Technische Universität München

Mathematik-Vorkurs

2022w

Dr. Michael Luttenberger – Jeremias Bohn – Julian Geheeb

Welche Mengen sind Teilmengen von allen Elementen von M?

Blatt 2 - Mengenlehre

Aufgaben

$\bigcup M = \{1,2,3,4,5\}$

$|\bigcup M| = 2^5 = 32$

2.1 Fuuu-sion ...

Sei $M = \{\{1,2,3,5\},\{3,2,4,1\},\{4,3,1\}\}$.

(a) Bestimmen Sie $\bigcap M$ und $\bigcup M$.

(b) Bestimmen Sie $N = \{x \in 2^{\bigcup M} \mid \text{Für jedes } m \in M \text{ gilt } x \subseteq m\}$.

(c) Bestimmen Sie $\bigcup N$. $\{1,3\}$

$N = \{\{\}, \{3\}, \{1,3\}, \{3\}\}$

$\bigcup N = \{1,3\}$

3

$P(\{3\}) = \{\{\}\{3\}\}$ $\quad 2^0 = 1$

$P(\{1,2\}) = \{\{\}, \{1\}, \{2\}, \{1,2\}\}$ $\quad 1 \; 2 \; 3 \; 4$

$2^2 = 4 \to |P(\{1,2\})|$

2.2 ... ha?

4

Überlegen Sie sich, warum man den Schnitt nicht explizit als „Spielregel“ für die Konstruktion von Mengen benötigt.

2.3 „Beweis“

Wir fixieren eine Menge Ω als Grundmenge (Universum).

5

Seien $M, N \subseteq \Omega$ beliebige Teilmengen von Ω. Argumentieren Sie:

$$M \setminus N = M \cap \overline{N}$$

Zur Erinnerung: Nach Definition gilt

- $x \in M \setminus N$ <u>GDW</u> $(x \in M$ <u>UND</u> $x \notin N)$
- $x \in M \cap \overline{N}$ <u>GDW</u> $(x \in M$ <u>UND</u> $x \in \Omega \setminus N)$

$(A \cap B) \cup (C \cap \overline{D})$

$(A \cup B) \cap (C \cup \overline{D})$

2.4 Umformung von Mengentermen

Wir fixieren eine Menge Ω als Grundmenge (Universum).

Es seien A, B, C Teilmengen der Menge Ω. KDNF

6

Verwenden Sie die Rechenregeln aus den Folien, um die folgenden Mengendefinitionen schrittweise zu Ausdrücken der Form $M_1 \cup M_2 \cup \ldots \cup M_k$ umzuformen, wobei M_i ein beliebiger Schnitt der Mengen A, B, C und deren deren Komplementen bzgl. Ω ist.

Beispiel: Die Menge $A \setminus (B \cup C) \cup (B \cap C) \setminus A$ sollte als $(A \cap \overline{B} \cap \overline{C}) \cup (\overline{A} \cap B \cap C)$ geschrieben werden, da:

$$\begin{aligned} & A \setminus (B \cup C) \cup (B \cap C) \setminus A && \text{mit } X \setminus Y = X \cap \overline{Y} \\ = \; & (A \cap \overline{B \cup C}) \cup (B \cap C \cap \overline{A}) && \text{mit } \overline{X \cup Y} = \overline{X} \cap \overline{Y} \\ = \; & (A \cap \overline{B} \cap \overline{C}) \cup (B \cap C \cap \overline{A}) && \end{aligned}$$

Bringen Sie entsprechend folgende Ausdrücke schrittweise in die geforderte Form. Notieren Sie die Umformungen entsprechend des Beispiels.

(a) $((A \cup (B \cup C)) \cap (C \setminus A))$. $\quad (\overline{A} \cap C)$

(b) $\overline{(A \cap \overline{B \cap C})} \setminus (A \setminus B)$. $\quad \overline{A} \cup (B \cap C)$

2.1a) $\cap M = \{1,3\}$

$\cup M = \{1,2,3,4,5\}$

b) $N = \{\{\},\{1\},\{3\},\{1,3\}\}$

$2^{\cup M}$ ... Potenzmenge von $\cup M$

c) $\cup N = \{1,3\}$

2.2) siehe 2.1c $\quad \cup N = \cap M$

2.3) $M \setminus N = M \cap \overline{N}$

- $x \in M \setminus N$ GDW $(x \in M$ UND $x \notin N)$
- $x \in M \cap \overline{N}$ GDW $(x \in M$ UND $x \in \Omega \setminus N)$

O.B.d.A. $x \in M$

$x \notin N$ GDW $x \in \Omega \setminus N$

- $\underset{\text{wahr}}{x \notin N}$ IMP $\underset{\text{falsch}}{x \in \Omega \setminus N}$

alle Elemente liegen in $\Omega \rightarrow \underset{x \notin N}{x \in \Omega} \rightarrow x \in \Omega \setminus N$

$x \in \Omega \setminus N$ IMP $x \notin N$

trivial

- **Standardäquivalenzen I**:

  Für beliebige aussagenlogische Formeln F, G, H gilt:

$$
\begin{array}{lcll}
(F \wedge F) & \equiv & F & \text{(Idempotenz)} \\
(F \vee F) & \equiv & F & \\[6pt]
(F \wedge G) & \equiv & (G \wedge F) & \text{(Kommutativität)} \\
(F \vee G) & \equiv & (G \vee F) & \\[6pt]
((F \wedge G) \wedge H) & \equiv & (F \wedge (G \wedge H)) & \text{(Assoziativität)} \\
((F \vee G) \vee H) & \equiv & (F \vee (G \vee H)) & \\[6pt]
(F \wedge (F \vee G)) & \equiv & F & \text{(Absorption)} \\
(F \vee (F \wedge G)) & \equiv & F & \\[6pt]
(F \wedge (G \vee H)) & \equiv & ((F \wedge G) \vee (F \wedge H)) & \text{(Distributivität)} \\
(F \vee (G \wedge H)) & \equiv & ((F \vee G) \wedge (F \vee H)) &
\end{array}
$$

- **Standardäquivalenzen II**:

  Für beliebige aussagenlogische Formeln F, G, H gilt:

$\overline{A \cup B} = \bar{A} \cap \bar{B}$

$$
\begin{array}{lcll}
\neg\neg F & \equiv & F & \text{(Doppelnegation)} \\[6pt]
\neg(F \wedge G) & \equiv & (\neg F \vee \neg G) & \text{(deMorgan)} \\
\neg(F \vee G) & \equiv & (\neg F \wedge \neg G) & \\[6pt]
(F \wedge \neg F) & \equiv & \mathsf{false} & \text{(Triviale Kontradiktion)} \\
(F \vee \neg F) & \equiv & \mathsf{true} & \text{(Triviale Tautologie)} \\[6pt]
(F \wedge \mathsf{false}) & \equiv & \mathsf{false} & \text{(Dominanz)} \\
(F \vee \mathsf{true}) & \equiv & \mathsf{true} & \\[6pt]
(F \wedge \mathsf{true}) & \equiv & F & \text{(Identität)} \\
(F \vee \mathsf{false}) & \equiv & F &
\end{array}
$$

$\underline{\{1,2\}}$ $\quad$ $2^{\{1,2\}} = \{\emptyset, \{1\}, \{2\}, \{1,2\}\}$

$2^2 = 4$

$\{1,2\} = \{2,1\}$

$(1,2) \neq (2,1)$

$\{1,2,3,4,5\}$ $\quad$ $2^5 = 32$

2^M $\quad$ $|M| = 3$

$2^3 = 8$

$\{\{\}, A, B, C, \{A,B\}, \{A,C\}, \{B,C\},$
$M\}$

$$(A \cup (B \cup C)) \cap (C \cap \bar{A}) \quad | \text{ Distr.}$$

$$3 \cdot (1 + 2)$$

$$(A \cap (C \cap \bar{A})) \cup ((B \cup C) \cap (C \cap \bar{A}))$$

$$3 \cdot 1 + 3 \cdot 2$$

1

2.5 Operationen über Mengen

Gegeben sind die Mengen $A = \{1,3,5,7,9\}$, $B = \{2,4,6,8,10\}$ und $D = \{5,6,7,8,9,10\}$.

Geben Sie explizit für jeden der folgenden Mengenausdrücke die beschriebene Menge an:

a)	$A \cup B$,	b)	$A \cap B$,	c)	$A \setminus B$,
d)	$A \setminus D$,	e)	$B \setminus D$,	f)	$D \setminus A$,
g)	$D \setminus B$,	h)	$D \setminus (A \cup B)$,	i)	$D \setminus (A \cap B)$.

2

2.6 Mächtigkeit von Mengen

$\forall$ Menge M: $\emptyset \subseteq M$

Bestimme die Anzahl der Elemente der folgenden Mengen:

$$\text{a)}\ \{1,4,6\}, \quad \text{b)}\ \emptyset, \quad \text{c)}\ \{\emptyset\}, \quad \text{d)}\ \{\emptyset, \{1,2,3\}\}.$$

Kardinalität 3 0 1 2

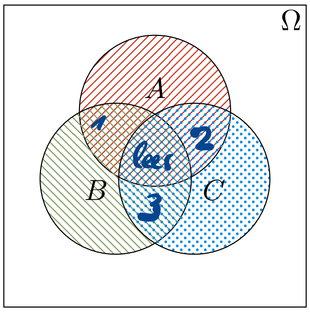
7

2.7 Venn-Diagramme

Beziehungen zwischen Mengen stellt man z.B. mittels Venn-Diagrammen (oder KV - Diagrammen, siehe DS) dar.

Hierzu zeichnet man die Mengen in **allgemeiner** Lage, d.h. so dass keine Schnittmenge ausgeschlossen wird, z.B. für drei Mengen $A, B, C \subseteq \Omega$:

1,2,3



$A = \{1,2\}$
$B = \{1,3\}$
$C = \{2,3\}$

Bestimmen Sie mittels des Venn-Diagramms konkrete Mengen A, B, C mit möglichst wenigen Elementen, so dass alle folgenden Anforderungen erfüllt sind

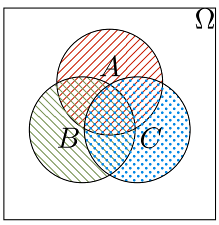
(1)	$A \cap B \cap C = \emptyset$	(2)	$A \cap B \neq \emptyset$
(3)	$B \cap C \neq \emptyset$	(4)	$A \cap C \neq \emptyset$

8

2.8 Venn-Diagramme zur Herleitung der Siebformel

Ist A eine Menge, so schreibt man $|A|$ für die Kardinalität der Menge. Für endliche Mengen, also Mengen, die man explizit durch Hinschreiben aller ihrer Elemente angeben kann, ist $|A|$ gerade die Anzahl der unterschiedlichen Elemente von A.

Überlegen Sie sich anhand des Venn-Diagramms



2.5)

Operationen über Mengen

Gegeben sind die Mengen $A = \{1,3,5,7,9\}$, $B = \{2,4,6,8,10\}$ und $D = \{5,6,7,8,9,10\}$.

Geben Sie explizit für jeden der folgenden Mengenausdrücke die beschriebene Menge an:

a) $A \cup B$,	b) $A \cap B$,	c) $A \setminus B$,
d) $A \setminus D$,	e) $B \setminus D$,	f) $D \setminus A$,
g) $D \setminus B$,	h) $D \setminus (A \cup B)$,	i) $D \setminus (A \cap B)$.

$\{\} = \emptyset$

a) $A \cup B = \{1 \dots 10\}$

c) $A \setminus B = A$

b) $A \cap B = \emptyset$

d) $A \setminus D = \{1, 3\}$

g) $D \setminus B = \{5, 7, 9\}$

h) $D \setminus (A \cup B) = \{\}$

i) $D \setminus (A \cap B) = D$

alternierende Vorzeichen

1 +
2 −
3 +
4 −

$|A \cup B \cup C \cup D| = |A| + |B| + |C| + |D| - |A \cap B| - |A \cap C| - |A \cap D| - |B \cap C| - |B \cap D| - |C \cap D| + |A \cap B \cap C| + |A \cap B \cap D| + |A \cap C \cap D| + |B \cap C \cap D| - |A \cap B \cap C \cap D|$

dass

$$|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |B \cap C| - |A \cap C| + |A \cap B \cap C|$$

gilt.

Zeichnen Sie entsprechend ein Venn-Diagramm für vier beliebige endliche Mengen und überlegen Sie sich dann, wie man $|A \cup B \cup C \cup D|$ anhand der Schnittmengen berechnen kann.

Bestimmen Sie dann die Anzahl der natürlichen Zahlen zwischen 1 und 100, die zumindest von einer der Zahlen 2, 3, 5 oder 7 ohne Rest geteilt werden.

9

2.9 Induktivität von Mengen

Sei $N = \{n \in \mathbb{N}_0 \mid 0 + 1 + 2 + \ldots + n = \frac{n(n+1)}{2}\}$.

Zeigen Sie, dass N induktiv ist.

Warum bedeutet dies, dass die Aussage

0

$n \quad n+1$

„FÜR ALLE natürlichen Zahlen $n \in \mathbb{N}_0$ gilt:

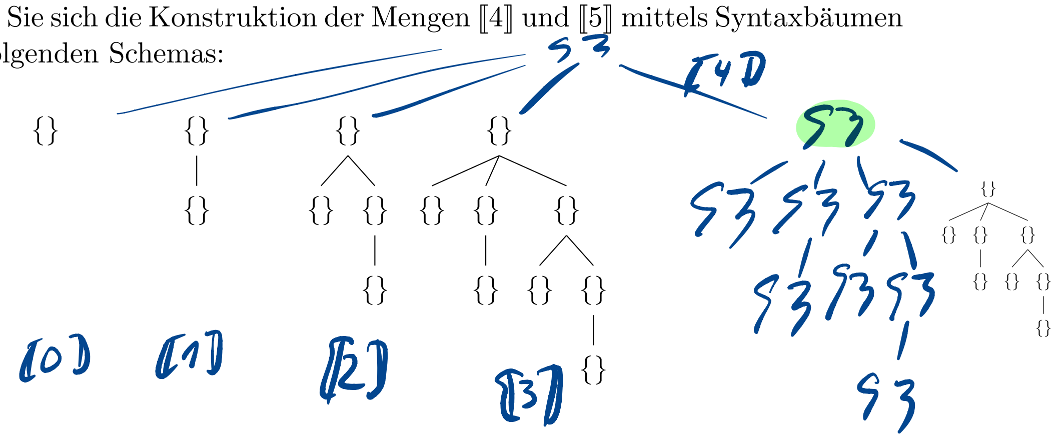
Die Summe der natürlichen Zahlen von 0 bis einschl. n ist gleich $\frac{n(n+1)}{2}$."

gültig ist?

10

2.10 Syntaxbäume zu Nat

Veranschaulichen Sie sich die Konstruktion der Mengen $[\![4]\!]$ und $[\![5]\!]$ mittels Syntaxbäumen entsprechend des folgenden Schemas:



Wie kann man sich dann Nat vorstellen? Warum kann man im Syntaxbaum zu Nat nur endlich viele Schritte nach unten machen?

Konstruktion der natürlichen Zahlen mit Mengen

11

2.11 Folgen und Induktivität

Wir definieren die Folge $(a_i)_{i \in \mathbb{N}_0}$ induktiv durch:

- $a_0 = 0$
- $a_i = i^2 + a_{i-1}$ für $i \in \mathbb{N}_0 \setminus \{0\}$

(a) Bestimmen Sie die „Folgenglieder" a_1, a_2, a_3, a_4 explizit.

(b) Sei $N = \{i \in \mathbb{N}_0 : a_i = \frac{i(i+1)(2i+1)}{6}\}$.
Prüfen Sie nach, dass N induktiv ist.

Induktivität von Mengen

Sei $N = \{n \in \mathbb{N}_0 \mid 0 + 1 + 2 + \ldots + n = \frac{n(n+1)}{2}\}$.

Zeigen Sie, dass N induktiv ist.

Warum bedeutet dies, dass die Aussage

„FÜR ALLE natürlichen Zahlen $n \in \mathbb{N}_0$ gilt:

Die Summe der natürlichen Zahlen von 0 bis einschl. n ist gleich $\frac{n(n+1)}{2}$."

gültig ist?

0

$n \curvearrowright n+1$

Gauß'sche Summenformel

zu zeigen: $\forall n \in \mathbb{N}_0: \quad \sum_{k=0}^{n} k = \frac{n(n+1)}{2}$

Induktionsanfang: $n = 0$

$$\sum_{k=0}^{0} k = \frac{0(0+1)}{2} = 0 \quad \checkmark$$

Induktionsschritt: Sei $n \in \mathbb{N}_0$ beliebig fixiert

Induktionsannahme: $\sum_{k=0}^{n} k = \frac{n(n+1)}{2}$

Induktionsbehauptung: $\sum_{k=0}^{n+1} k = \frac{(n+1)(n+2)}{2}$

Beweis der IB:
$$\sum_{k=0}^{n+1} k = (n+1) + \sum_{k=0}^{n} k$$
$$= (n+1) + \frac{n(n+1)}{2}$$
$$= \frac{2(n+1)}{2} + \frac{n(n+1)}{2}$$
$$= \frac{2(n+1) + n(n+1)}{2}$$
$$= \frac{(n+1) \cdot (2+n)}{2} \quad \square$$

- $a_0 = 0$
- $a_i = i^2 + a_{i-1}$ für $i \in \mathbb{N}_0 \setminus \{0\}$

(a) Bestimmen Sie die „Folgenglieder“ a_1, a_2, a_3, a_4 explizit.

(b) Sei $N = \{i \in \mathbb{N}_0 : a_i = \frac{i(i+1)(2i+1)}{6}\}$.
Prüfen Sie nach, dass N induktiv ist.

Quantoren

$\forall$ ... für alle

$\exists$ ... es existiert

a) $a_0 = 0$

$a_1 = 1^2 + a_{1-1} = 1^2 + a_0 = 1$

$a_2 = 2^2 + 1 = 5$

$a_3 = 3^2 + 5 = 14$

$a_4 = 4^2 + 14 = 30$

b) $\forall i \in \mathbb{N}_0 : \quad a_i = \frac{i(i+1)(2i+1)}{6}$

Induktionsbasis: $i = 0$

$$a_0 = \frac{0(0+1)(2 \cdot 0+1)}{6} = \frac{0}{6} = 0$$

Induktionsschritt: sei $i \in \mathbb{N}_0$ beliebig fixiert

Induktionsannahme: $a_i = \frac{i(i+1)(2i+1)}{6}$ $\quad 2(i+1)+1$

Induktionsbehauptung: $a_{i+1} = \frac{(i+1)(i+2)(2i+3)}{6}$

Beweis der IB:

$$a_{i+1} = (i+1)^2 + a_i$$
$$= (i+1)^2 + \frac{i(i+1)(2i+1)}{6}$$
$$= \frac{6(i^2+2i+1)}{6} + \frac{i(i+1)(2i+1)}{6}$$
$$= \frac{6i^2 + 12i + 6 + (i^2+i)(2i+1)}{6}$$

$$= \frac{6i^2 + 12i + 6 + (i^2 + i)(2i+1)}{6}$$

$$= \frac{6i^2 + 12i + 6 + 2i^3 + i^2 + 2i^2 + i}{6}$$

$$= \frac{2i^3 + 9i^2 + 13i + 6}{6}$$

$$= \frac{(2i^2 + 7i + 6)(i+1)}{6}$$

$$= \frac{(i+1)(i+2)(2i+3)}{6} \qquad \square$$

NAT – „Natürliche Zahlen mit Mengen simuliert"

$0 \mathrel{\hat{=}} \emptyset$

$1 \mathrel{\hat{=}} \{\emptyset\}$

$2 \mathrel{\hat{=}} \{\{\emptyset\}, \emptyset\}$

$3 \mathrel{\hat{=}} \{\{\{\emptyset\}, \emptyset\}, \{\emptyset\}, \emptyset\}$

Zusätzliche Aufgaben für Interessierte

2.1* Wohlfundierung

(a) Geben Sie Mengen M_1, M_2, M_3 an, für die „$M_1 \in M_2$ ODER $M_2 \in M_3$ ODER $M_3 \in M_1$" gilt

(b) Warum kann es keine Mengen M_1, M_2, M_3 geben, für die „$M_1 \in M_2$ UND $M_2 \in M_3$ UND $M_3 \in M_1$" gilt?

Beachten Sie: Falls M_1, M_2, M_3 Mengen sind, dann ist auch $\{M_1, M_2, M_3\}$ eine Menge. Wenden Sie dann die Fundierungsregel auf diese an.

2.2* Paare

Der Begriff des Paars (a, b) lässt sich auf den Mengenbegriff zurückführen.

Wir definieren die Menge $P(a, b)$ durch

$$P(a,b) = \{\{\varnothing, \{a\}\}, \{b\}\}$$

Argumentieren mittels der Spielregeln für Mengen, dass $P(a, b) = P(c, d)$ genau dann gilt, wenn $a = c$ und $b = d$ gilt.

(Nach der Paarmengenregel handelt es sich bei $P(a, b)$ um eine Menge, falls a, b Mengen sind.)

2.3* Python - Spielereien

Lösen Sie die „Rechenaufgaben" aus 2.5 mit Hilfe des Computers.

Python verwendet eine sehr Mathematik nahe Schreibweise für Mengen.

Z.B. lässt sich der Ausdruck $D \setminus (A \cup B)$ für $A = \{1, 3, 5, 7, 9\}$, $B = \{2, 4, 6, 8, 10\}$ und $D = \{5, 6, 7, 8, 9, 10\}$ durch folgendes Programm auswerten:

```python
A = {1,3,5,7,9}
B = {2,4,6,8,10}
D = {5,6,7,8,9,10}

D_ohne_Vereinigung_von_A_und_B = D - (A | B)
D_ohne_Schnitt_von_A_und_B = D - (A & B)
print(D_ohne_Vereinigung_von_A_und_B)
print(D_ohne_Schnitt_von_A_und_B)
```

Doku zu Mengen (Sets) in Python

Das „Programm" kann direkt auf `https://www.python.org/` ausgewertet werden (oben auf „> _" klicken, dann copy&paste); ansonsten bieten die meisten Linux-Distributionen einen Python-Installation direkt an (aus eigenem Interesse: Python3 verwenden, nicht Python2; schont die Nerven); unter Windows 10 kann man entweder das Linux-Subsystem (Ubuntu/Debian) verwenden; ansonsten ist z.B. Pycharm eine frei verfügbare IDE für Python.

2.4* Python und Nat - (1/3)

In den Folien werden Addition add und Multiplikation mul für Nat $= \{[\![n]\!] \mid n \in \mathbb{N}_0\}$ definiert.

Überprüfen Sie explizit anhand dieser Definitionen, dass

- $\mathsf{add}(\llbracket 2\rrbracket, \llbracket 1\rrbracket) = \llbracket 3\rrbracket = \mathsf{add}(\llbracket 2\rrbracket, \llbracket 1\rrbracket)$
- $\mathsf{mul}(\llbracket 2\rrbracket, \llbracket 1\rrbracket) = \llbracket 2\rrbracket = \mathsf{mul}(\llbracket 2\rrbracket, \llbracket 1\rrbracket)$
- $\mathsf{add}(\mathsf{add}(\llbracket 2\rrbracket, \llbracket 0\rrbracket), \llbracket 1\rrbracket) = \llbracket 2\rrbracket = \mathsf{add}(\llbracket 2\rrbracket, \mathsf{add}(\llbracket 0\rrbracket, \llbracket 1\rrbracket))$
- $\mathsf{mul}(\llbracket 2\rrbracket, \llbracket 0\rrbracket) = \llbracket 0\rrbracket = \mathsf{mul}(\llbracket 0\rrbracket, \llbracket 2\rrbracket)$
- $\mathsf{mul}(\llbracket 2\rrbracket, \llbracket 1\rrbracket) = \llbracket 2\rrbracket = \mathsf{mul}(\llbracket 1\rrbracket, \llbracket 2\rrbracket)$

Zum Spielen: Im Folgenden ist eine (sehr einfache, und daher u.U. sogar fehlerfreie) Implementierung der Definitionen von add und mul (unter Verwendung von $\mathsf{dec}(\llbracket n\rrbracket) = \bigcup\llbracket n\rrbracket$); durch Umdefinieren von pretty kann die Ausgabe von Nat auf $\mathbb{N}_0$ umgestellt werden.

Sinn: Üblicherweise können Computer zunächst nur natürliche Zahlen im Bereich $\{0, 1, \ldots, 2^{n-1} - 1\}$ mit z.B. $n = 32$ oder $n = 64$ darstellen (siehe Register); die mengentheoretische Darstellung Nat von $\mathbb{N}_0$ bietet in dem Sinn eine einfache, daher fehlerunanfällige (aber sehr langsame) Möglichkeit, mit natürlichen Zahlen prinzipiell unbeschränkter Größe zu rechnen.

Code: https://www7.in.tum.de/um/courses/mathevorkurs/2019/folien/nat_set.py

```
n0 = frozenset()   # n0 = ∅
Nat = set([n0])


def pretty_pure_set(s):
    return '{%s}' % ','.join([pretty(x) for x in s])


def pretty_set_of_predecessors(s):
    return '{%s}' % ','.join([str(x) for x in range(len(s))])


def pretty_number(s):
    return str(len(s))


def pretty(s):
    assert s in Nat
    return pretty_number(s)


def suc(n):
    assert n in Nat
    s = frozenset(n | frozenset([n]))   # n ∪ {n}
    Nat.add(s)
    return s


def dec(n):
    assert n in Nat
    res = set()
    for k in n:
        res = res.union(k)   # dec(n) = ⋃n
    return frozenset(res)


def less_than(m, n):
    assert m in Nat and n in Nat
    return m.issubset(n)   # ⟦m⟧ ⊊ ⟦n⟧


def less_than_or_equal_to(m,n):
    assert m in Nat and n in Nat
    return m == n or less_than(m, n)   # m = n oder ⟦m⟧ ⊊ ⟦n⟧


def add(m, n, verbose=' '):
    assert m in Nat and n in Nat

    if verbose:
        print(verbose + "add(%s, %s)" % (pretty(m), pretty(n)))

    # Falls n = ∅, dann add(m, ∅) = m
    if n == n0:
        return m
    # Ansonsten: add(m, suc(n')) = suc(add(m, n')) mit n = suc(n') und n' = dec(n)
    res = suc(add(m, dec(n), verbose=verbose + ' '))
    if verbose:
        print(verbose + "add(%s, %s) = %s" % (pretty(m), pretty(n), pretty(res)))
    return res


def mul(m, n, verbose=' '):
    assert m in Nat and n in Nat

    if verbose:
        print(verbose + "mull(%s, %s)" % (pretty(m), pretty(n)))

    # Falls n = ∅, dann mul(m, ∅) = ∅
```

```
    if n == n0:
        return n0
    # Ansonsten: mul(m, suc(n')) = add(mul(m, n'), m)  mit  n = suc(n')  und  n' = dec(n)
    res = add(mul(m, dec(n), verbose=verbose + ' '), m, verbose=verbose + ' ')
    if verbose:
        print(verbose + "mull(%s, %s) = %s" % (pretty(m), pretty(n), pretty(res)))
    return res

def examples():
    n1 = suc(n0)
    n2 = suc(n1)
    n3 = suc(n2)

    print(pretty(dec(n0)))
    print(pretty(dec(n1)))
    print(pretty(dec(n2)))
    print(pretty(dec(n3)))

    print(pretty(add(n3, n2)))
    print(pretty(add(n2, n3)))
    print(pretty(mul(n3, n2)))
    print(pretty(mul(n2, n3)))
    return


if __name__ == "__main__":
    examples()
    exit(0)
```

2.5* Python und Nat - (2/3)

Erweitern Sie das Programm entsprechend um eine Funktion exp, die die Exponentation m^n nur mittels mul und dec berechnet.

Verwenden Sie der Einfachheit wegen als Definition „$m^0 := 1$“.

Überprüfen Sie anhand von (kleinen!) Beispielen, dass i.A.

$m^{(n^k)} \neq (m^n)^k = m^{(n \cdot k)}$ bzw.

$\mathsf{pow}(m, \mathsf{pow}(n, k)) \neq \mathsf{pow}(\mathsf{pow}(m, n), k) = \mathsf{pow}(m, \mathsf{mul}(n, k))$.

2.6* Python und Nat - (3/3)

Ergänzen Sie eine Funktion, die allein unter Verwendung der Funktionen add und mul für eine Zahl n in Mengendarstellung $[\![n]\!] \in \mathsf{Nat}$ die entsprechende Dezimaldarstellung ausgibt.