1: Grundlagen
1.1: Natürliche Zahlen

Es gibt drei Arten von Menschen:

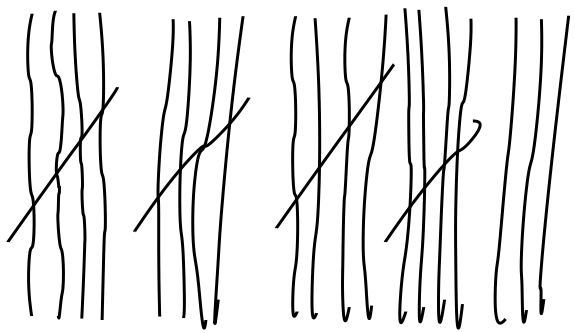
- die, die bis drei zählen können,
- und die anderen.

Aus dem “Zählen” wurden die folgenden Konzepte entwickelt:

(A) Menge $\mathbb{N} = \{1, 2, 3, \ldots\}$ der *natürlichen Zahlen*

(B) Arithmetik: Addition $a + b$, Multiplikation $a * b$, Potenz-Bildung b^a und Umkehroperationen

(C) Vergleiche: “=”, “$\neq$”, “<”, “>”, “$\leq$”, “$\geq$”



(A) Die Peano-Axiome

Eine formale Definition der natürlichen Zahlen

1. 1 ist eine natürliche Zahl.
2. Jede natürliche Zahl besitzt eine eindeutig bestimmte natürliche Zahl als Nachfolger.
3. 1 ist nicht Nachfolger einer natürlichen Zahl.
4. Verschiedene natürliche Zahlen haben verschiedene Nachfolger. Es gibt keine anderen natürlichen Zahlen.
5. Ist eine Aussage, die von einer natürlichen Zahl abhängt, wahr, wenn diese Zahl 1 ist, und ist sie außerdem wahr für den Nachfolger einer natürlichen Zahl, wenn sie für diese natürliche Zahl wahr ist, dann ist sie für alle natürlichen Zahlen wahr.

Folgerung: Jede natürliche Zahl außer der 1 hat einen “Vorgänger”.

(B) Arithmetik

- Addition $a + b$:
  - $a + 1 =$ (Nachfolger von a)
  - $a + b =$ (Nachfolger von a) + (Vorgänger von b), falls b einen Vorgänger hat
- Multiplikation $a * b$:
  - $a * 1 = a$
  - $a * b = a +$ (a*(Vorgänger von b)), falls b einen Vorgänger hat
- Potenzbildung a hoch b:
  - a hoch $1 = a$
  - a hoch $b = a *$ (a hoch (Vorgänger von b)), falls b einen Vorgänger hat
- Subtraktion, Division, und Logarithmierung als Umkehroperationen von Addition, Multiplikation und Potenzbildung

Arithmetische Gesetze

- Assoziativgesetze:
  - Addition: $a + (b + c) = (a + b) + c$
  - Multiplikation: $a(bc) = (ab)c$.
- Kommutativgesetze:
  - Addition: $a + b = b + a$ und
  - Multiplikation: $ab = ba$.
- Neutrales Element:
  - (Nur) Multiplikation: $a1 = 1a = a$
- Distributivgesetz:
  - $a(b + c) = ab + ac$
- (Konventionen):
  - (Weglassen des "$*$"-Operators)
  - (Punkt-vor-Strich)

(C) Vergleiche

- $1 = 1$
- $a > 1$ und $1 < a$, falls a einen Vorgänger hat
- wenn a und b jeweils einen Vorgänger haben:
  - $a = b \Leftrightarrow$ (Vorgänger von a) $=$ (Vorgänger von b)
  - $a < b \Leftrightarrow$ (Vorgänger von a) $<$ (Vorgänger von b)
  - $a > b \Leftrightarrow$ (Vorgänger von a) $>$ (Vorgänger von b)

Ordnungsgesetze

- Trichotomie: Für $a, b \in \mathbb{N}$ gilt genau eine der drei Beziehungen: "$a < b$", "$a = b$" oder "$a > b$".
- Transitivität: $(a < b \text{ und } b < c) \Rightarrow a < c$.
- Verträglichkeit: Für $a, b, c \in \mathbb{N}$ gilt
  - Additiv: $a < b \Rightarrow (a + c < b + c)$.
  - Multiplikativ: $a < b \Rightarrow (ac < bc)$.

Die Peano-Axiome (anders dargestellt)

Die ersten vier Axiome sind “offensichtlich richtig”.

1. $1 \in \mathbb{N}$.
2. $n \in \mathbb{N} \Longrightarrow n+1 \in \mathbb{N}$.
3. $\forall n \in \mathbb{N} : n+1 \neq 1$. (Anders ausgedrückt: $0 \notin \mathbb{N}$.)
4. $(a, b \in \mathbb{N}$ und $a \neq b) \Longrightarrow a+1 \neq b+1$.

Das fünfte ist merkwürdig. Doch ohne dieses “Induktionsaxiom” würden viele mathematische Beweise nicht funktionieren:

5. Sei $P(n)$ ein von einer natürlichen Zahl n abhängiges Prädikat. Dann gilt:

$$(P(1) \wedge (P(n) \to P(n+1))) \implies (\forall n \in \mathbb{N} : P(n)) .$$

Ein “Induktionsbeweis”

Die folgende Aussage können wir induktiv beweisen:

Satz 1

Sei $n \in \mathbb{N}$. Dann ist

$$\sum_{1 \leq i \leq n} i = \frac{n(n+1)}{2}.$$

1.2: Mathematische Beweise

Neben der Induktion gibt es andere wichtige Beweisprinzipien:

- Direkte Beweise
- Indirekte Beweise
- Mischformen von mehr als einer Beweistechnik

Direkte Beweise (Beispiele)

Satz 2

Sei $n \in \mathbb{N}$. Dann ist $n^2 \geq n$. Insbesondere gilt für $n \neq 1$: $n^2 > n$.

Beweis.

Die Ungleichung $n^2 \geq n$ kann man durch n teilen.
Dann erhält man die (äquivalente) Aussage $n \geq 1$. □

Satz 3

Das Quadrat einer ungeraden natürlichen Zahl ist ungerade.

Noch ein direkter Beweis

Satz 4

Die folgenden beiden Aussagen sind logisch gleich:

- *“aus A folgt B”*
- *“aus (nicht B) folgt (nicht A)”.*

Beweis.

Zum Beweis geben wir die Wahrheitstabelle an:

A	B	$A \to B$	$\overline{A}$	$\overline{B}$	$\overline{B} \to \overline{A}$
f	f	w	w	w	w
f	w	w	w	f	w
w	f	f	f	w	f
w	w	w	f	f	w

(w:= wahr, f:= falsch)

□

Indirekte Beweise (Vorbemerkung)

Indirekte Beweise nutzen Satz 4:

1. Sei A bewiesen bzw. vorausgesetzt.
   Wir wollen beweisen, dass, wenn A gilt, dann auch B gilt.
2. Dafür tun wir so, als würde "nicht B" gelten ("Annahme").
3. Wir zeigen, dass *dann* auch "nicht A" gilt ("Widerspruch").
4. Damit ist "aus A folgt B" eine bewiesene Tatsache.
5. Ist A selbst eine bewiesene Tatsache,
   dann ist jetzt auch B eine bewiesene Tatsache.

Indirekter Beweis (Beispiele)

Satz 5

Ist die natürliche Zahl m die Quadratwurzel einer geraden natürlichen Zahl, dann ist m selbst gerade.

Beweis.

Voraussetzung: $m = \sqrt{n} \in \mathbb{N}$ und n ist gerade.
Zu zeigen: "m ist gerade".

Annahme des Gegenteils: "m ist ungerade".
Mit Satz 3 folgt: dann ist auch $n = m^2$ ungerade.
Das widerspricht "n ist gerade". □

Satz 6

Sei n eine Primzahl. Dann ist $\sqrt{n} \notin \mathbb{N}$.

Beweis durch Indirektion *und* Induktion

Satz 7

Sei A eine nichtleere Teilmenge der natürlichen Zahlen. Dann enthält A genau ein kleinstes Element.

Bemerkung

Die Aussage von Satz 7 mag Leuten mit gesundem Menschenverstand “offensichtlich” erscheinen – auch ohne mathematisches Fachwissen. Tatsächlich gibt es *ein kleinstes und sogar ein größtes Element* , wenn A endlich ist (dank der Ordnungsgesetze).

Bei unendlichen Mengen stößt der gesunde Menschenverstand aber an seine Grenzen:

- Ist $A \subseteq \mathbb{N}$ *unendlich*, gibt es kein größtes Element in A.
- Ist $B \subseteq \mathbb{Z}$, dann sind sowohl die Existenz eines kleinsten als auch die Existenz eines größten Elements fraglich. Man denke z.B. an $B = \{\ldots, -6, -4, -2, 0, 2, 4, 6, \ldots\}$.

1.3: Funktionen

[. . .] eine Beziehung zwischen zwei Mengen, die jedem Element der einen Menge (Funktionsargument, [...]) genau ein Element der anderen Menge (Funktionswert, [...]) zuordnet. (Zitat aus Wikipedia)

- Eine Menge $f \subseteq A \times B$ ist eine *Funktion*
  wenn zu jedem $a \in A$ *genau ein* $b \in B$ mit $(a, b) \in f$ existiert.
- Wir schreiben $f(a)$ für b und $f : A \to B$ für $f \subseteq A \times B$.
- A ist der *Definitionsbereich* von f, B der *Wertebereich*.
- Ist $A = A_1 \times \cdots \times A_n$ schreiben wir $f : A_1 \times \cdots \times A_n \to B$
  und $f(a_1, \ldots a_n)$.
- Beispiele:
  - Funktionen $\mathbb{N} \to \mathbb{N}$: $f(x) = x^2$, $h(x) = x + 1$.
  - Funktionen $\mathbb{N} \times \mathbb{N} \to \mathbb{N}$: $a(x, y) = x + y$, $c(x, y) = 1$.
  - Funktionen $\mathbb{N} \times \mathbb{N} \to \{\text{wahr}, \text{falsch}\}$: $e(x, y) = (x = y)$,
    $g(x, y) = (x > y)$

Permutationen

- Eine Funktion $f : A \to B$ ist eine *Permutation*,
  falls für jedes $b \in B$ *genau ein* $a \in A$ mit $f(a) = b$ gibt.
- Ist f eine Permutation und $b = f(a)$, schreiben wir auch $a = f^{-1}(b)$. Wir nennen f^{-1} die *Umkehrfunktion* von f.
- Beispiele:
  - $h : \mathbb{Z} \to \mathbb{Z}$, $h(x) = x + 1$ ist eine Permutation mit der Umkehrfunktion $h^{-1}(x) = x - 1$.
  - $h : \mathbb{N} \to \mathbb{N}$, $h(x) = x + 1$ ist keine Permutation. (Warum nicht?)
  - $d : \mathbb{N} \to \mathbb{N}$, $d(x) = x \operatorname{div} 2$ ist auch keine Permutation. (Warum nicht?)
- Andere Formulierung:
  Eine Funktion f ist genau dann *keine* Permutation, wenn ein b in B existiert, für das kein $a \in A$ existiert mit $f(a) = b$, *oder* wenn $a, a' \in A$ existieren mit $a \neq a'$ aber $f(a) = f(a')$.

1.4: Eine Woche in Hilberts Hotel – ein Dramolett

“Hilberts Hotel ist ein vom Mathematiker **David Hilbert** erdachtes Beispiel zur Veranschaulichung verblüffender Konsequenzen der Nutzung des Unendlichkeitsbegriffes.” Siehe ⟨http://de.wikipedia.org/wiki/Hilberts_Hotel⟩. Hilberts Hotel wurde sogar verfilmt.

Ich habe aus der Idee ein kleines Theaterstück gemacht.

Bühnenbild: *(Rezeption eines Hotels.)*

Besetzung: *(**Alice** und **Bob** in Hotel-Uniformen. **Stimmen** aus dem Off.)*

Erster Tag

Alice: *(verträumt)* Ausgebucht! Unendlich viele Zimmer, und keins ist frei. $\forall n \in \mathbb{N}$: (Zimmer n ist belegt).

Bob: Schau mal, es kommt noch ein Gast!

Weibliche Stimme: Brr, furchtbares Wetter! Ich hoffe, Sie sind noch nicht ausgebucht?

Bob: Doch! *(grinst)* Keine Sorge, trotzdem haben wir Platz für Sie! Wir werfen auch niemanden raus, und keiner muss sich ein Zimmer mit irgendwem teilen.

Weibliche Stimme: Das geht nicht! *(hoffnungsvoll)* Oder doch?

Alice: Jeder Gast wird gebeten, von Zimmer i in Zimmer $i + 1$ umzuziehen. Schon ist Zimmer 1 frei – für Sie!

Zweiter Tag – Ein Bus kommt

Bob: Oh, da ist ein Bus mit Gästen angekommen.

Viele Stimmen: Hallo, hallo, hallo? Haben Sie Zimmer für uns?

Alice: Wie viele sind Sie denn?

Viele Stimmen: Wir sind unendlich viele.

Bob: Können Sie durchzählen? “Eins, Zwei, Drei, . . . ”?

Viele Stimmen: Ja, wir sind abzählbar.

Bob: Dann zählen Sie bitte durch. Wer Nummer i hat, wird in Zimmer $2i - 1$ untergebracht.

Alice: Dazu müssen wir natürlich alle ungerade-zahligen Zimmer frei machen: Jeder Gast in Zimmer i wird gebeten, in Zimmer $2i$ umzuziehen.

Abzählbarkeit

Definition 8

Jede endliche Menge ist *abzählbar*. Eine unendliche Menge M ist *abzählbar*, wenn eine Permutation $\pi : M \to \mathbb{N}$ existiert.

Jede Teilmenge einer endlichen Menge ist abzählbar. (Beweis bald!)

Satz 9

Die Mengen $\mathbb{N}, \mathbb{N}_0$ und $\mathbb{Z}$ sind abzählbar.

Beweisidee

Man überzeuge sich, das folgenden Funktionen Permutationen sind: $\pi_1 : \mathbb{N} \to \mathbb{N},\ \pi_1(x) = x$, $\pi_2 : \mathbb{N}_0 \to \mathbb{N},\ \pi_2(x) = x + 1$ und

$$\pi_3 : \mathbb{Z} \to \mathbb{N},\ \pi_3(x) = \begin{cases} 2x + 1 & \text{für } x \geq 0 \text{ und} \\ -2x & \text{für } x < 0. \end{cases}$$

Dritter Tag – viele Busse kommen

Männliche Stimme: Hallihallo! Cantor-Reisen ist wieder da!

Alice und Bob: Guten Tag, Georg.

Männliche Stimme: Wir haben abzählbar viele Busse. In jedem Bus sitzen abzählbar viele Gäste. Euer Chef hat uns versprochen, alle unsere Gäste unterzubringen.

Bob: *(zweifelnd)* Aaaalso ...

Alice: *(triumphal)* Ha! Georg, mit Deinem ersten Diagonalisierungsargument ist das kein Problem!

Überraschung?

Satz 10

Satz: $\mathbb{N} \times \mathbb{N}$ *ist abzählbar.*

(Beweisskizze folgt gleich.)

Aus diesem Satz folgt:

Folgerung 11

Die Menge $\mathbb{Z} \times \mathbb{N}$ *ist abzählbar.*

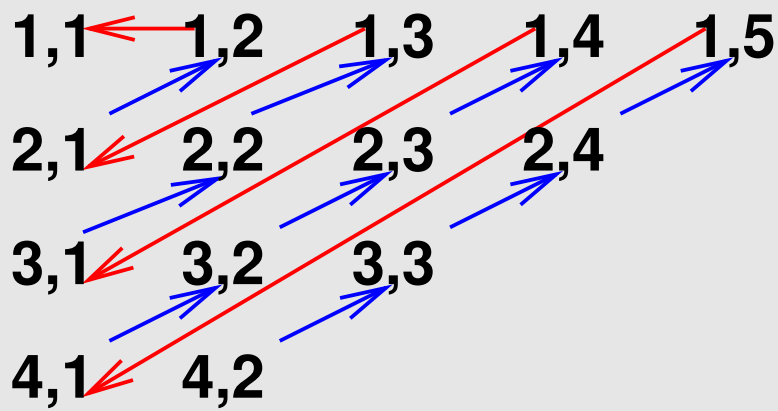
Beweisidee:

Wir definieren eine Permutation $\pi : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$:

$$\pi(a,b) = \begin{cases} 1 & \text{falls } a = 1 \text{ und } b = 1, \\ \mathbf{1} + \pi(\mathbf{b-1}, \mathbf{1}) & \text{falls } a = 1 \text{ und } b > 1 \text{ und} \\ \mathbf{1} + \pi(\mathbf{a-1}, \mathbf{b+1}) & \text{falls } a > 1. \end{cases}$$

Zum Beispiel ist

$$\begin{aligned} \pi(4,2) &= \mathbf{1} + \pi(\mathbf{3},\mathbf{3}) \\ &= \mathbf{2} + \pi(\mathbf{2},\mathbf{4}) \\ &= \mathbf{3} + \pi(\mathbf{1},\mathbf{5}) \\ &= \mathbf{4} + \pi(\mathbf{4},\mathbf{1}) \\ &= \mathbf{5} + \pi(\mathbf{3},\mathbf{2}) \\ &= \dots \\ &= \mathbf{13} + \pi(\mathbf{1},\mathbf{1}) \\ &= 14. \end{aligned}$$



Vierter Tag – eine defekte Sprechanlage

Alice: Ab heute gilt ein allgemeines Rauchverbot in allen Zimmern. Und die Sprechanlage ist kaputt! Wir bräuchten buchstäblich ewig, um allen Gästen mitzuteilen, dass sie nicht mehr rauchen dürfen.

Bob: Kein Problem! Das machen wir mit Induktion.

Alice: Häh?

Bob: Wir schreiben eine Tafel, auf der steht:

1. *Bitte beachten Sie, dass Ihr Zimmer ab sofort ein Nichtraucherzimmer ist.*
2. *Wenn i die Nummer Ihres Zimmers ist, geben Sie diese Tafel bitte in Zimmer $i + 1$ ab.*

Wir geben dann nur die Tafel in Zimmer 1 ab.

Alice: Klar! Den Rest erledigen unsere Gäste selbst!

Zur Erinnerung

Induktionsaxiom

Sei $P(n)$ ein von einer natürlichen Zahl n abhängiges Prädikat. Dann gilt:

$$(P(1) \wedge (P(n) \to P(n+1))) \implies (\forall n \in \mathbb{N} : P(n)) .$$

Fünfter Tag: Fast alle Gäste reisen ab

Alice: Heute haben wir ein Problem! Alle Gäste, sind abgereist – nur die in Primzahl-Zimmern sind geblieben. Es gibt doch nur einige wenige Primzahlen, und viel mehr andere Zahlen!
(weint) Wir sind pleite – das Hotel ist so gut wie leer!

Bob: Ach, hast Du in Zahlentheorie nicht aufgepasst?

Alice: *(hört auf zu weinen)* Warum?

Bob: Es gibt unendlich viele Primzahlen. Also gibt es genauso viele Primzahlen, wie natürliche Zahlen.
Wenn p_i die i-te Primzahl ist, soll der Gast von Zimmer p_i in Zimmer i umziehen. Schon sind alle unsere Zimmer wieder belegt.

Alice: *(seufzt erleichtert)*

Primzahlen

Definition 12

Eine *Primzahl* ist eine natürliche Zahl, die genau zwei natürliche Teiler hat.

Eine *zusammengesetzte Zahl* hat mehr als zwei natürliche Teiler.

Folgerung 13

Die 1 ist weder eine Primzahl, noch ist sie zusammengesetzt.

Satz 14

Es gibt unendlich viele Primzahlen.

(Indirekter Beweis; verwendet den folgenden Hilfssatz.)

Hilfssatz

Seien x und y natürliche Zahlen. Ist x durch y teilbar und $y > 1$, dann ist $x + 1$ eine natürliche Zahl, die nicht durch y teilbar ist.

Teilmengen abzählbarer Mengen

Satz 15

Jede Teilmenge einer abzählbaren Menge ist abzählbar.
Insbesondere ist die Menge aller Primzahlen abzählbar.

Beweisidee:

Sei M eine unendliche Teilmenge von $\mathbb{N}$. Wir zeigen, dass M abzählbar ist. (Endliche Mengen sind sowieso abzählbar.)
Zu jedem $m \in M$ sei $\sigma(m) = |\{x \in M | x \leq m\}|$ die Anzahl an Elementen aus M, die kleiner oder gleich m sind.
Für jedes $i \in \mathbb{N}$ gibt es genau ein $m \in M$ mit $\sigma(m) = i$.

- Für $i = 1$ ist m das kleinste Element aus M.
- Für $i > 1$ ist m das kleinste Element aus $\{y \in M | \sigma(y) \geq i\}$.

Wir hatten bereits bewiesen, dass jede nichtleere Teilmenge von $\mathbb{N}$ stets genau ein kleinstes Element enthält.

Fünfter Tag (2)

Alice: Bob, wenn wir nicht noch mehr Gäste verlieren wollen, dürfen wir sie nicht mehr dauernd in andere Zimmer verlegen oder ihnen die Weitergabe irgendwelcher Tafeln abverlangen.

Bob: Und, was sollen wir tun? Bei Regenwetter einen Gast abweisen, nur weil unser Hotel zufällig besetzt ist? Das ist doch unfair – dann wären wir nicht besser als jedes beliebige endliche Hotel!

Alice: Nein. Heute verteilen wir unsere jetzigen Gäste nicht auf $1, 2, 3, 4, \ldots$ sondern auf $2^1, 2^2, 2^2, 2^4, \ldots$, Neuankömmlinge morgen dann auf $3^1, 3^2, 3^3, 3^4, \ldots$, und übermorgen ...

Bob: ... und dann geht's schief: $4^1 = 2^2$, $4^2 = 2^4$ ... – die Zimmer sind bereits belegt! Schon müssen wieder Gäste umziehen.
(überheblich) Erst Nachdenken, Alice, dann reden!

Fünfter Tag (3)

Alice: *(noch überheblicher)* Selber!
Übermorgen verteilen wir die Neuankömmlinge auf die Zimmer $5^1, 5^2, 5^3, 5^4, \ldots$. Sei p_i die i-te Primzahl. Am i-ten Tag (ab heute) bringen wir unsere Gäste bzw. Neuankömmlinge auf den Zimmern $p_i^1, p_i^2, p_i^3, p_i^4, \ldots$ unter.

Bob: *(staunend)* Alice! *(Kurze Pause)*
Du bist genial! Für jeden neuen Gast haben wir ein Zimmer frei, und nie wieder muss ein Gast das Zimmer wechseln, denn für $i \neq j$ und beliebige $a, b \in \mathbb{N}$ gilt:

$$p_i^a \neq p_j^b.$$

Natürlich haben wir unendlich viele Zimmer, die nie belegt werden können: $\{1, 6, 10, 12, 14, 15, 18, 20, 21, 22, \ldots\}$

Alice: Aber solange unendlich viele Zimmer belegt sind, ist egal, wie viele Zimmer frei sind ...

Sechster Tag

Alice: Du, Bob?

Bob: Ja?

Alice: Warum ist die 1 keine Primzahl?

Bob: Weil man das so definiert hat.

Alice: Klar! Ich meine, warum hat man die Primzahlen extra so definiert, dass die 1 keine ist?

Bob: Hm! *(Kurze Pause)*
Mathematische Definitionen sind kein Selbstzweck – sie motivieren sich aus bestimmten Fragestellungen. Bei den Primzahlen ist die Fragestellung, wie man natürliche Zahlen eindeutig in elementare Faktoren zerlegen kann, z.B.: $6 = 2 * 3$, $7 = 7$, $8 = 2 * 2 * 2$, $9 = 3 * 3$, $10 = 2 * 5$, $11 = 11$, $12 = 2 * 2 * 3$, usw.

Alice: Ah! Die 1 würde die Eindeutigkeit zerstören:
$6 = 2 * 3 = 1 * 2 * 3 = 1 * 1 * 2 * 3$ usw.

Was Bob meint:

Satz (Eindeutigkeit der Primfaktorzerlegung)

Jede natürliche Zahl $n \geq 2$ kann als Produkt von einer oder mehreren Primzahlen dargestellt werden. Bis auf die Reihenfolge der Primzahlen ist diese Darstellung eindeutig.

(Ohne Beweis)

Folgerung 16

Die 1 ist die einzige natürliche Zahl, die weder Primzahl noch zusammengesetzt ist.

Letzter Tag – endlich Ruhe

Alice: *(entspannt)* Keiner kommt, keiner geht. Nach all dem Stress diese Woche gefällt mir das jetzt!

Bob: Ich habe nachgedacht. Du hattest vor einigen Tagen ein “erstes Diagonalisierungsargument” von Georg Cantor benutzt.

Alice: Und?

Bob: Gibt es unendlich viele Diagonalisierungsargumente?

Alice: Ich kenne nur ein zweites – auch von Georg.

Bob: Und was besagt das?

Alice: Man kann die reellen Zahlen nicht abzählen.

Bob: Ach nein? Warum denn nicht?

Über-Abzählbarkeit

Da die Teilmenge einer abzählbaren Menge auch abzählbar ist, muss eine Menge, die nicht abzählbar ist, unendlich und irgendwie “größer” als alle abzählbaren Mengen sein:

Eine unendliche Menge, die nicht abzählbar ist, nennen wir *über-abzählbar.*

Bleibt die Frage, ob es überhaupt über-abzählbare Mengen gibt!

Oder kann man jede Menge abzählen?

Satz 17

Die Menge $\mathbb{R}$ und die Menge $2^{\mathbb{N}}$ aller Teilmengen von $\mathbb{N}$ sind beide über-abzählbar.

(Wir skizzieren den Beweis für $2^{\mathbb{N}}$.)

Paradox? $2^{\mathbb{N}}$ ist eine überabzählbare Menge von abzähbaren Mengen!

Beweisidee (erster Teil):

Annahme: Die Menge aller Teilmengen von $\mathbb{N}$ ist abzählbar.
(Achtung: Widerspruchsbeweis!)

Jede Menge $T \subseteq \mathbb{N}$ kann man als abzählbar lange Folge von “Bits” aus $\{0, 1\}$ schreiben. Das i-te Bit der Folge ist $1 \Leftrightarrow i \in T$, z. B.:

Gerade Zahlen:	010101010...
Ungerade Zahlen:	101010101...
Primzahlen:	011010100...
Quadratzahlen:	100100001...

Beweisidee (2. Teil):

Aufgrund der Annahme, dass die Menge aller Teilmengen abzählbar ist, kann man eine nummerierte Liste aller Teilmengen von $\mathbb{N}$ anlegen, z.B.:

$$\begin{array}{ll} T_1: & \mathbf{0}10101010\ldots \\ T_2: & 1\mathbf{0}1010101\ldots \\ T_3: & 01\mathbf{1}010100\ldots \\ T_4: & 100\mathbf{1}00001\ldots \\ \vdots & \vdots \end{array}$$

Die **Hauptdiagonale** dieser Liste besteht aus den Werten $T_1(1), T_2(2), \ldots$, z.B. **0011...**

Beweisidee (Schluss):

Invertieren der Hauptdiagonalen ergibt eine Folge T^*:

$$T^*(i) = \begin{cases} 1 & T_i(i) = 0 \\ 0 & T_i(i) = 1 \end{cases},$$

z.B.: T^*=1100...

- Ebenso wie die Menge T_i stellt T^* eine Teilmenge von $\mathbb{N}$ dar.
- Deshalb muss T^* in der Liste enthalten sein.
- Also gibt es ein $j \in \mathbb{N}$ mit

$$T^* = T_j.$$

Weil T^* durch Invertieren definiert wurde, gilt aber $T^*(j) \neq T_j(j)$.
Daraus folgt

$$T^* \neq T_j.$$

Widerspruch!

Beweisidee (Zusammenfassung):

Wenn die Menge aller Teilmengen von $\mathbb{N}$ abzählbar wäre, könnte man die Teilmengen in einer Liste darstellen, wie z.B.:

T_1 : **0**10101010...
T_2: 1**0**1010101...
T_3: 01**1**010100...
T_4: 100**1**00001...

Die Hauptdiagonale bilden die Bits $T_1(1), T_2(2), \ldots$, z.B. 0011...

Durch Invertieren der Hauptdiagonalen enthält man eine neue Teilmenge T^*, z.b. 1100..., die nicht in der Liste steht.

Also gibt es keine solche Liste.

Also ist die Menge aller Teilmengen von $\mathbb{N}$ nicht abzählbar.

1.5: Arithmethische Algorithmen

Es gibt Algorithmen, die lernt man schon in der Grundschule.

Z.B. die schriftlichen oder halbschriftlichen Algorithmen zur Addition, Subtraktion, Multiplikation und Division, z.B. zur Berechnung von

- 14219 + 4711,
- 14219 - 4711,
- 9508 * 95
- 903260 / 25

Rechenalgorithmen in Computern funktionieren fast genauso!

Stellenwertsysteme

Sei $b \geq 2$ eine "Basis" *(z.B. $b = 10$)*.

Jedes $a \in \mathbb{N}_0$, kann man als *"b-adische Zahl"*, d.h. als Folge von "Ziffern" aus $\{0, \ldots, b-1\}$ darstellen.

Ist $a < b^k$ genügen k Ziffern $(*)$,
wir schreiben $a = (a_{k-1}, \ldots, a_0) \in \{0, \ldots, b-1\}^k$, und es gilt

$$a = \sum_{0 \leq i < k} a_i b^i.$$

Beispiel: Im Dezimalsystem ($b = 10$) ist

$$123 = 3 * 10^0 + 2 * 10^1 + 1 * 10^2 = 3 * 1 + 2 * 10 + 1 * 100.$$

$(*)$ Nur wenn $a = 0$ ist, brauchen wir eine Ziffer statt null Ziffern.

Wie rechnen Computer?

Gottfried Wilhelm Leibnitz (1703)

(Public Domain:

`https://commons.wikimedia.org/wiki/File:`

`Gottfried_Wilhelm_Leibniz,_Bernhard_Christoph_`

`Francke.jpg`)

Computer rechnen *binär*,
also zur Basis $b = 2$.

Meist werdem mehrere binäre Ziffern (“Bits”) zu einem “Wort” zummengefasst.

Add(x, y)

Die Addition von Binärzahlen

Eingabe: $x, y \in \mathbb{N}_0$, $x = (x_{n-1}, \ldots, x_0) \in \{0,1\}^n$, $y = (\ldots) \in \{0,1\}^n$
Ausgabe: $s = x + y$

carry $:= 0$
for i **from** 0 **to** $n-1$ **do**
  $s_i := (x_i + y_i + \text{carry}) \bmod 2$
  carry $:= (x_i + y_i + \text{carry}) \operatorname{div} 2$
end for
$S_n :=$ carry

Die Laufzeit von Add

- Ziel: Grob abschätzen, wie schnell man n-bit Zahlen addieren kann. Keine “Benchmarks”.
- Unser Algorithmus besteht im Wesentlichen aus einer Schleife, die n-mal durchlaufen wird.
- In der Schleife werden nur einfache Operationen durchgeführt, deren Laufzeiten nicht von n abhängen.
  [a_1 := Zeit pro Schleifendurchlauf]
- Etwas Zusatzaufwand vor und nach der Schleife [a_0].
- *Laufzeit insgesamt $a_0 + na_1$ (linear in n).*

Grobe Einteilung von Laufzeiten

$T(n)$	Begriff	$T(2n) \leq$
a_0	konstant	$T(n)$
$a_0 + na_1$	linear	$2 * T(n)$
$a_0 + na_1 + n^2 a_2$	quadratisch	$4 * T(n)$
$a_0 + na_1 + n^2 a_2 + n^3 a_3$	kubisch	$8 * T(n)$
$\vdots$	$\vdots$	$\vdots$
$a_0 + na_1 + \ldots + n^k a_k$	polynomiell	$2^k * T(n)$
$\vdots$	$\vdots$	$\vdots$
2^n	exponentiell	$T(n)^2$

Bemerkung: In anderen Lehrveranstaltungen wird dies formalisiert und verfeinert, z.B. steht $O(n)$ bzw. $O(n^2)$ für höchstens lineare bzw. quadratische Laufzeit.

Mult(x, y)

Der Double_And_Add-Algorithmus zur Multiplikation von Binärzahlen

Eingabe: $x, y \in \mathbb{N}_0$, $x = (x_{n-1}, \ldots, x_0) \in \{0,1\}^n$
Ausgabe: $p = x * y$

$p := 0$
for i **from** 0 **to** $n-1$ **do**
  if $x_i \neq 0$ **then**
    $p := p + y$ $\quad (*$ *add conditionally* $*)$
  end if
  $y := y + y$ $\quad (*$ *double always* $*)$
end for

1. Für alle $x, y \in \mathbb{N}_0$ gilt: **Mult**(x, y)=$x * y$. *(Warum?)*
2. Die Laufzeit ist höchstens quadratisch in n. *(Warum?)*
   (. . . und ist die Laufzeit nicht sogar linear?)

1.6: Lösen eines Problems mit Python

Problem: Wir würfeln 60-mal mit einem “fairen” Würfel. Wie oft würfeln wir eine Sechs?

Ergebnis (Erwartung): Genau 10-mal.

Abweichungen: Wir wären kaum überrascht, wenn wir tatsächlich nur 9-mal oder sogar 11-mal eine Sechs würfeln würden.

Frage: Müssen wir an der Fairness des Würfels zweifeln, wenn wir, zum Beispiel, nur 7-mal oder sogar 13-mal eine Sechs würfeln?

Herangehenswweise: Wir simulieren das Würfeln im Computer.
LSbznV: Python-Handout und `dice.py`.

Simulation eines Würfels mit Python

Definition eines neuen Moduls “dice” in einer Datei “dice.py”:

Listing 1: Zeilen 1–8 aus dice.py

```
1  import random
2
3  def throw():
4      """
5      Simulates a dice throw.
6      Returns a random value in {1,2,3,4,5,6}.
7      """
8      return random.randrange(1,7) # uniformly from {1,2,3,4,5,6}
```

```
>>> import dice
>>> dice.throw()
1
>>> dice.throw()
2
>>> dice.throw()
6
```

```
>>> dice.throw()
2
>>> dice.throw()
5
>>> dice.throw()
5
```

Würfele *r*-mal und zähle die Sechsen

Ergänzen des Moduls “dice”:

Listing 2: Zeilen 10–16 aus dice.py

```
10  def count_six(r):
11      """Throws dice r times, counts how often a six occurs."""
12      result = 0
13      for throws in range(r):
14          if throw() == 6:
15              result = result + 1
16      return result
```

```
>>> import dice
>>> dice.count_six(60)
10
>>> dice.count_six(60)
13
>>> dice.count_six(60)
11
```

```
>>> dice.count_six(60)
9
>>> dice.count_six(60)
10
>>> dice.count_six(60)
10
>>> dice.count_six(60)
17
```

Die Implementation des eigentlichen Experiments

Listing 3: Zeilen 18–29 aus dice.py

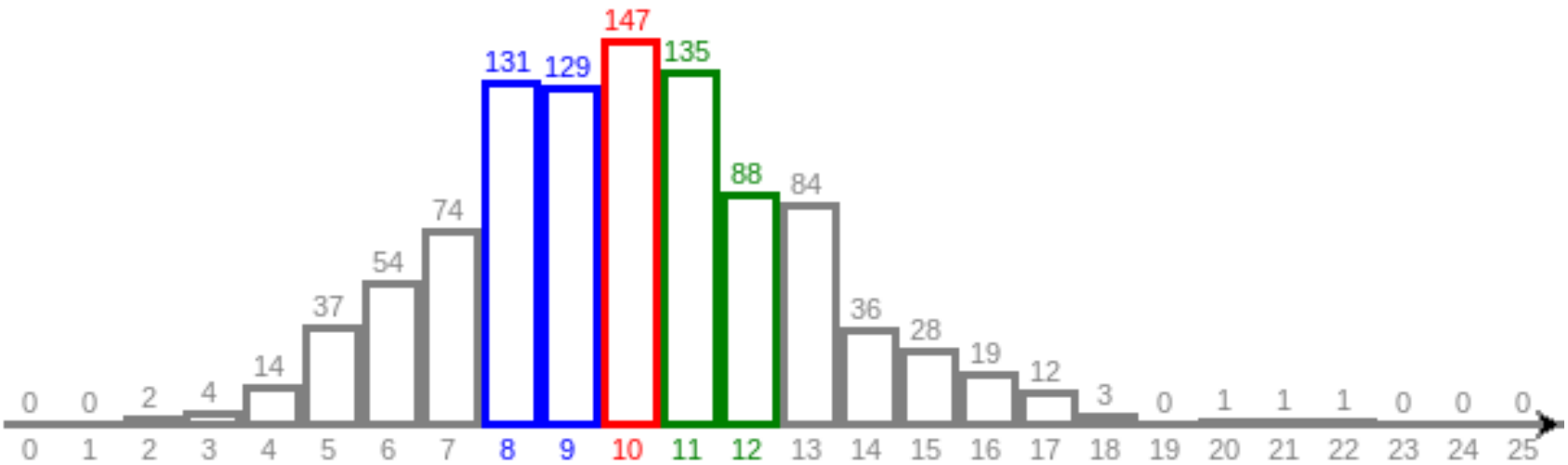
```
18  def experiment(n):
19      """
20      Counts how often count_six(n) gives a specific value;
21      this is repeated 1000 times;
22      experiment(n) returns a list 'results' of length n+1
23      'results[x]' is the number of times count_six(n) returned x
24      """
25      results = [0]*(n+1)
26      for repetitions in range(1000):
27          x = count_six(n)
28          results[x] = results[x] + 1
29      return results
```

Die Durchführung des Experiments

```
>>> dice.experiment(60)
[0, 0, 2, 4, 14, 37, 54, 74, 131, 129, 147, 135, 88, 84, 36,
28, 19, 12, 3, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0]
```

Also, zweimal hatten wir 2 Sechsen, viermal drei Sechsen, ...

Diese Liste von Werten ist wenig übersichtlich. Mehr sieht man, wenn man die Ergebnisse als *Histogramm* darstellt:



Antwort auf die Frage: Nein!

Müssen wir an der Fairness des Würfels zweifeln, wenn . . . ?

Die relative Häufigkeit, bei 60 Würfen nur 7-mal eine Sechs zu würfeln, oder sogar 13-mal, ist jeweils über 7 %. Mit Hilfe der Mathematik hätte man die Frage auch ohne vieltausendfaches Computer-simuliertes “Würfeln” herausfinden können.

Mehr dazu demnächst, im Kapitel “Diskrete Wahrscheinlichkeit”.