Prüfungsrelevante Verfahren, Sätze und Rechenregeln

2 Diskrete Strukturen

2.1 Mengenlehre und Kombinatorik

- zwei Mengen A und B sind gleich wenn sie die selben Elemente haben, d.h. wenn $A \subseteq B \wedge B \subseteq A$
- Beachte z.B. dass $\{\{1,2\}\,7\} \nsubseteq \mathbb{N}$
- Schnitt und Vereinigung sind kommutativ, assoziativ, distributiv in beide Richtungen; für Beweise kann es nützlich sein sich die Definitionen dieser Operationen in Erinnerung zu rufen, $\overline{A \cup B} = \overline{A} \cap \overline{B}, \overline{A \cap B} = \overline{A} \cup \overline{B}$
- $A \times B = \{(a,b) \mid a \in A \wedge b \in B\}$ heißt **kartesisches Produkt** oder **Produktmenge**, $|A \times B| = |A| \cdot |B|$
- **Potenzmenge** $\mathcal{P}(A)$ ist die Menge aller (auch unechten) Teilmengen von A, $|\mathcal{P}(A)| = 2^{|A|}$, es gilt stets $\emptyset \in \mathcal{P}(A)$
- $\binom{n}{k} = \dfrac{n!}{k!(n-k)!}$
- **Handschlaglemma:** Anzahl der Teilnehmer einer Konferenz, die einer ungeraden Anzahl von Teilnehmern die Hand geben, ist immer gerade

2.2 Abbildungen

- für $f : A \to B, A' \subseteq A$ heißt $f[A'] = \{f(a) \mid a \in A'\}$ Bild von A' unter f
- **injektiv:** $f(a_1) = f(a_2) \Rightarrow a_1 = a_2$ ("für jedes $b \in B$ existiert höchstens ein $a \in A$ mit $f(a) = b$")
  Beweise über Gegenbeispiel oder $f(a_1) = f(a_2)$ setzen
- **surjektiv:** $f[A] = B$ ("für jedes $b \in B$ existiert mindestens ein $a \in A$ mit $f(a) = b$")
  Beweise über Gegenbeispiel oder Definitionsbereich der Umkehrfunktion untersuchen
- **bijektiv:** injektiv und surjektiv ("für jedes $b \in B$ existiert genau ein $a \in A$ mit $f(a) = B$")
- für f *injektiv (!!)* definieren wir $\boldsymbol{f^{-1}}$ **:** $f[A] \to A, b \mapsto f^{-1}(b) = a$ mit $f^{-1}(b) = a$ g.d.w. $f(a) = b$
- für $f : A \to B, g : B \to C$ ist **Komposition** $g \circ f : A \to C, x \mapsto g(f(x))$ ($\Rightarrow$ von rechts nach links ausführen!!)
- **Permutation** von X ist bijektive Abbildung von X nach X, für $X = \{1, \ldots, n\}$ ist S_n Menge aller Permutationen und $\pi \in S_n$ mit $\pi = \begin{pmatrix} 1 & \ldots & n \\ \pi(1) & \ldots & \pi(n) \end{pmatrix}$ und $|S_n| = n!$

2.3 Beweis mittels vollständiger Induktion (Beispiel)

Beweis. Die Aussage A_n sei $\sum_{k=0}^{n} q^k = \dfrac{1-q^{n+1}}{1-q}$ mit $n \in \mathbb{N}, q \in \mathbb{R}, q \neq 1$.

(IA): $n_0 = 0$: $\sum_{k=0}^{0} q^k = q^0 = 1 = \dfrac{1-q^{0+1}}{1-q}$ w.A.
$\Rightarrow$ Es gilt A_0

(IV): $\forall \tilde{n} : n_0 \leq \tilde{n} \leq n : \sum_{k=0}^{\tilde{n}} q^k = \dfrac{1-q^{\tilde{n}+1}}{1-q}$

(IS): $$\sum_{k=0}^{n+1} q^k = \sum_{k=0}^{n} q^k + q^{n+1} \overset{\text{(IV)}}{=} \frac{1-q^{n+1}}{1-q} + q^{n+1} = \frac{1-q^{n+1}+(1-q)q^{n+1}}{1-q}$$
$$= \frac{1-q^{n+1}+q^{n+1}-q^{n+2}}{1-q} = \frac{1-q^{(n+1)+1}}{1-q}$$

$\Rightarrow$ Damit ist die Behauptung für alle $n \in \mathbb{N}$ vollständig bewiesen □

- "Die Aussage A_n sei..." nur in VL und AuD Skript, *evtl.* wird sonst aber z.Z.: erwartet; IV muss auch nicht unbedingt notiert werden
- alles nochmal mit (n+1)+1 hinschreiben ist nicht nötig
- Varianten: $A_n \Rightarrow A_{n+1}$/ aus A_n folgt A_{n+1} für alle $n \in \mathbb{N}$
  w.A. /Folglich gilt A_n für alle $n \in \mathbb{N}, n \geq n_0$
- Beachte dass oft auch nur für $n \in \mathbb{N}, n \geq k$ bewiesen wird (kein $\tilde{n}$)!! und $n_0 = 0$ nicht immer gelten muss

2.4 Zahlentheorie

- $n \in \mathbb{N}, n \geq 1$ kann *eindeutig* geschrieben werden als $n = \prod_{i=1}^{k} p_i^{\alpha_i}$ mit p_i prim, $\alpha_i \in \mathbb{N}$
- für $a, b \in \mathbb{N}$ gilt $a \mid b \Leftrightarrow \exists k : ak = b$; $\quad a \mid b_1 \wedge a \mid b_2 \Rightarrow a \mid (b_1 + b_2) \wedge a \mid (b_1 - b_2)$
- $\mathrm{ggT}(0,0) = 0$; $\quad \mathrm{kgV}(m,0) = \mathrm{kgV}(0,n) = 0$; $\quad \mathrm{ggT}(m,n) \cdot \mathrm{kgV}(m,n) = m \cdot n$
- **Euklidischer Algorithmus** (EA, keine offizielle Abkürzung):
- **Erweiterter Euklidischer Algorithmus** (EEA, keine offizielle Abkürzung):

2.5 Kryptographie

- **diskreter Logarithmus:**
- **Diffie-Hellman-Merkle:**
- **RSA:**
  1. Bob wählt zufällig 2 Primzahlen p, q und berechnet $n := pq$
  2. Bob wählt zufällig $d \in \mathbb{Z}_{\phi(n)}^*$ und berechnet $i, h \in \mathbb{Z}$ mit $i \cdot d + h \cdot \phi(n) = \mathrm{ggT}(d, \phi(n)) = 1$ (EEA)
  3. n und i sind öffentliche Schlüssel und werden an Alice weitergegeben, d ist privater Schlüssel
  4. Alice schickt $c = m^i \bmod n$ an Bob mit Nachricht m $(0 \leq m < n)$
  5. Bob berechnet $m = c^d \bmod n$ (Potenzieren mit Al Kashi oder Euler-Fermat oder Zahl zerlegen)

2.6 Gruppentheorie

2.7 Graphentheorie

2.8 Aussagenlogik

2.9 Relationen