Kongruenzen

In der Zahlentheorie heißen zwei ganze Zahlen *a* und *b* ***kongruent modulo m***, wenn ***m*** die Differenz (*a* − *b*) teilt.

$a \equiv b \bmod m \quad :\Leftrightarrow m \mid a-b \quad m \in \mathbf{N}$

zwei Zahlen sind kongruent modulo einer Zahl *m*, wenn sie bei der Division durch *m* den selben Rest ergeben.

Jede Zahl ist zu sich selbst kongruent $\Rightarrow a \equiv a \bmod m \quad m \mid a-a$

Rechenregeln

$a \equiv b \bmod m \Rightarrow a+c \equiv b+c \bmod m \quad \forall a,b,c \in \mathbf{Z}$

$\Rightarrow a*c \equiv b*c \bmod m \quad \forall a,b,c \in \mathbf{Z}$

$a*c \equiv b*c \bmod m$ und ggT $(m*c) = 1 \Rightarrow a \equiv b \bmod m$

m, c … teilerfremd

Eine Kongruenz der Form $a*x \equiv b \bmod m$ ist nur lösbar, wenn ggT(a, m) ein Teiler von b ist.

Restklassen

die Menge aller zu a (modulo m) kongruenten ganzen Zahlen bezeichnet man als die Restklasse von a modulo m

$a = \{x \in \mathbf{Z} \mid x \equiv a \bmod m\}$ Restklasse von $a \equiv m$

Es gibt genau m Restklassen (0,1,...m-1) modulo m

$\mathbf{Z}_m = \{0, 1, 2, \ldots m-1\}$ Menge der Restklassen modulo m

Beispiel:

$3x \equiv 9 \bmod 12$

$a = b + q*m \quad q = 1, 2, 3, \ldots$
$3x = 9 + q*12$

Ergebnisse für alle q > 0

q	x	
0	3	
1	7	
2	11	
3	15	$\Rightarrow \bmod 12 = 3$
4	19	$\Rightarrow \bmod 12 = 7$
5	23	$\Rightarrow \bmod 12 = 11$
6	27	$\Rightarrow \bmod 12 = 3$

$\Rightarrow$ es gibt 3 Restklassen für $3x \equiv 9 \bmod 12$ 3, 7, 11

Beispiel:

Prüfziffer p ISBN

ISBN $x_1 - x_2x_3x_4 - x_5x_6x_7x_8x_9 - p$

$10x_1 + 9x_2 + ... 2x_9 + p \bmod 11$ $p \in \{0,1,2, \dots 9,x\}$

$p \equiv -10x_1 - 9x_2 - ... 2x_9 \bmod 11$ $\mid +11(x_1 + \dots + x_{9)}$

$\Rightarrow$ $\mathbf{p \equiv x_1 + 2x_2 + \dots + 9x_9 \bmod 11}$

Behauptung: Jeder Fehler in einer Ziffer wird vom ISBN-Code erkannt.

Beweis: angenommen, zwei ISBN-Nummern unterscheiden sich an einer Stelle x bzw. y

Prüfsummen:

$s + nx \equiv s + ny$ $1 \leq n \geq 10$ $x,y \in \{0,1,\dots 9,x\}$

$s + nx - s - ny \equiv 0 \bmod 11$

$n (x - y) \equiv 0 \bmod 11$

$\Rightarrow$ $x - y \equiv 0 \bmod 11$ n ist teilerfremd zu 11, ggT(n,11) = 1

$x \equiv y \bmod 11$

$\Rightarrow$ $x = y$ QED

Behauptung : Alle Vertauschungen zweier Ziffern werden vom ISBN-Code erkannt.

Beweis: angenommen, zwei ISBN-Nummern, x und y, werden vertauscht

$s + nx + my \equiv s + nx + my$ $1 \leq n,m \geq 10$ $x,y \in \{0,1,\dots 9,x\}$ $n \neq m$

$n (x - y) + m (y - x) \equiv 0 \bmod 11$

$(n - m)*(x - y) \equiv 0 \bmod 11$

$\Rightarrow$ $x - y \equiv 0 \bmod 11$ n –m ist teilerfremd zu 11

$\Rightarrow$ $x = y$ QED

Komplexe Zahlen

$a*x^2 + b*x + c = 0$ quadratische Gleichung

kann mit dieser Formel gelöst werden

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

$\Rightarrow$ $b^2 - 4ac$ ... Diskriminante D

für die Diskriminate D gilt

$D > 0 \Rightarrow$ zwei Lösungen
$D = 0 \Rightarrow$ eine Lösung
$D < 0 \Rightarrow$ keine reele Lösung

$C = \{a + b*i \mid a,b \in R\}$ Menge der komplexen Zahlen

$z = a + b*i$ Gauss`sche Darstellung der komplexen Zahlen

a ...Realteil

b ...Imaginärteil

i ...imaginäre Einheit

C ist eine Erweiterung von **R** weil,

$z = a + 0*i \leftrightarrow a \in R \Rightarrow R \subseteq C$

$i = 0 + 1*i \Rightarrow i^2 = (0 + 1*i)*(0 + 1*i) = -1 + 0*i = -1$

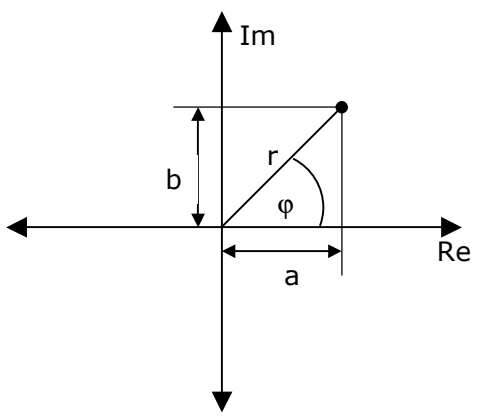
Rechenregeln

$(a+b*i) + (c + d*i) = (a + c) + (b + d)*i$

$(a+b*i) * (c + d*i) = (ac - bd) + (ad + bc)*i$

$\Rightarrow$ $\langle C, +, * \rangle$ bildet einen Körper

Darstellung in der Gauss'schen Zahlenebene



(a,b) ... Kartesische Koordinaten

[r,φ] ... Polarkoordinaten

Beweisführung

Zahlenzuordnung

Indirekter Beweis nach Euklid

Annahme dass $\sqrt{2}$ eine rationale Zahl ist, also es existiert eine rationale Zahl *x*, deren Quadrat 2 ist.

$x \in Q$ mit $x^2 = 2$

Als rationale Zahl lässt sich *x* als vollständig gekürzter Bruch ganzer Zahlen darstellen. Sei $p \in Z$ der Zähler und $q \in Z \setminus \{0\}$der Nenner dieses Bruchs.

Dann gilt $x = p/q$

Durch Quadrieren erhält man $x^2 = p^2/q^2$

unter Beachtung der Voraussetzung $x^2 = 2$ ergibt die Multiplikation mit q^2

$$2q^2 = p^2$$

Da die linke Seite der Gleichung gerade ist, ist auch die rechte Seite p^2 gerade. Daraus folgt, dass bereits die Zahl p gerade ist. p ist also das Doppelte einer ganzen Zahl. Wir bezeichnen diese ganze Zahl mit r und es gilt

$$p = 2r$$

Dies nun in die Gleichung einsetzen

$$2q^2 = p^2 = (2r)^2 = 4r^2$$

durch 2 kürzen

$$q^2 = 2r^2$$

$\Rightarrow$ q^2 und damit auch *q* ist eine gerade Zahl

$\Rightarrow$ Der Bruch kann durch 2 gekürzt werden. Widerspruch, da dieser Bruch bereits als vollständig gekürzt vorausgesetzt worden war.

Vollständige Induktion

ist eine Beweismethode, die üblicherweise eine Aussage für alle natürlichen Zahlen beweist.

Lässt sich die bestimmte Behauptung über natürliche Zahlen für eine gewisse Anfangszahl begründen (Induktionsanfang), und lässt sich außerdem zeigen, dass aus ihrer Geltung für eine beliebige Zahl n ihre Geltung für die nächste Zahl n + 1 folgt (Induktionsvoraussetzung), so gilt diese Behauptung für alle auf die Anfangszahl folgenden natürlichen Zahlen (Induktionsschritt).

Gilt für eine Aussage A(n), $n \in \mathbb{N}$, dass

(i)	A(0) wahr ist		Induktionsanfang
(ii)	$A(n) \Rightarrow A(n+1)$	$\forall\, n \in \mathbb{N}$	Induktionsvorraussetzung

dann ist A(n) wahr für alle natürlichen Zahlen $n \in \mathbb{N}$

A(n) ... Induktionsvorraussetzung

A(n+1) ...Induktionsbehauptung

(i) kleinstmögliches n in gegebenen Term einsetzen, ausrechnen (beide Seiten)

(ii) alle n durch (n + 1) ersetzen (rechte Seite), ausrechnen

(iii) Variablen auf der linken Seite durch (n+1) ersetzen und dann zur rechten Seite addieren, ausrechnen

Stimmen die Ergebnisse von (ii) und (iii) überein, stimmt die Aussage

Graphentheorie

Ein Graph besteht aus **Knoten (Ecken)** und **Kanten**, eine Kante verbindet genau zwei Knoten. Je zwei Knoten können also durch keine, eine oder mehr als eine Kante verbunden sein.

G(E,K)

E ...Knotenmenge

K...Kantenmenge

G = ⟨V, E⟩

V ...Knotenmenge (engl. Vertex)

E ...Kantenmenge (engl. Edge)

schlichte Graphen ... Graphen ohne Mehrfachkanten und Schleifen

endliche Graphen ... die Menge der Knoten und Kanten ist endlich

unendliche Graphen ... die Menge der Knoten und Kanten ist unendlich

ungerichtete Graphen

gerichtete Graphen ... die Kanten sind gerichtet (dargestellt durch Pfeile statt Linien)

Graph G ist ein Tripel ⟨V, E, f⟩, bestehend aus der Knotenmenge

V = V(G), der Kantenmenge E = E(G) und der

Indizenzabbildung $f : E \to V^2$ bzw. $f : E \to \{\{v,w\} \mid v,w \in V\}$

Es gilt f(e) = {v,w}, wenn die Kante e vom Anfangsknoten v zum

Endknoten w führt

Knotengrad Die Anzahl der Kanten, die von einem Knoten $e \in E$ ausgehen, bezeichnet man als Grad des Knotens deg(e).

Adjazenz Zwei *Knoten* heißen in einem ungerichteten Graph adjazent oder benachbart, wenn sie in diesem *durch eine Kante verbunden* sind.

Zwei *Kanten* heißen adjazent oder benachbart, wenn sie *sich an einem Knoten berühren*, dass heißt diesen gemeinsam besitzen.

Indizienz Ist ein Knoten ein Endknoten einer Kante sind diese indizent

Adjazenzmatrix A = A(G)

Ein Graph mit *n* Knoten kann durch eine $n \times n$ - Matrix repräsentiert werden. Dazu nummeriert man die Knoten von 1 bis *n* durch und trägt in die Matrix die Beziehungen der Knoten zueinander ein.

In ungerichteten Graphen ohne Mehrfachkanten wird in die *i*-te Zeile und *j*-te Spalte eine 1 eingetragen, wenn der *i*-te und *j*-te Knoten durch eine Kante verbunden sind. Andernfalls wird eine 0 eingetragen. Da ungerichtete Graphen keine Schleifen besitzen steht in der Hauptdiagonalen der Matrix überall die 0.

In gerichteten Graphen ohne Mehrfachkanten wird in die *i*-te Zeile und *j*-te Spalte eine 1 eingetragen, wenn der *i*-te Knoten Vorgänger des *j*-ten Knoten ist. Andernfalls wird eine 0 eingetragen.

Handschlaglemma

In einem ungerichteten Graphen gilt

$$\sum_{v \in V(G)} d(v) = 2|E(G)|$$

denn jede Kante liefert genau zweimal einen Beitrag zu der Summe der Grade

d(v) … Knotengrad, also die Anzahl der Kanten, die von einem Knoten $v \in E$ ausgehen

$\Rightarrow$ In endlichen Graphen ist die Anzahl der Knoten mit ungeradem Grad gerade

Zusammenhängende Graphen

$G = \langle V, E \rangle$ heißt *zusammenhängend* genau dann, wenn je zwei Knoten $e,f \in E$ durch einen Weg verbunden werden können

Stark zusammenhängend heißt ein gerichteter $G = \langle V, E \rangle$ dann, wenn es zu je zwei Knoten $v,w \in V(G)$ einen Weg von v nach w und einen Weg von w nach v gibt.

Schwach zusammenhängend heißt ein $G = \langle V, E \rangle$, wenn sein Schatten G_u (ergibt sich aus dem Weglassen der Kantenorientierung) zusammenhängend ist.

Sei $G = \langle V, E \rangle$ ein gerichteter oder ungerichteter Graph, ergibt sich aus diesem eine Folge

$x_0, (x_0, x_1), x_1, (x_1, x_2), x_2, \ldots, x_{k-1}, (x_{k-1}, x_k)$ mit $x_i \in V$ und $(x_i, x_j) \in E, k \in \mathbb{N}$

diese heißt:

Kantenfolge	der Länge k von x_0 nach x_k
Offene Kantenfolge	wenn $x_0 \neq x_k$
Geschlossene Kantenfolge	wenn $x_0 = x_k$
Kantenzug	wenn alle Kanten paarweise verschieden sind (wenn keine Kanten in W mehrfach auftreten)
Weg	wenn alle Knoten und alle Kanten verschieden sind

Kreis wenn alle Knoten und Kanten verschieden sind, mit Ausnahme von $x_0 = x_k$

Eine offene Kantenfolge von v nach w kann auf einen Weg von v nach w reduziert werden.

Ein geschlossener Kantenzug kann auf einen Kreis durch v reduziert werden.

Größtzusammenhängende Teilgraphen von $G = \langle V, E \rangle$ heißen *Komponenten*.

Jeder.$G = \langle V, E \rangle$ enthält mindestens $|V| - |E|$ viele Komponenten.

Für jeden zusammenhängenden Graphen $G = \langle V, E \rangle$ gilt $|E| \geq |V| - 1$

Isomorphie $\cong$

Zwei Graphen $G (V_1, E_1)$ und $G (V_2, E_2)$ heißen isomorph wenn eine bijektive Abbildung

$\varphi : V_1 \rightarrow V_2$ existiert, sodaß für alle $x, y \in V_1$ gilt $\{x, y\} \in E_1$, genau dann wenn

$\{\varphi(x), \varphi(Y)\} \in E_2$

Zwei Graphen sind genau dann isomorph, wenn der eine Graph aus dem anderen

Graphen durch Umbenennung der Knoten hervorgeht.

$G (V_1, E_1) \cong G (V_2, E_2)$

Eulersche Linie

ist ein Kantenzug, der jede Kante eines Graphen genau einmal enthält

In einem ungerichteten, zusammenhängenden Graphen G existiert genau dann eine geschlossene Euler'sche Linie wenn alle Knotengrade gleich sind.

Eine *Eulertour* in einem Graphen ist ein Weg, der jede Kante des Graphen genau einmal enthält und dessen Anfangs- und Endknoten identisch sind.

Hamiltonsche Linie

ist ein Weg, der jeden Knoten von G genau einmal enthält.

Ein Kreis, der jeden Knoten von G genau einmal enthält, heißt hamiltonscher Kreis.

Erfüllt ein Graph G (V, E) die Bedingung

$\deg(x) + \deg(y) \geq |V| \qquad \forall x, y \in V ; x \neq y ; \{x, y\} \notin E$

so enthält er einen Hamiltonkreis.

Erfüllt ein Graph G (V, E) die Bedingung

$\deg(v) \geq |V|/2$

für jeden Knoten, so ist er hamiltonsch.

Bäume und Wälder

Vorraussetzung: schlichte, ungerichtete Graphen

Wald ungerichteter Graph G ohne Kreise

Baum ungerichteter, zusammenhängender Graph G ohne Kreise

Ein Graph G ist genau dann ein Baum, wenn je zwei verschiedene Knoten durch genau einen Weg verbunden sind.

Für jeden endlichen Baum $T = \langle V, E \rangle$ gilt: $|V| = |E|+1$

Aufspannender Baum (Gerüst)

G $\langle V, E \rangle$ sei ein zusammenhängender Graph.

$T_G = \langle V', E' \rangle$ ist ein Teilgraph, der ein Baum ist und $V' = V$ erfüllt

Wurzelbaum

Zeichnet man in einem Baum einen Knoten als Wurzel aus, erhält man einen Wurzelbaum.

Binärer Baum

Ein vollständiger Binärbaum ist ein spezieller Wurzelbaum, bei dem jeder innere Knoten genau zwei Teilbäume besitzt.

$B = \langle V, E \rangle$ mit m inneren Knoten $\Rightarrow$ B hat genau m+1 Endknoten

$|V| = m + (m+1) = 2m + 1$

$|E| = 2m$

Algorithmus von Kruskal

ist ein Algorithmus der Graphentheorie mit dessen Hilfe man minimale Spannbäume in zusammenhängenden, ungerichteten, kantengewichteten Graphen berechnen kann.

minimaler Spannbaum Teilgraph eines ungerichteten Graphen, der ein Baum ist und alle seine Knoten enthält.

Die Grundidee ist, die Kanten in Reihenfolge aufsteigender Kantengewichte zu durchlaufen und jede Kante zur Lösung hinzuzufügen, die mit allen zuvor gewählten Kanten keinen Kreis bildet.
Es werden somit sukzessiv sogenannte Komponenten zum minimalen Spannbaum verbunden.

$G = \langle V, E, l \rangle$ $l : E \rightarrow \mathbf{R}$ bewerteter Graph

$l : E \rightarrow \mathbf{R}$ reelwertige Abbildung, ordnet jeder Kante einen Zahlenwert zu.

Der Algorithmus liefert einen minimalen Spannbaum $T_{min}(G) = \langle V, E' \rangle$ $E' \subseteq E$

Der Algorithmus von Kruskal arbeitet nicht deterministisch, d.h. er liefert unter Umständen beim wiederholten Ausführen unterschiedliche Ergebnisse. Diese Ergebnisse sind allesamt minimale Spannbäume von *G* und damit gleichwertige Lösungen.

Vorgehensweise:

1. Sortiere die Kanten von *G* aufsteigend nach ihrer Bewertung

2. Setze E' : = 0 und L : = E

3. Solange $|E| < |V|-k$ k ist die Anzahl der Komponenten

   wähle in L eine Kante e mit kleinster Bewertung. Entferne e aus L.

   Wenn $\langle V, E' \cup \{e\} \rangle$ keinen Kreis enthält, füge e zu E' hinzu.

4. Dann ist $T_{min}(G) = \langle V, E' \rangle$ ein minimaler Spannbaum von G.

   Notation $E' = \{(v_1,v_2),...\}$

also: Kanten nach Bewertung ordnen,
kleinste macht den Anfang,
dann nach der Reihe (Bewertung) wieder hinzufügen,
bildet eine Kante einen Kreis, weg damit,
voilá

Dieser Algorithmus ist endlich, und produziert einen minimal aufspannenden Baum von G

Algorithmus von Dijkstra und Danzig

dient der Berechnung eines kürzesten Weges in einem zusammenhängenden, bewerteten

Graphen $G = \langle V, E, l \rangle$ mit nichtnegativer Bewertungsfunktion l vom Knoten a zu b

1. Setze $l(a) = 0$; $l(v) = \infty$ $\forall\ v \neq a$, $U = \{a\}$, $u=a$

   U ...Menge der noch zu bearbeitenden Knoten

   u ...Vorgänger von v

   A ...Startknoten

2. $\forall$ Kanten $(u, v) \in E$ mit $v \in V\backslash U$ berechne man die neue Markierung

   $l(v) := \min \{l(v), l(u) + l(u+v)\}$

3. Wähle einen Knoten $v \in V\backslash U$ für den l(v) minimal ist und setze

   $U := U \cup \{v\}$, $u := v$

4. Wenn u = b dann ist l(b) die Länge des kürzesten Weges von a nach b

   wenn nicht Fortsetzung bei Schritt 2

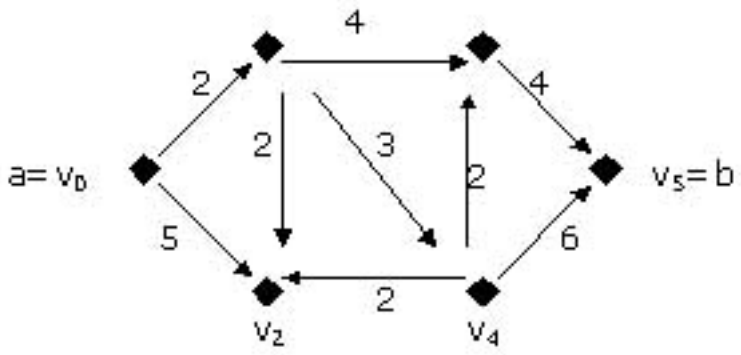
$\Rightarrow$ $V = V \cup (V \backslash U)$

U ...endgültig markierte Knoten, l(v) ist der kürzeste Weg von a nach b

V\U ...vorläufig markierte Knoten, l(v) kann sich fortlaufend ändern

Der Algorithmus endet nach höchstens $|V|$-1 Iterationen

Beispiel:



1. Iteration

(1) $l(v_0) = 0$, $l(v_1) = l(v_2) = ... = l(v_5) = \infty$, $U = \{v_0\}$, $u = v_0$

(2) Nachbarn von v_0 : v_1, v_2

$l(v_1) := \min\{l(v_1), l(v_0) + l(v_0, v_1)\} = \min\{\infty, 0 + 2\} = 2$

$l(v_2) := \min\{l(v_2), l(v_0) + l(v_0, v_2)\} = \min\{\infty, 0 + 5\} = 5$

(3) $l(v_1) = 2$ minimal, $U = \{v_0, v_1)$, $u = v_1$

(4) $v_1 \neq v_5$ $\Rightarrow$ weiter bei (2)

2. Iteration

(2) Nachbarn von v_1 : v_2, v_3, v_4

$l(v_2) := \min\{l(v_2), l(v_1) + l(v_1, v_2)\} = \min\{5, 2 + 2\} = 4$

$l(v_3) := \min\{l(v_3), l(v_1) + l(v_1, v_3)\} = \min\{\infty, 2 + 4\} = 6$

$l(v_4) := \min\{l(v_4), l(v_1) + l(v_1, v_4)\} = \min\{\infty, 2 + 3\} = 5$

(3) $l(v_2) = 4$ minimal, $U = \{v_0, v_1, v_2\}$, $u = v_2$

(4) $v_2 \neq v_5$ $\Rightarrow$ weiter bei (2)

Tabelle zur besseren Übersicht

Iteration	v_0	v_1	v_2	v_3	v_4	v_5	Auswahl	Vorgänger
0	0	∞	∞	∞	∞	∞	v_0	
1		2	5	∞	∞	∞	v_1	v_0
2			4	6	5	∞	v_2	v_1
3				6	5	∞	v_4	v_1
4				6		11	v_3	v_1
5						10	$v_5 = b$	v_3