3 = (z + 1)x_2, \dots$$

$$y_1 = y_0 + 1, \quad y_2 = y_1 + 1, \quad y_3 = y_2 + 1, \quad \dots$$

- ▶ First iteration ( $N = 0$ ):

$$\forall x_0, y_0, z : z \geq -1 \implies x_0 \geq 1 + y_0 z \quad \text{false.}$$

- ▶ Second iteration ( $N = 1$ ):

$$\begin{aligned} \forall x_0, y_0, x_1, y_1, z : z \geq -1 \wedge x_0 \geq 1 + y_0 z \wedge x_1 = (z + 1)x_0 \\ \wedge y_1 = y_0 + 1 \implies x_1 \geq 1 + y_1 z \end{aligned}$$

## Example: Bernoulli's Inequality

- ▶ Let's prove  $(z + 1)^n \geq 1 + nz$  for  $z \geq -1$ ,  $n \geq 0$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (z + 1)^{n+i} \quad y_i \sim n + i \quad (i = 0, 1, 2, 3, \dots)$$

- ▶ Then we know

$$\begin{aligned} x_1 &= (z + 1)x_0, & x_2 &= (z + 1)x_1, & x_3 &= (z + 1)x_2, & \dots \\ y_1 &= y_0 + 1, & y_2 &= y_1 + 1, & y_3 &= y_2 + 1, & \dots \end{aligned}$$

- ▶ First iteration ( $N = 0$ ):

$$\forall x_0, y_0, z : z \geq -1 \implies x_0 \geq 1 + y_0 z \quad \text{false.}$$

- ▶ Second iteration ( $N = 1$ ):

$$\begin{aligned} \forall x_0, y_0, x_1, y_1, z : z \geq -1 \wedge x_0 \geq 1 + y_0 z \wedge x_1 &= (z + 1)x_0 \\ \wedge y_1 = y_0 + 1 \implies x_1 \geq 1 + y_1 z &\quad \text{true.} \end{aligned}$$

## Example: Bernoulli's Inequality

- ▶ Let's prove  $(z + 1)^n \geq 1 + nz$  for  $z \geq -1$ ,  $n \geq 0$ .
- ▶ Introduce some variables  $x_i, y_i$  with the correspondence

$$x_i \sim (z + 1)^{n+i} \quad y_i \sim n + i \quad (i = 0, 1, 2, 3, \dots)$$

- ▶ Then we know

$$\begin{aligned} x_1 &= (z + 1)x_0, & x_2 &= (z + 1)x_1, & x_3 &= (z + 1)x_2, & \dots \\ y_1 &= y_0 + 1, & y_2 &= y_1 + 1, & y_3 &= y_2 + 1, & \dots \end{aligned}$$

- ▶ First iteration ( $N = 0$ ):

$$\forall x_0, y_0, z : z \geq -1 \implies x_0 \geq 1 + y_0 z \quad \text{false.}$$

- ▶ Second iteration ( $N = 1$ ):

$$\begin{aligned} \forall x_0, y_0, x_1, y_1, z : z \geq -1 \wedge x_0 \geq 1 + y_0 z \wedge x_1 &= (z + 1)x_0 \\ \wedge y_1 &= y_0 + 1 \implies x_1 \geq 1 + y_1 z \quad \text{true.} \end{aligned}$$

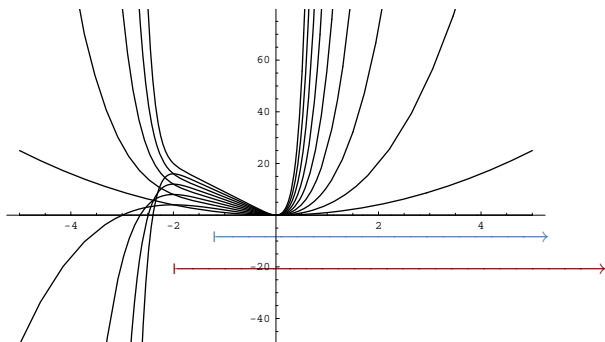
- ▶ The proof is completed by checking the claim for  $n = 0$ .





## Example: Bernoulli's Inequality

- ▶ Let's have a look at the functions  $(z + 1)^n - (1 + nz)$  for  $n = 1, 2, 3, \dots$ :



- ▶ The picture suggests that Bernoulli's inequality already holds for  $z \geq -2$ . Is this true?

## Example: Bernoulli's Inequality

- ▶ Apply the method:

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots \text{false}$ .

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$  false.

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$  false.
  - ▶  $N = 2 \dots$

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$  false.
  - ▶  $N = 2 \dots$  false.



## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$  false.
  - ▶  $N = 2 \dots$  false.
  - ▶  $N = 3 \dots$

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$  false.
  - ▶  $N = 2 \dots$  false.
  - ▶  $N = 3 \dots$  true.

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$  false.
  - ▶  $N = 2 \dots$  false.
  - ▶  $N = 3 \dots$  true.
- ▶ Now it only remains to check  $n = 0, 1, 2$ :

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$  false.
  - ▶  $N = 2 \dots$  false.
  - ▶  $N = 3 \dots$  true.
- ▶ Now it only remains to check  $n = 0, 1, 2$ :
  - ▶  $n = 0$ :  $1 \geq 1$

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$  false.
  - ▶  $N = 2 \dots$  false.
  - ▶  $N = 3 \dots$  true.
- ▶ Now it only remains to check  $n = 0, 1, 2$ :
  - ▶  $n = 0$ :  $1 \geq 1$  OK.

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$  false.
  - ▶  $N = 2 \dots$  false.
  - ▶  $N = 3 \dots$  true.
- ▶ Now it only remains to check  $n = 0, 1, 2$ :
  - ▶  $n = 0$ :  $1 \geq 1$  OK.
  - ▶  $n = 1$ :  $z + 1 \geq z + 1$

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$  false.
  - ▶  $N = 2 \dots$  false.
  - ▶  $N = 3 \dots$  true.
- ▶ Now it only remains to check  $n = 0, 1, 2$ :
  - ▶  $n = 0$ :  $1 \geq 1$  OK.
  - ▶  $n = 1$ :  $z + 1 \geq z + 1$  OK.

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$  false.
  - ▶  $N = 2 \dots$  false.
  - ▶  $N = 3 \dots$  true.
- ▶ Now it only remains to check  $n = 0, 1, 2$ :
  - ▶  $n = 0$ :  $1 \geq 1$  OK.
  - ▶  $n = 1$ :  $z + 1 \geq z + 1$  OK.
  - ▶  $n = 2$ :  $z^2 + 2z + 1 \geq 1 + 2z$



## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$  false.
  - ▶  $N = 2 \dots$  false.
  - ▶  $N = 3 \dots$  true.
- ▶ Now it only remains to check  $n = 0, 1, 2$ :
  - ▶  $n = 0$ :  $1 \geq 1$  OK.
  - ▶  $n = 1$ :  $z + 1 \geq z + 1$  OK.
  - ▶  $n = 2$ :  $z^2 + 2z + 1 \geq 1 + 2z$  OK.  $\square$ .

## Example: Bernoulli's Inequality

- ▶ Apply the method:
  - ▶  $N = 0 \dots$  false.
  - ▶  $N = 1 \dots$  false.
  - ▶  $N = 2 \dots$  false.
  - ▶  $N = 3 \dots$  true.
- ▶ Now it only remains to check  $n = 0, 1, 2$ :
  - ▶  $n = 0$ :  $1 \geq 1$  OK.
  - ▶  $n = 1$ :  $z + 1 \geq z + 1$  OK.
  - ▶  $n = 2$ :  $z^2 + 2z + 1 \geq 1 + 2z$  OK.  $\square$ .
- ▶ Conclusion: We have generalized Bernoulli's inequality.

# Some further Examples

## Some further Examples

$$\blacktriangleright \prod_{k=0}^n \frac{3k+4}{3k+2} > 1 + \frac{2}{3} \sum_{k=1}^{n+1} \frac{1}{k}$$

## Some further Examples

$$\blacktriangleright \prod_{k=0}^n \frac{3k+4}{3k+2} > 1 + \frac{2}{3} \sum_{k=1}^{n+1} \frac{1}{k}$$

$$\blacktriangleright \sum_{k=1}^n \frac{(2F_{k+1} - F_k)^2}{F_k} \geq \frac{(3F_{n+1} + F_n - 3)^2}{F_{n+2} - 1}$$

## Some further Examples

$$\blacktriangleright \prod_{k=0}^n \frac{3k+4}{3k+2} > 1 + \frac{2}{3} \sum_{k=1}^{n+1} \frac{1}{k}$$

$$\blacktriangleright \sum_{k=1}^n \frac{(2F_{k+1} - F_k)^2}{F_k} \geq \frac{(3F_{n+1} + F_n - 3)^2}{F_{n+2} - 1}$$

$$\blacktriangleright \left( \sum_{k=1}^n x_k y_k \right)^2 \leq \sum_{k=1}^n x_k^2 \sum_{k=1}^n y_k^2$$

## Some further Examples

- ▶ 
$$\prod_{k=0}^n \frac{3k+4}{3k+2} > 1 + \frac{2}{3} \sum_{k=1}^{n+1} \frac{1}{k}$$
- ▶ 
$$\sum_{k=1}^n \frac{(2F_{k+1} - F_k)^2}{F_k} \geq \frac{(3F_{n+1} + F_n - 3)^2}{F_{n+2} - 1}$$
- ▶ 
$$\left( \sum_{k=1}^n x_k y_k \right)^2 \leq \sum_{k=1}^n x_k^2 \sum_{k=1}^n y_k^2$$
- ▶ 
$$\prod_{k=1}^{n-1} (a^k + 1) < \frac{1-a}{a^n - 2a + 1} \quad (\text{for } 0 < a < \frac{1}{2})$$

## Some further Examples

- ▶ 
$$\prod_{k=0}^n \frac{3k+4}{3k+2} > 1 + \frac{2}{3} \sum_{k=1}^{n+1} \frac{1}{k}$$
- ▶ 
$$\sum_{k=1}^n \frac{(2F_{k+1} - F_k)^2}{F_k} \geq \frac{(3F_{n+1} + F_n - 3)^2}{F_{n+2} - 1}$$
- ▶ 
$$\left( \sum_{k=1}^n x_k y_k \right)^2 \leq \sum_{k=1}^n x_k^2 \sum_{k=1}^n y_k^2$$
- ▶ 
$$\prod_{k=1}^{n-1} (a^k + 1) < \frac{1-a}{a^n - 2a + 1} \quad (\text{for } 0 < a < \frac{1}{2})$$
- ▶ 
$$P_{n+1}(x)^2 - P_n(x)P_{n+2}(x) \geq 0 \quad (\text{for } -1 \leq x \leq 1)$$



## Some further Examples

- ▶ 
$$\prod_{k=0}^n \frac{3k+4}{3k+2} > 1 + \frac{2}{3} \sum_{k=1}^{n+1} \frac{1}{k}$$
- ▶ 
$$\sum_{k=1}^n \frac{(2F_{k+1} - F_k)^2}{F_k} \geq \frac{(3F_{n+1} + F_n - 3)^2}{F_{n+2} - 1}$$
- ▶ 
$$\left( \sum_{k=1}^n x_k y_k \right)^2 \leq \sum_{k=1}^n x_k^2 \sum_{k=1}^n y_k^2$$
- ▶ 
$$\prod_{k=1}^{n-1} (a^k + 1) < \frac{1-a}{a^n - 2a + 1} \quad (\text{for } 0 < a < \frac{1}{2})$$
- ▶ 
$$P_{n+1}(x)^2 - P_n(x)P_{n+2}(x) \geq 0 \quad (\text{for } -1 \leq x \leq 1)$$
- ▶ ...

# Some Computational Theoretic Remarks

## Some Computational Theoretic Remarks

- ▶ Is this a Decision Procedure?

## Some Computational Theoretic Remarks

- ▶ Is this a Decision Procedure?
  - ▶ No. There are examples where the procedure does not terminate.

## Some Computational Theoretic Remarks

- ▶ Is this a Decision Procedure?
  - ▶ No. There are examples where the procedure does not terminate.
  - ▶ But a decision procedure is too much to hope for.

## Some Computational Theoretic Remarks

- ▶ Is this a Decision Procedure?
  - ▶ No. There are examples where the procedure does not terminate.
  - ▶ But a decision procedure is too much to hope for.
  - ▶ If a decision procedure existed, we could also decide  $\exists n : f_n = 0$  (root finding), by simply applying the algorithm to  $f_n^2 > 0$ .

## Some Computational Theoretic Remarks

- ▶ Is this a Decision Procedure?
  - ▶ No. There are examples where the procedure does not terminate.
  - ▶ But a decision procedure is too much to hope for.
  - ▶ If a decision procedure existed, we could also decide  $\exists n : f_n = 0$  (root finding), by simply applying the algorithm to  $f_n^2 > 0$ .
  - ▶ Already for small classes of sequences, subincluded in ours, it is open whether root finding is decidable.

## Some Computational Theoretic Remarks

- ▶ Is this a Decision Procedure?
  - ▶ No. There are examples where the procedure does not terminate.
  - ▶ But a decision procedure is too much to hope for.
  - ▶ If a decision procedure existed, we could also decide  $\exists n : f_n = 0$  (root finding), by simply applying the algorithm to  $f_n^2 > 0$ .
  - ▶ Already for small classes of sequences, subincluded in ours, it is open whether root finding is decidable.
- ▶ Then, is it a Semi Decision Procedure?



## Some Computational Theoretic Remarks

- ▶ Is this a Decision Procedure?
  - ▶ No. There are examples where the procedure does not terminate.
  - ▶ But a decision procedure is too much to hope for.
  - ▶ If a decision procedure existed, we could also decide  $\exists n : f_n = 0$  (root finding), by simply applying the algorithm to  $f_n^2 > 0$ .
  - ▶ Already for small classes of sequences, subincluded in ours, it is open whether root finding is decidable.
- ▶ Then, is it a Semi Decision Procedure?
  - ▶ Also not.

## Some Computational Theoretic Remarks

- ▶ Is this a Decision Procedure?
  - ▶ No. There are examples where the procedure does not terminate.
  - ▶ But a decision procedure is too much to hope for.
  - ▶ If a decision procedure existed, we could also decide  $\exists n : f_n = 0$  (root finding), by simply applying the algorithm to  $f_n^2 > 0$ .
  - ▶ Already for small classes of sequences, subincluded in ours, it is open whether root finding is decidable.
- ▶ Then, is it a Semi Decision Procedure?
  - ▶ Also not.
  - ▶ Because it can be semidecided that an inequality does not hold (enumerate all  $n \geq 0$  in search of a counterexample)

## Some Computational Theoretic Remarks

- ▶ Is this a Decision Procedure?
  - ▶ No. There are examples where the procedure does not terminate.
  - ▶ But a decision procedure is too much to hope for.
  - ▶ If a decision procedure existed, we could also decide  $\exists n : f_n = 0$  (root finding), by simply applying the algorithm to  $f_n^2 > 0$ .
  - ▶ Already for small classes of sequences, subincluded in ours, it is open whether root finding is decidable.
- ▶ Then, is it a Semi Decision Procedure?
  - ▶ Also not.
  - ▶ Because it can be semidecided that an inequality does not hold (enumerate all  $n \geq 0$  in search of a counterexample)
  - ▶ Together with a semi decision procedure for proving inequalities, we would obtain a decision procedure.

## Some Computational Theoretic Remarks

- ▶ Is this a Decision Procedure?
  - ▶ No. There are examples where the procedure does not terminate.
  - ▶ But a decision procedure is too much to hope for.
  - ▶ If a decision procedure existed, we could also decide  $\exists n : f_n = 0$  (root finding), by simply applying the algorithm to  $f_n^2 > 0$ .
  - ▶ Already for small classes of sequences, subincluded in ours, it is open whether root finding is decidable.
- ▶ Then, is it a Semi Decision Procedure?
  - ▶ Also not.
  - ▶ Because it can be semidecided that an inequality does not hold (enumerate all  $n \geq 0$  in search of a counterexample)
  - ▶ Together with a semi decision procedure for proving inequalities, we would obtain a decision procedure.
- ▶ Then, What is it?

## Some Computational Theoretic Remarks

- ▶ Is this a Decision Procedure?
  - ▶ No. There are examples where the procedure does not terminate.
  - ▶ But a decision procedure is too much to hope for.
  - ▶ If a decision procedure existed, we could also decide  $\exists n : f_n = 0$  (root finding), by simply applying the algorithm to  $f_n^2 > 0$ .
  - ▶ Already for small classes of sequences, subincluded in ours, it is open whether root finding is decidable.
- ▶ Then, is it a Semi Decision Procedure?
  - ▶ Also not.
  - ▶ Because it can be semidecided that an inequality does not hold (enumerate all  $n \geq 0$  in search of a counterexample)
  - ▶ Together with a semi decision procedure for proving inequalities, we would obtain a decision procedure.
- ▶ Then, What is it?
  - ▶ It's just a method that often succeeds.