Discrete Mathematics Summary

Code zu Algorithmen und umgesetzten Konzepten auf:

https://github.com/omeldar/mathematics-and-algorithms
Zu anderen meiner Zusammenfassungen:

https://github.com/omeldar/hslu

Propositionen

Propositionen sind anders gesagt auch Aussagen. Also zum Beispiel $1 + 1 = 2$ und $1 + 2 = 4$ sind Aussagen. Aussagen können richtig oder falsch sein.

Warheitswert einer Aussage, wenn sie wahr ist.

- w
- wahr
- true
- T
- 1

Wahrheitswert einer Aussage, wenn sie falsch ist.

- f
- falsch
- false
- F
- 0

Bezeichnungen für Aussagen sind meist: $p, q, r, s, \ldots$

Operanden

Negation

Ist p eine Proposition, dann ist die Proposition "Es ist nicht der Fall, das p gilt" die Negation von p. Man schreibt dies als $\neg p$

Wahrheitstabellen

Wahrheitstabellen stellen die Beziehungen zwischen den Wahrheitswerten von Propositionen dar.

p	¬ p
W	F
F	W

Der Negationsoperator ¬ wirkt auf die Proposition p. Es handelt sich um einen unären (einstelligen) Operator, weil er nur auf einen einzigen Operanden, nämlich die Proposition p wirkt.

Konjunktion

Die Propositionen $p \wedge q$ (gelesen: "p und q") heisst Konjunktion der Propositionen p und q, falls diese genau dann wahr ist, wenn p und q wahr sind; andernfalls ist sie falsch.

Disjunktion

Die Proposition $p \vee q$ gelesen: "p oder q") heisst Disjunktion der Propositionen p und q falls diese wahr ist, wenn mindestens eine der Propositionen p oder q wahr ist; andernfalls ist sie falsch.

Konjunktion und Disjunktion

p	q	p ∧ q	p ∨ q
w	w	w	w
w	f	f	w
f	w	f	w
f	f	f	f

- Konjunktion und Disjunktion werden auch **UND** und **ODER**-Verknüpfung genannt.
- Oder ist nicht exklusiv gemeint, also es kann auch beides `true` sein, damit die Aussage gesamthaft als gewertet werden kann.
- Es gilt folgende Warheitstabelle

XOR

p	q	p ⊕ q
w	w	f
w	f	w
f	w	w
f	f	f

Die Proposition XOR ($p \oplus q$) (gelesen: "p exor q") heisst XOR-Verknüpfung der Propositionen p und q, da diese genau dann wahr ist, wenn genau eine der Propositionen p oder q wahr ist, aber nicht beide gleichzeitig; ansonsten ist sie falsch

Implikationen

p	q	p → q
w	w	w
w	f	f
f	w	w
f	f	w

Die Implikation $p \rightarrow q$ (gelesen: "p impliziert q" oder "falls p, dann q") ist die jenige Proposition, die genau dann falsch ist, wenn p wahr und q falsch ist; andernfalls ist die Implikation wahr. P heisst auch Hypothese und q Konklusion.

p = "Ich werde als Politiker gewählt"

q = "Ich senke die steuern"

Die Aussage $p \rightarrow q$ ist nur dann falsch, wenn er gewählt wird, die Steuern aber nicht senkt. Sonst hatte er ja die Gelegenheit nicht dazu. Für q muss also p wahr sein.

Bikonditional (Bijunktion)

p	q	p ↔ q
w	w	w
w	f	f
f	w	f
f	f	w

Das Bikonditional $p \leftrightarrow q$ (gelesen: "p genau dann, wenn q") ist die jenige Proposition, die wahr ist, wenn p und q die selben Wahrheitswerte haben und sonst falsch.

Falls p = "Sie können den Flug nehmen" und q = "Sie kaufen ein Ticket"

q muss nicht wahr sein, wenn es p auch nicht ist. Wenn q wahr ist, muss dies p auch sein.

Priorität logischer Operatoren

In einem logischen Ausdruck wird der Operator mit der höchsten Priorität mit seinen Operanden zuerst zusammengefasst. Danach wir dder nächst höhere Operator mit seinen Operanden zusammengefasst. Gibt es mehrere Operanden mit der selben Priorität dann erfolgt die Auswertung von links nach rechts.

Prioritäten

Operator	Priorität
¬	1
∧	2
∨	2
→	3
↔	3

Tautologie und Widerspruch

Eine zusammengesetzte Aussage, die immer wahr (falsch) ist, heisst Tautologie (Kontradiktion oder Widerspruch).

p	¬p	p ∨ ¬ p	p ∧ ¬ p
w	f	w	f
f	w	w	f

Was stellt man bei den Propositionen $p \vee \neg p$ und $p \wedge \neg p$ fest? Um dies herauszufinden, kann man die Wahrheitstabelle zu diesen Propositionen angeschaut werden. Denn die erste Proposition ist immer wahr und die zweite Proposition ist immer falsch, egal welchen Wert die einzelnen Propositionen in der zusammengesetzten Proposition sind.

Wir können also sagen, dass $p \vee \neg p = T$, bedeutet, dass $p \vee \neg p$ eine Tautologie ist. Für $p \wedge \neg p$ können wir sagen, dass sich diese Proposition immer widerspricht, also ein Widerspruch ist:

$$p \wedge \neg p = F$$

Logische Äquivalenzgesetze

ℹ️ Dieses Modul wird von Informatikern besucht. Der Pseudo-Code soll als Hilfestellung für das Verständnis dienen.

Identität

$p \wedge T \equiv p$

Dieses Gesetz besagt, wenn die Proposition p und eine Tautologie in einer zusammengesetzten Proposition unter dem Operanden und stehen, wird der Wahrheitswert anhand der Proposition p bestimmt.

```
if (p && true)
```

⚠️ Es hängt nur an p, ob diese Proposition `true` oder `false` ist.

$p \vee F \equiv p$

Das gleiche hier. Wenn die Proposition p und eine Kontradiktion in einer zusammengesetzten Proposition unter dem Operanden oder stehen, wird der Wahrheitswert anhand der Proposition p bestimmt.

```
if (p || false)
```

⚠️ Es hängt nur an p, ob diese Proposition `true` oder `false` ist.

Dominanz

$p \vee T \equiv T$

Dieses Gesetz besagt, wenn die Proposition p und eine Tautologie in einer zusammengesetzten Proposition unter dem Operanden oder stehen, wird der Wahrheitswert immer `true` sein.

```
if (p || true)
```

⚠️ Diese Proposition ist immer `true`

$p \wedge F \equiv F$

Dieses Gesetz besagt, wenn die Proposition p und eine Kontradiktion in einer zusammengesetzten Proposition unter dem Operanden und stehen, wird der Wahrheitswert immer `false` sein.

```
if (p && false)
```

⚠️ Diese Proposition ist immer `false`

Idempotenz

$p \vee p \equiv p$

Dieses Gesetz sagt nichts mehr als, wenn eine Proposition mit sich selber mit dem oder Operanden verknüft wird, reicht es aus, wenn man die alleinige nicht zusammengesetzte Proposition betrachtet, um den Wahrheitswert zu ermitteln.

```
if (p || p)
```

⚠️ Diese Proposition ist gleichgestellt mit nur `if ( p )`

$p \wedge p \equiv p$

Dieses Gesetz sagt nichts mehr als, wenn eine Proposition mit sich selber mit dem und Operanden verknüft wird, reicht es aus, wenn man die alleinige nicht zusammengesetzte Proposition betrachtet, um den Wahrheitswert zu ermitteln.

```
if (p && p)
```

⚠️ Diese Proposition ist gleichgestellt mit nur `if ( p )`

Doppelnegation

$\neg(\neg p) \equiv p$

Den Wahrheitswert einer Proposition die doppelt negiert wird, kann durch die Proposition selbst ohne Negation ermittelt werden.

```
if (!(!p))
```

⚠ Diese Proposition ist gleichgestellt mit nur `if ( p )`

Tautologie / Kontradiktion

$p \vee \neg p \equiv T$

Dieses Gesetz zeigt, dass wenn die gleiche Proposition einmal negiert und einmal un-negiert mit dem oder Operanden zusammengesetzt wird, ist dies eine Tautologie, also immer wahr. Dies, da mindestens einer dieser Werte wahr sein wird.

```
if (p || !p)
```

⚠ Wird immer `true` sein

$p \wedge \neg p \equiv F$

Dieses Gesetz zeigt, dass wenn die gleiche Proposition einmal negiert und einmal un-negiert mit dem und Operanden zusammengesetzt wird, ist dies eine Kontradiktion, also immer falsch. Dies, da mindestens einer der Werte falsch sein wird.

```
if (p && !p)
```

⚠ Wird immer `false` sein

Kommutativität

$p \vee q \equiv q \vee p$

Dieses Gesetz zeigt, dass die Reihenfolge einer Proposition die mit dem oder Operanden zusammengesetzt wird keine Rolle spielt.

```
if (p || q)
if (q || p)
```

⚠ Es spielt keine Rolle, wie die Reihenfolge ist.

$p \wedge q \equiv q \wedge p$

Dieses Gesetz zeigt uns, dass die Reihenfolge einer Proposition die mit dem und Operanden zusammengesetzt wird keine Rolle spielt.

```
if (p && q)
if (q && p)
```

⚠ Es spielt keine Rolle, wie die Reihenfolge ist.

Absorption

ℹ Ab hier ist logisches überlegen gefragt, diese Gesetze sind recht schwierig kurz zu beschreiben, so dass es leicht verständlich ist. Am besten alle Szenarien mal durchspielen mit `true` und `false` oder mit `0` und `1`

$p \vee (p \wedge q) \equiv p$

```
if (p || (p && q))
```

⚠ Dies ist das gleiche wie `if ( p )`

$p \wedge (p \vee q) \equiv p$

```
if (p && (p || q))
```

⚠ Dies ist das gleiche wie `if ( p )`

Assoziativgesetz

$(p \vee q) \vee r \equiv p \vee (q \vee r) \equiv p \vee q \vee r$ $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r) \equiv p \wedge q \wedge r$

Das Assoziativgesetz zeigt, dass es egal ist, wo die Klammern gesetzt werden, solange der binomiale Operand zwischen den Propositionen der selbe bleibt.

```
if ((p || q) || r)
if (p || (q || r)
if (p || q || r)
```

⚠️ Diese ergeben alle den selben Wahrheitswert

```
if ((p && q) && r)
if (p && (q && r)
if (p && q && r)
```

⚠️ Diese ergeben alle den selben Wahrheitswert

Distributivgesetz

$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$ $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$

Dieses Gesetz funktioniert wie das ausmultiplizieren in der Algebra. Hierbei ist wichtig, die richtigen Operanden an die richtigen Stellen zu setzen.

```
if (p || (q && r))
if ((p || q) && (p || r))
```

⚠️ Diese ergeben beide den selben Wahrheitswert

```
if (p && (q || r))
if ((p && q) || (p && r))
```

⚠️ Diese ergeben beide den selben Wahrheitswert

De Morgan's Gesetz

$\neg(p \wedge q) \equiv \neg p \vee \neg q$ $\neg(p \vee q) \equiv \neg p \wedge \neg q$

Dieses Gesetz zeigt, dass auch der Negationsoperand auf eine ganze Klammer angewandt werden kann. Dies bewirkt dann, dass jede Proposition in dieser Klammer sowie auch die Operanden darin, negiert werden.

```
if (!(p && q))
if (!p || !q)
```

⚠️ Diese ergeben beide den selben Wahrheitswert

```
if (!(p || q))
if (!p && !q))
```

⚠️ Diese ergeben beide den selben Wahrheitswert

Logische Äquivalenzgesetze - Lookup

Gesetz	Duales Gesetz
$p \wedge T \equiv p$	$p \vee F \equiv p$
$p \vee T \equiv T$	$p \wedge F \equiv F$
$p \vee p \equiv p$	$p \wedge p \equiv p$
$\neg(\neg p) \equiv p$	
$p \vee \neg p \equiv T$	$p \wedge \neg p \equiv F$
$p \vee q \equiv q \vee p$	$p \wedge q \equiv q \wedge p$
$p \vee (p \wedge q) \equiv p$	$p \wedge (p \vee q) \equiv p$
$(p \vee q) \vee r \equiv p \vee (q \vee r) \equiv p \vee q \vee r$	$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r) \equiv p \wedge q \wedge r$
$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
$\neg(p \wedge q) \equiv \neg p \vee \neg q$	$\neg(p \vee q) \equiv \neg p \wedge \neg q$

Erweiterte Gesetze / Rechenregeln

$p \to q \equiv \neg p \vee q$	$p \leftrightarrow q \equiv (p \to q) \wedge (q \to p)$
$p \to q \equiv \neg q \to \neg p$	$p \leftrightarrow q \equiv \neg p \leftrightarrow \neg q$
$p \vee q \equiv \neg p \to q$	$p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$
$p \wedge q \equiv \neg(p \to \neg q)$	$\neg(p \leftrightarrow q) \equiv p \leftrightarrow \neg q$
$\neg(p \to q) \equiv p \wedge \neg q$	
	$(p \to q) \wedge (p \to r) \equiv p \to (q \wedge r)$
$p \oplus q \equiv (p \vee q) \wedge (\neg p \vee \neg q)$	$(p \to r) \wedge (q \to r) \equiv (p \vee q) \to r$
$\neg(p \oplus q) \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$	$(p \to q) \vee (p \to r) \equiv p \to (q \vee r)$
$\neg(p \oplus q) \equiv p \leftrightarrow q$	$(p \to r) \vee (q \to r) \equiv (p \wedge q) \to r$

Prädikat

Falls eine Folge von Wörtern bei geeigneter Wahl einer oder mehrer Variablen zu einer Aussage wird, dann spricht man von einem Prädikat.

Zum Beispiel: $P(x) =$ "$x > 3$"

Prädikate enthalten mindestens eine Variable (hier: x). Sie haben aber nur dann einen eindeutigen Wahrheitswert, wenn man für diese Variable einen bestimmten Wert einsetzt. Man kann also nach den Wahrheitswerten von $P(4)$ und $P(2)$ fragen.

> ⚠ **Definition**
> Ein Prädikat ist eine Folge von Wörtern die Variablen enthalten und für jede (erlaubte) Belegung dieser Variablen zu einer Aussage werden. Man nennt z.B. die Aussage P(2) auch den Wert der propositionalen Funktion (Prädikats) $P(x)$ für $x = 2$

Quantoren

FORALL

Der Allquantor: FOR∀LL

Der ∀ Quantor beschreibt alle Werte einer Menge. z.B $\forall x \in \mathbb{R}$ beschreibt alle reellen Werte, während $\forall x \in \mathbb{Z}$ nur die ganzen Zahlenwerte beschreibt.

EXISTS

Der Existenzquantor: ∃XISTS

Der ∃ Quantor beschreibt, dass ein gewisser Wert in einer Menge existiert. z.B. $\exists x \in \mathbb{Z} f(x)$ sagt uns, dass ein Wert x in der Menge der ganzen Zahlen existiert, der eine Lösung der Funktion f ist.

> ℹ️ Wird ein Quantor auf die Variable x angewendet, dann nennt man diese Variable gebunden; ansonsten frei.

Verschachtelte Quantoren

Die Reihenfolge der Quantoren ist wesentlich; ausser alle Quantoren sind vom gleichen Typ. Also alles Allquantoren oder Existenzquantoren.

Beispiel: $\forall x \exists y(xy = 1)$

Für jedes x, gibt es ein y, so dass: $x \cdot y = 1$. Und zwar immer $x \cdot \frac{1}{x} = 1$. Allerdings nur wenn die Universalmenge $\mathbb{R}$ ist. Denn es gibt keine ganz-zahligen Werte für jedes x, die diese Proposition erfüllt. Somit stimmt $\forall x \exists y(xy = 1)$ für die Universalmenge $\mathbb{R}$.

> ⚠️ $\exists y \forall x(xy = 1)$ ist nicht das Selbe. Denn dies wäre falsch!

Ein Beispiel mit Code

Wenn für $\forall y \forall y P(x, y)$ der Wahrheitswert geprüft werden soll, könnte man dies so mit Code beschreiben:

```
∀x∀yP(x,y) := w;
for x:
    for y:
        if ¬P(x,y):
            ∀x∀yP(x,y) := f;
            break;
```


CS:GO RALL
USE CODE
NEDSÚCHTIG

Mathematisches Beweisen

Ein **Theorem** ist eine **Aussage**, von der man zeigen kann, dass sie wahr ist. Um zu zeigen, dass ein Theorem wahr ist, verwendet man eine Sequenz von zusammenhängenden Aussagen, die den **Beweis** ergeben. Aussagen können **Axiome** oder **Postulate** enthalten. Durch logische Zusammenhänge werden Folgerungen gemacht, die zusammen den Beweis ergeben.

Ein **Lemma** ist eine einfache Aussage, die in Beweisen von komplizierteren Aussagen verwendet wird.

Ein **Korollar** ist eine einfache Folgerung einer Aussage.

Direkter Beweis

Bei einem direkten Beweis, beweist man die Aussage mithilfe von Aussagen ohne die Aussage dazu zu verändern.

Die Implikation $p \rightarrow q$ kann bewiesen werden, in dem man zeigt, dass wenn p wahr ist, q wahr sein muss.

Beispiel:

Das Quadrat einer geraden natürlichen Zahl n, ist gerade.

$$\underbrace{n \in \mathbb{N} \text{ gerade}}_{p} \rightarrow \underbrace{n^2 \text{ gerade}}_{q}$$

Beweis: Jede natürliche Gerade Zahl ist durch 2 teilbar. Das heisst, sie enthält den Faktor 2. Wenn n also gerade sein soll, muss eine gerade natürliche Zahl k existieren, so dass $n = 2k$ gilt. Daraus entnehmen wir, dass:

$$n^2 = (\text{gerade Zahl})^2 \text{ also } n^2 = (2k)^2$$
$$n^2 = 4k^2 = 2 \cdot (2k^2)$$

Somit haben wir bewiesen, dass n^2 das Doppelte einer natürlichen Zahl, also gerade ist.

Indirketer Beweis

Bei einem indirekten Beweis beweisen wir nicht die Aussage direkt, sondern ändern das, was wir beweisen möchten, etwas ab. Zum Beispiel beweist man, dass das Gegenteil davon (die Kontraposition). Statt also zu zeigen dass $p \rightarrow q$ gilt, zeigt mann, dass die logisch äquivalente Aussage $\neg q \rightarrow \neg p$ gilt.

Man benötigt dieses Vorgehen dann, wenn ein direkter Beweis schwierig ist.

$$\underbrace{n \in \mathbb{N} \text{ gerade}}_{p} \rightarrow \underbrace{n^2 \text{ gerade}}_{q} \overset{\text{Kontraposition}}{\equiv} \underbrace{n \text{ nicht gerade}}_{\neg q} \rightarrow \underbrace{n^2 \text{ nicht gerade}}_{\neg p}$$

Anstelle direkt zu beweisen, dass wenn n^2 gerade ist, dass n gerade ist, beweisen wir hier, dass wenn n nicht gerade ist, n^2 nicht gerade ist. Sei n eine ungerade natürliche Zahl, dann existiert eine natürliche Zahl k, so dass $n = 2k + 1$ gilt. Daraus folgt:

$$n^2 = (\text{gerade Zahl} + 1)^2 = \text{ungerade Zahl}^2$$
$$n^2 = 4k^2 + 4k + 1 = 2 \cdot (2k^2 + 2k) + 1$$

Somit haben wir bewiesen, dass n^2 ungerade ist.

Beweis durch Kontradiktion

Ein anderes Verfahren für einen indirekten Beweis, ist der Beweis durch Kontradiktion (Widerspruch). Um die Aussage p zu beweisen, nimmt man an, $\neg p$ ist wahr. Diese Aussage führt man durch logisches Schliessen auf den Widerspruch q $(q = F)$, das heisst $\neg p \rightarrow q$ (also $\neg p \rightarrow F$). Somit muss $\neg p$ falsch und damit p wahr sein.

Einfache Erklärt an einem Beispiel:

Aristoteles glaubte, dass schwere Objekte schneler fallen, als leichte Objekte. Dies soll man widerlegen!

Angenommen, schwere Objekte fallschen schneller als leichtere. Dann nehmen wir ein schweres und ein leichtes Objekt und kleben beide zusammen

- Einerseits müsste die Fallgeschwindigkeit dieses neuen Objektes zwischen den Geschwindigkeiten der Ausgangsobjekte liegen, denn das leichte (langsam fallende) Teilobjekt wird den Fall des schweren Teilobjektes abbremsen.
- Andererseits ist das neue Objekt sicher schwerer als jedes einzelne Teilobjekt und müsste somit auch schneller fallen als beide.

Beide logischen Überlegungen widersprechen sich, unsere Annahme muss also falsch sein.

Zahlenbeispiel: Wir zeigen, dass $\sqrt{2} \notin \mathbb{Q}$ mit Hilfe eines Beweises durch Widerspruch. Also $p = \sqrt{2} \notin \mathbb{Q}$

Annahme: $\neg p$ ist wahr, das heisst: $\sqrt{2} = \frac{\mathbb{Z}}{\mathbb{N}} \in \mathbb{Q}$

Wir können $\frac{\mathbb{Z}}{\mathbb{N}}$ vollständig kürzen. Es entsteht $\sqrt{2} = \frac{\mathbb{Z}}{n}$, wobei Z und n keine gemeinsamen Teiler > 1 haben.

Wir rechnen also:

$$\sqrt{2} = \frac{\mathbb{Z}}{n} \rightarrow 2 = \frac{\mathbb{Z}^2}{n^2} \rightarrow 2n^2 = \mathbb{Z}^2 \rightarrow \mathbb{Z}^2 \text{ ist gerade} \rightarrow \mathbb{Z} \text{ ist gerade} \rightarrow 2 \text{ teilt } \mathbb{Z}$$
$$2n^2 = \mathbb{Z}^2 \rightarrow 2n^2 = (2k)^2 \rightarrow 2n^2 = 4k^2 \rightarrow n^2 = 2k^2 \rightarrow n^2 \text{ ist gerade } \rightarrow n \text{ ist gerade} \rightarrow 2 \text{ teilt } n$$

Somit haben wir gezeigt, dass $\neg p \rightarrow F$ und damit folgt p ist wahr.

> 💡 Bei beiden indirekten Beweismethoden versucht man, die negierte Aussage zu beweisen, aber der Schluss und das Ziel sind unterschiedlich:
>
> 1. Bei der Kontraposition versucht man, die Kontrapositive Aussage (die Umkehrung und Negation der Konsequenz) direkt zu beweisen. Wenn diese Kontrapositive Aussage wahr ist, schliesst man darauf, dass die ursprüngliche Aussage ebenfalls wahr ist.
> 2. Bei der Kontradiktion versucht man, die negierte Aussage zu beweisen, indem man annimmt, dass sie falsch ist. Dann leitet man aus dieser Annahme Schlussfolgerungen ab, die zu einem Widerspruch führen. Der Widerspruch zeigt, dass die ursprüngliche Aussage wahr sein muss, da die Annahme ihrer Negation zu einem Konflikt führt.
>
> In beiden Fällen wird die negierte Aussage in irgendeiner Form verwendet, aber der Schluss und die logischen Schritte sind unterschiedlich, um die ursprüngliche Aussage zu beweisen.

Mengen

Eine Menge ist eine ungeordnete Zusammenfassung definierter, unterscheidbarer Objekte, genannt Elemente, zu einem Ganzen. Für ein Objekt x gilt dann bezüglich der Menge A entweder $x \in A$ oder dann $x \notin A$.

Endliche Mengen lassen sich durch das Aufzählen der in ihnen enthaltenen Elemente beschreiben. Zum Beispiel die Menge aller natürlichen Zahlen kleiner als 80: $A = \{0, 1, 2, ..., 99, 100\}$. Hier ist $99 \in A$ aber $101 \notin A$.

Beschreibende Schreibweisen für Mengen sind:

$A = \{n \in \mathbb{N} | n < 101\} = \{n \in \mathbb{N} : n \leq 100\} = \{n | n \in \mathbb{N} \wedge n \leq 100\}$

> 💡 Zwei Mengen A und B sind gleich (A = B), wenn sie die selben Elementen beinhalten. Also wenn $(A \subset B) \wedge (B \subset A)$.

Bekannte Mengen

$\mathbb{N} = \{0, 1, 2, 3, ...\}$	Menge der natürlichen Zahlen $(\mathbb{N}^* = \mathbb{N} \backslash \{0\})$
$\mathbb{Z} = \{..., -3, -2, -1, 0, 1, 2, 3, ...\}$	Menge der ganzen Zahlen
$\mathbb{Z}^+ = \{1, 2, 3, ...\} = \mathbb{N}^*$	Menge der positiven ganzen Zahlen
$\mathbb{Q} = \{\frac{p}{q} \| p \in \mathbb{Z} \wedge q \in \mathbb{N} \backslash \{0\}\}$	Menge der Brüche
$\mathbb{R}$ z.B. $\pi \in \mathbb{R}; \sqrt{2} \in \mathbb{R}; e \in \mathbb{R};$	Menge der reellen Zahlen
$\mathbb{C}$ z.B. $i \in \mathbb{C}; 2 - 1.5i \in \mathbb{C};$	Menge der komplexen Zahlen

💡 Es braucht $\mathbb{R}$, weil die Gleichung $x^2 = 2$ hat in $\mathbb{Q}$ keine Lösung. Analog braucht es $\mathbb{C}$, denn die Gleichung $x^2 = -1$ hat keine Lösung in $\mathbb{R}$. $i^2 = i \cdot i = -1$

Spezielle Mengen

- Teilmenge: Eine Teilmenge beinhaltet alle Elemente einer anderen Menge. Also z.B: ist A eine Teilmenge von B, wenn $\forall x(x \in A \rightarrow x \in B)$. Also gilt $A \subset B$.
- Leere Menge $\emptyset$: Für jede Menge A gilt: $\emptyset \subset A$
- Kardinalität: Ist S eine endliche Menge, dann bezeichnet $|S|$ die Kardinalität (Anzahl Elemente) von S. Eine nicht endliche Menge heisst unendliche Menge.
- Potenzmenge: Die Potenzmenge $P(S)$ oder 2^S der Menge S besteht aus der Menge aller Teilmengen $A \subset S$.

Bestimmen der Potenzmenge von $S = \{1, 2\}$

$P(S) = 2^S = \emptyset, \{1\}, \{2\}, \{1, 2\}\}.$

Die Menge S besteht also aus der 0-elementigen Teilmenge von $S : \{\ \}$, der 1-elementigen Teilmenge von $S : \{1\}\{2\}$ und 2-elementigen Teilmenge von $S : \{1, 2\}$. Es gilt allgemein $|2^S| = 2^{|S|}$. In diesem Beispiel also: $|2^{\{1,2\}}| = 2^{|\{1,2\}|} = 2^2 = 4$

Endliche Mengen mit dem Computer darstellen

Wenn man endliche Mengen mit dem Computer darstellen möchte, können Bitstrings verwendet werden. Dazu weist man jedem Element der Menge ein Bit im Bitstring zu. Also in der Menge

$S = \{1, 2, a, *, o, \oplus\}$ machen wir die Zuteilung wie folgt:

1	2	a	*	o	⊕
0	0	0	0	0	0

Um einige Teilmengen aufzuzählen, können wir nun dies tun:

$\emptyset =: 000000$	$\{1\} =: 100000$	$\{2\} =: 010000$	$\{a\} =: 001000$
$\{\oplus\} =: 000001$	$\{2, *\} =: 010100$	$\{1, 2, a, \oplus\} =: 111001$	$\{1, 2, a, *, o\} =: 111110$

Wie man also sieht, zeigt eine 1 im Bitstring an einer bestimmten Stelle, dass das Element aus der Tabelle an der selben Stelle in dieser Teilmenge vorhanden ist. Eine 0 zeigt, dass dieses Element in dieser Teilmenge fehlt. S kann also so beschrieben werden:

$S = \{1, 2, a, *, o, \oplus\}$ als Bitstring: 111111

Die Menge S hat genau so viele Teilmengen wie es Bitstrings der Länge 6 gibt. Es gibt für jedes Bit, unabhängig von den anderen Bits 2 Möglichkeiten, 0 und 1. Somit ist die Anzahl Möglichkeiten und Teilmengen von 2^6, was auch zeigt, dass die Formel $|2^s| = 2^{|S|}$ stimmt.

Kartesische Produkt (Kreuzprodukt) zweier Mengen

Das Kartesische Produkt zweier Mengen ergibt jede Mögliche Kombination aus allen Elementen in zweier Paaren dieser Menge.

Die mathematische Definition des kartesischen Produkts ist: $A \times B = \{(a, b) | a \in A \wedge b \in B\}$.

Beispiel:

Das kartesische Produkt aus den Mengen A und B, wenn $A = \{1, 2, 3\}$ und $B = \{a, b\}$ ist:

$$A \times B = \{(1, a), (2, a), (3, a), (1, b), (2, b), (3, b)\}$$
$$B \times A = \{(a, 1), (a, 2), (a, 3), (b, 1), (b, 2), (b, 3)\}$$

Bei Tupeln ist die Reihenfolge wesentlich, das heisst $(1, a) \neq (a, 1)$. Dementsprechend ist also $A \times B \neq B \times A$. Also gilt das Kommutativgesetz beim Kreuzprodukt nicht.

In **Haskell** kann man dies mithilfe **List Comprehensions** wie folgt implementieren:

`[(a,b) | a <- [1..3], b <- ["a","b"]]` , was der mathematischen Definition recht ähnelt.

Wobei die Menge A die Zahlen von 1 bis und mit 3 enthält, und die Menge B die Buchstaben “a” und “b”. Das Ergebnis von
$A \times B$ ist dann:

`[(1,"a"),(1,"b"),(2,"a"),(2,"b"),(3,"a"),(3,"b")]`
Das ist das selbe Resultat, wie wenn wir dies mathematisch berechnen, nur die Reihenfolge ist etwas anderst.

Dies könnte man in Haskell so korrigieren:
`[(a,b) | b <- ["a","b"], a <- [1..3]]`
Dann stimmt die Reihenfolge:
`[(1,"a"),(2,"a"),(3,"a"),(1,"b"),(2,"b"),(3,"b")]`

Natürlich kann auch das Ergebnis von
$B \times A$ berechnet werden:

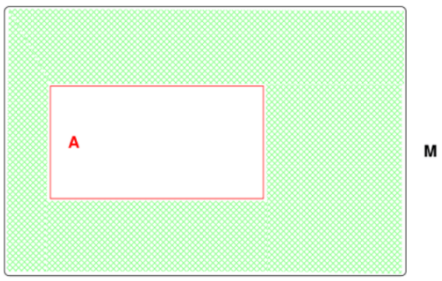
`[(a,b) | a <- ["a","b"], b <- [1..3]]` gibt dann:

`[("a",1),("a",2),("a",3),("b",1),("b",2),("b",3)]`

Mengenoperationen

Mengenoperationen verknüpfen Mengen zu neuen Mengen, indem Eigenschaften der zu konstruierenden Mengen definiert werden. Dazu gibt es einige Operationen.

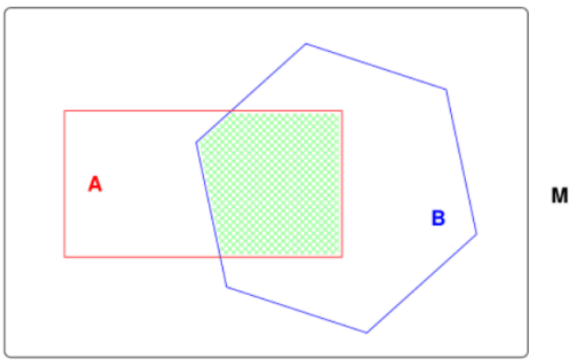
Komplement



Wenn A eine Teilmenge der Menge M ist, bezeichnet man
$A^c = \overline{A} = \{ m \in M \mid m \notin A \}$ als das Komplement von A bezüglich M.

Also alle Elemente, die keine Teilmenge von A sind. Aber poteziell in der Gesamtheit aller möglichen Elemente M vorhanden sind.

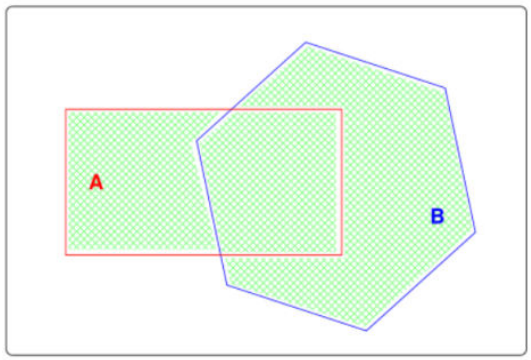
Durchschnitt



Sind A und B Teilmengen einer Menge M, bezeichnet man:
$A \cap B = \{ m \in M \mid m \in A \wedge m \in B \}$
als den Durchschnitt von A und B.

Also alle Elemente aus M, die zu beiden Teilmengen A und B gehören.

Vereinigung



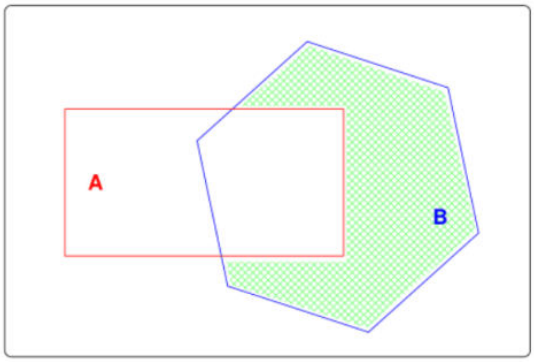
Sind A und B Teilmengen der Menge M, bezeichnet man

$$A \cup B = \{ m \in M \mid m \in A \vee m \in B \}$$

als die Vereinigung von A und B.

> Also alle Elemente aus M, die in mindestens einer der Teilmengen A oder B enthalten sind.

Differenz



Sind A und B Teilmengen einer Menge M, bezeichnet man

$$B \setminus A = \{ m \in M \mid m \in B \wedge m \notin A \}$$

als Differenz.

> Also alle Elemente aus der Menge M, die zur Menge B gehören. Ohne die Elemente die auch zur Menge A gehören.

> Die Rechenregeln sind die selben, wie sie es bei den logischen Äquivalenzen sind.

Funktionen

> 💡 Da Funktionen schon auch Bestandteil des Moduls Anaylsis waren, werde ich hier nicht all zu genau auf Funktionen eingehen. Diese sind aber in der Formelsammlung und Summary von Analysis beschrieben: https://github.com/omeldar/hslu/blob/main/I.BA_ANLIS.H22/I.BA_ANLIS.H22 - Formelsammlung - Eldar Omerovic 2023.pdf

Eine Funktion ist die Zuordnung einer Menge Y, für jedes Element einer Menge X.

Beispiel: ceiling- und floorfunction

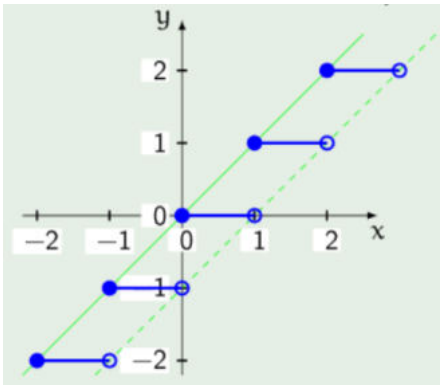
Die ceiling-function ordnet der reellen Zahl x die nächst grössere ganze Zahl n zu.

Definition

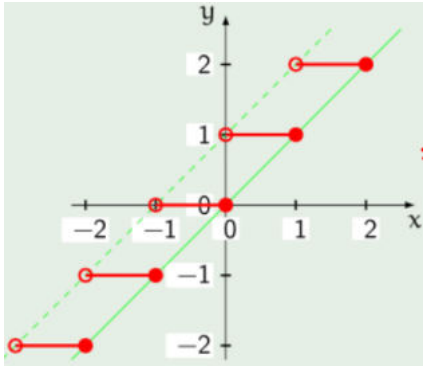
$$\lceil \cdot \rceil : \mathbb{R} \to \mathbb{Z}, x \mapsto \lceil x \rceil = min\{\, n \in \mathbb{Z} \mid x \leq n \,\}$$
$$\lfloor \cdot \rfloor : \mathbb{R} \to \mathbb{Z}, x \mapsto \lfloor x \rfloor = max\{\, n \in \mathbb{Z} \mid n \leq x \,\}$$

Die erste Definition ist die Definition von ceiling "nächst grössere ganze Zahl n nach x" und die zweite Definition ist die Definition der floor-function "nächst kleinere ganze Zahl n nach x".

Floor-function



Ceiling-function



Injektive, Surjektive und Bijektive Funktionen

- Injektive Funktionen: Eine Funktion ist injektiv, wenn verschiedene $x_1, x_2 \in X$ stets auf verschiedene Werte im Wertebereich abgebildet werden. Also für jeden y-Wert, genau ein x-Wert. Also ist x^2 nicht injektiv, aber eine Funktion wie $2x + 3$ ist injektiv.
- Surjektive Funktionen: Eine Funktion ist surjektiv, wenn für jedes Element $y \in Y$ (mindestens) ein Element $x \in X$ existiert, so dass $f(x) = y$ gilt.
- Bijektive Funktionen: Eine Funktion ist bijektiv, wenn sie injektiv und surjektiv ist. Diese Funktionen sind Umkehrbar. Aus $f(x) = f(5) = 10$ ergibt sich $f^{-1}(10) = 5$

Zusammengesetze Funktionen

Funktionen können zusammengesetzt werden. Funktionen werden zusammengesetzte Funktionen genannt, wenn der Wertebereich von g im Definitionsbereich von f enthalten ist. Dann kann man also die sogenannte Komposition (zusammengesetze Funktion) von f und g bilden: $f \circ g$

Dies ist das selbe, wie wenn wir $f(g(x))$ schreiben würden. Nur wird es leserlicher, je mehr Funktionen wir zusammenhängen müssen, wenn wir die erste Schreibweise verwenden.

Folgen

> 💡 Da Folgen schon auch Bestandteil des Moduls Anaylsis waren, werde ich hier nicht all zu genau auf Folgen eingehen. Diese sind aber in der Formelsammlung und Summary von Analysis beschrieben: https://github.com/omeldar/hslu/blob/main/I.BA_ANLIS.H22/I.BA_ANLIS.H22 - Formelsammlung - Eldar Omerovic 2023.pdf

Eine Folge ist eine Abbildung von $\mathbb{N}$ oder auch $\mathbb{N}^* = N \setminus \{0\}$ in eine Menge A.

Man nennt a_n das Glied der Folge mit der Nummer n.

Geometrische Folge

Bei einer geometrischen Folge (Progression) ist der Quotient zweier aufeinander folgender Glieder immer gleich. Es existiert also ein konstanter Quotient zwischen Gliedern (q).

Beispiel: $1, 2, 4, 8, 16, 32, 64, 128, 256, 512...$ ist eine geometrische Folge, da ein konstanter Quotient 2 zwischen den Gliedern existiert.

Eine rekursive Definition der geometrischen Folge wäre: $a_{k+1} = a_k q$

Geometrische Folgen nähern sich oft einem Wert an.

Arithmetische Folge

Bei einer arithmetischen Folge ist die Differenz zweier aufeinander folgender Glieder immer gleich (d).

Beispiel: $1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, ...$ ist eine arithmetische Folge, da die Differenz zwischen zwei Gliedern immer $+1$ ist.

Eine rekursive Definition der arithmetischen Folge wäre: $a_{k+1} = a_k + d$

Reihen

> 💡 Da Reihen sowie das Summenzeichen schon auch Bestandteil des Moduls Anaylsis waren, werde ich hier nicht all zu genau auf diese eingehen. Diese sind aber in der Formelsammlung und Summary von Analysis beschrieben: https://github.com/omeldar/hslu/blob/main/I.BA_ANLIS.H22/I.BA_ANLIS.H22 - Formelsammlung - Eldar Omerovic 2023.pdf
> Ich werde jedoch detailliert auf das Produktzeichen eingehen, da dies in Analysis nicht behandelt wurde.

Eine Reihe ist die Aggregation von Elementen aus einer Folge durch eine bestimmte mathematische Operation.

Summation

Das Summenzeichen ermöglicht es Summen einfacher zu schreiben:

$$a_m + a_{m+1} + a_{m+2} + ... + a_n = \sum_{j=m}^{n} a_j$$

Der Injdex j wird Summationsindex, m wird untere und n wird obere Grenze der Summe genannt.

Addiert man die Glieder einer arithmetischen Folge (a_k), entsteht die arithmetische Reihe:

$$\sum_{k=0}^{n-1} a_0 + d \sum_{k=0}^{n-1} k = na_0 + d\frac{n(n-1)}{2}$$

Addiert man die Glieder einer geometrische Folge (a_k), entsteht die geometrische Reihe:
(wobei
$q \neq 1$ gelten muss, da wenn $q = 1$ wäre, man eine arithmetische Reihe hätte)

$$\sum_{k=0}^{n-1} q^k = a_0 \frac{q^n - 1}{q - 1}$$

Nützliche Summenformen

$$\sum_{k=1}^{n} k = \frac{n(n+1)}{2}$$
$$\sum_{k=1}^{n} k^2 = \frac{n(n+1)(2n+1)}{6}$$
$$\sum_{k=1}^{n} k^3 = \frac{n^2(n+1)^2}{4}$$
$$\sum_{k=0}^{\infty} x^k, |x| < 1 = \frac{1}{1-x}$$
$$\sum_{k=0}^{\infty} kx^{k-1}, |x| < 1 = \frac{1}{(1-x)^2}$$
$$\sum_{k=0}^{n-1} q^k = \frac{q^n - 1}{q - 1} \overset{\text{(oder)}}{=} \sum_{k=0}^{n} q^k = \frac{q^{n+1} - 1}{q - 1}$$

Produkt

Durch das Produktzeichen lassen sich Produkte einfacher schreiben.

$$a_m \cdot a_{m+1} \cdot a_{m+2} \cdots a_n = \prod_{j=m}^{n} a_j \quad n \geq m$$

Der Index j geht hier von der unteren Grenze m bis zur oberen Grenze n. Wie auch beim Summenzeichen kann majn hier eine Indextransofmration durchführen

$$n! = \begin{cases} 1 & n = 0 \\ n(n-1)(n-2) \cdots 2 \cdot 1 = \prod_{k=1}^{n} k & n > 0 \end{cases}$$

$\Rightarrow 0! = 1; 1! = 1; 2! = 2 \cdot 1 = 2; 3! = 3 \cdot 2 \cdot 1 = 6; 4! = 4 \cdot 3! = 24$

Wachstum von Funktionen (Big-O)

Seien f und g Funktionen von $\mathbb{Z}$ oder $\mathbb{R}$ nach $\mathbb{R}$. Dann sagt man $f(x)$ ist $\mathcal{O}(g(x))$, falls es Konstanten C und k gibt, so dass gilt:

$$|f(x)| \leq C|g(x)|, \quad \forall x > k$$

Wir versuchen mit der Big-O Abschätzung einen möglichst kleinen Wert zu finden für die Funktion von $f(x)$ der am schnellsten wächst. Dieser Wert beschreibt dann den Wachstum dieser Funktion.

Beispiel einer Big-$\mathcal{O}$ Abschätzung:

Wir zeigen hier, dass $f(x) = x^2 + 2x + 1$ eine Big-$\mathcal{O}$ Abschätzung von $\mathcal{O}(x^2)$ hat.

Wir betrachten dabei nur die reellen Zahlen x mit $x > 1$. Für diese Zahlen gilt auch $x^2 > x$ und $x^2 > 1$ und weiterhin, da f in diesem Bereich nur positive Werte annehmen kann,

$$|f(x)| = |x^2 + 2x + 1| = x^2 + 2\underbrace{x}_{<x^2} + \underbrace{1}_{<x^2} \leq x^2 + 2x^2 + x^2 = 4x^2$$

Insgesamt haben wir also gezeigt, für alle $x > \underbrace{1}_{=k}$ gilt:

$$\underbrace{x^2 + 2x + 1}_{=|f(x)|} \leq \underbrace{4}_{=C} \quad \underbrace{|x^2|}_{=|g(x)|}$$

also $f(x) = x^2 + 2x + 1$ ist $\mathcal{O}(x^2)$ mit den Zeugen $k = 1$ und $C = 4$.

Rechenregeln

Diese Rechenregeln werden oft benötigt, wenn es darum geht, Funktionen Big-O abzuschätzen.

Logarithmus Rechenregeln

$log_a(u \cdot v) = log_a(u) + log_a(v)$

$log_a(\frac{u}{v}) = log_a(u) - log_a(v)$

$log_a(u^w) = w \cdot log_a(u)$

$(log(u))^2 = log^2(u)$

Wenn ein Term ohne Log-Funktion erhöht wird auf einen Term mit Log-Funktion wird $\forall n \geq a$ und somit auch $k = a$.

Potenzregeln

$a^n \cdot a^m = a^{n+m}$

$a^n \cdot b^n = (a \cdot b)^n$

$(a^n)^m = a^{n \cdot m}$

$\frac{a^n}{b^n} = (\frac{a}{b})^n$

$\frac{a^n}{a^m} = a^{m-n}$

Zahlen und Division

Falls $a, b \in \mathbb{Z}$ mit $a \neq 0$ dann sagt man: a teilt b, falls $\exists c(b = ac)$ in der Universalmenge $\mathbb{Z}$. Dann ist a ein Faktor von b und b ein Vielfaches von A. Man schreibt dann $a|b$ und andernfalls $a \nmid b$

Falls $a, b, c \in \mathbb{Z}$, dann gilt:

$$a|b \wedge a|c \rightarrow a|(b+c)$$
$$a|b \rightarrow \forall c(a \neg bc)$$
$$a|b \wedge b|c \rightarrow a|c$$

Mersenne Primzahlen:

$2^{Primzahl} - 1$

Abschätzung Primzahlen ≤ x: $\pi(x) \approx \frac{x}{ln(x)}$

Falls $a \equiv b \mod m$ und $c \equiv d \mod m$ dann gillt:

$a + c \equiv b + d \mod m$ *und*

$ac \equiv bd \mod m$.

Primzahlen

Eine positive Zahl $n \in \mathbb{Z}$ wenn $n > 1$ heisst Primzahl, wenn Sie lediglich die Faktoren 1 und n hat. Andernfalls heisst die Zahl zusammengesetzt: in diesem Fall gilt: $\exists a(a|n \wedge (1 < a < n))$.

Falls n eine zusammengesetzte Zahl ist, dann hat n (mind.) einen Primzahlteiler kleiner gleich $\sqrt{n}$. Das bedeutet: Um herauszufinden, ob n eine Primzahl ist, genügt es zu kontrollieren, ob eine der Primzahlen bis $\sqrt{n}$ die Zahl n teilt.

Es gibt unendlich viele Primzahlen. Der Beweis dafür kommt durch Widerspruch, wenn wir annehmen, dass es endlich viele Primzahlen gäbe:

Nehmen wir uns die Zahl Q. $Q = P_1 \cdot P_2 \cdots P_n + 1$, wobei $P_{1...n}$ Primzahlen sind. Nach dem Fundamentalsatz der Arithmetik ist Q entweder eine Primzahl oder kann als Produkt von zwei oder mehr Primzahlen geschrieben werden. Keine der Primzahlen teilt aber Q, da $Q \mod P_j = 1$ gilt. Also ist Q eine Primzahl. Diese ist aber grösser als jede P_j in der Liste. Dieser Widerspruch zur Annahme, dass die obige Aufzählung vollständig ist, ist also falsch.

ggT (gcd) und kgV (lcd)

Der **ggT** beschreibt den grössten gemeinsamen Teiler zweier ganzer Zahlen.

$$ggT(a, b) = max\{d \in \mathbb{N}^* : d|a \text{ und } d|b\}$$

Das **kgV** beschreibt die kleinste positive Zahl, die das gemeinsame Vielfache zweier Zahlen bildet.

$$kgV(a, b) = min\{m \in \mathbb{N}^* : a|m \text{ und } b|m\}$$

Das kgV kann berechnet werden mit: $kgV(a, b) = \frac{a \cdot b}{ggT(a, b)}$

Modulare Arithmetik

Definition von Kongruenz

Man nennt zwei ganze Zahlen a und b **kongruent modulo m** (Sei $m \in \mathbb{N} \setminus \{0\}$), falls $m|(a - b)$. Also wenn m die Differenz von a und b teilt. Man schreibt dann $a \equiv b \mod m$ und sagt "a ist kongruent zu b modulo m".

Auch sieht man, dass a und b beide das Selbe geben, wenn modulo m angewendet wird:

$a \equiv b \mod \leftrightarrow a \mod m = b \mod m$

Auch muss es eine Zahl k geben, welche multipliziert mit dem Modulowert und addiert mit b dann a gibt: $a \equiv b \mod m \leftrightarrow \exists k \in \mathbb{Z}(a = b + km)$.

Die Menge aller Zahlen kongruent zu $a \mod m$ heisst Kongruenzklasse von a modulo m.

Euklidische Algorithmus

Mit dem euklidischen Algorithmus berechnet man den ggT zweier Zahlen. Dies macht man in folgenden Schritten:

1. Man benötigt zwei positive ganzen Zahlen a und b (wobei $a > b$, ansonsten Zahlen vertauschen)
2. Nun wird a durch b geteilt, wobei der Rest behalten wird. Der Rest wird als nächste Zahl gebraucht, für diese man diesen Schritt wiederholt.
3. Wenn der Rest 0 gibt, ist der letzte Restwert bevor der Rest 0 war (die Zahl b) der ggT.

Das ganze an einem Beispiel für den ggT von $ggT(12345, 54321)$:

Wie man sieht, ist hier $a < b$, also tauschen wir die Zahlen zu $ggT(54321, 12345)$.

Nun berechnet man für diese zwei Zahlen den Rest:

$$
\begin{gathered}
a = 54321, \quad b = 12345 \\
54321 \bmod 12345 = 4941 \\
12345 \bmod 4941 = 2463 \\
4941 \bmod 2463 = 15 \\
2463 \bmod 15 = 3 \\
15 \bmod 3 = 0
\end{gathered}
$$

Matrizen

Addition:

$$A + B = \begin{bmatrix} a_a + a_b & b_a + b_b \\ c_a + c_b & d_a + d_b \end{bmatrix}$$

Subtraktion ist das addieren der negativen Werte von der Matrix B zu A.

$$A - B = A + (-1)B$$

$$A - B = \begin{bmatrix} a_a - a_b & b_a - b_b \\ c_a - c_b & d_a - d_b \end{bmatrix}$$

Transponierte Matrix

Transponierte Matrixen werden oft als A^T angegeben. Die Spalten werden zu Reihen, die Reihen zu Spalten:

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad A^T = \begin{bmatrix} a & c \\ b & d \end{bmatrix}$$

Multiplikation mit einer **Zahl**

$$3A = 3 \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} 3a & 3b \\ 3c & 3d \end{bmatrix}$$

Matrixmultiplikation

Bei der Matrixmultiplikation bildet man das Produkt jedes Paares aus der Zeile aus Matrix A mit der Spalte aus Matrix B. Diese Paarse addiert man dann.

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 1 \\ 4 & 2 \end{bmatrix} \quad B = \begin{bmatrix} -2 & 1 \\ 2 & -4 \end{bmatrix} \quad A \cdot B = \begin{bmatrix} 1 \cdot -2 + 2 \cdot 2 & 1 \cdot 1 + 2 \cdot -4 \\ 3 \cdot -2 + 1 \cdot 2 & 3 \cdot 1 + 1 \cdot -4 \\ 4 \cdot -2 + 2 \cdot 2 & 4 \cdot 1 + 2 \cdot -4 \end{bmatrix}$$

Die Matrixmultiplikation ist nicht kommutativ, also $AB \neq BA$

Inverse Matrix

Die inverse Matrix einer ursprünglichen Matrix, wenn multipliziert mit der ursprünglichen Matrix, soll die Einheitsmatrix ergeben.

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 1 \end{bmatrix} \quad I_2 = A^{-1} \cdot A$$

Um die inverse Matrix einer 2x2 Matrix zu finden, können wir Gleichungen aufstellen, nachdem wir die Matrix mit einer Variablen-Matrix für die gesuchten Faktoren zur Umwandlung zur inversen Matrix multiplizieren:

$$\begin{bmatrix} 1 & 2 \\ 3 & 1 \end{bmatrix} \cdot \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} a + 2c & b + 2d \\ 3a + c & 3b + d \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Nun löst man diese Gleichungen bestehend aus der mittleren (multiplizierten Matrix) und der Einheitsmatrix.

$$\begin{aligned} a + 2c = 1 &\implies a + 2(-3a) = 1 \implies -5a = 1 \implies a = -0.2 \\ b + 2d = 0 &\implies b = -2d \implies b = -2 \cdot -0.2 \implies b = 0.4 \\ 3a + c = 0 &\implies c = -3a \implies c = -3 \cdot -0.2 \implies c = 0.6 \\ 3b + d = 1 &\implies 3 \cdot (-2d) + d = 1 \implies -5d = 1 \implies d = -0.2 \end{aligned}$$

Diese Werte für a, b, c, d setzen wir nun für die Variablen in der inversen Matrix ein. Also haben wir dies als unsere inverse Matrix zu A.

$$A^{-1} = \begin{bmatrix} -0.2 & 0.4 \\ 0.6 & -0.2 \end{bmatrix}$$

Und nun gilt natürlich: $A \cdot A^{-1} = I_2$

> ⚠️ Es kann sein, dass man beim berechnen einer inversen Matrix feststellt, dass man für den selben Ausdruck unterschiedliche Resultate bekommt. z.B $a + c = 0, a + c = 1$. Dies kann vorkommen, dann geht aber das berechnen der inversen Matrix nicht. Somit gibt es keine inverse Matrix für diese Matrix.

Generelle Rechenregeln

$A + B = B + A$

$(A + B) + C = A + (B + C)$

$A + 0 = A$

$AB = 0 \nRightarrow A = 0 \text{ oder } B = 0$

$\lambda(A + B) = \lambda A + \lambda B, \quad \lambda \in \mathbb{R}$

$(A + B)C = AC + BC$

$(A^{-1})^{-1} = A$

$(A^T)^T = A$

$(AB)^T = B^T A^T$

Im Allgemeinen: $AB \neq BA$

$AB(C) = A(BC)$

$AI = IA = A$, (A quadratisch)

$AB = AC \nRightarrow B = C$

$A(B + C) = AB + AC$

$(AB)^{-1} = B^{-1}A^{-1}$

$(A + B)^T = A^T + B^T$

$(A^{-1})^T = (A^T)^{-1}$

$$\text{Für } A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \text{ mit } ad - bc \neq 0 \text{ gilt } A^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

Null-Eins Matrizen

Null-Eins Matrizen werden z.B. in der Graphentheorie verwendet.

Null-Eins Matrizen können mithilfe boolscher Operatoren verknüpft werden (z.B. $\wedge, \vee$).

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \text{ und } B = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \end{bmatrix},$$
$$A \vee B = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}, \quad A \wedge B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

Wie man sieht, wird der $\vee$ Operator auf jede Stelle in Matrix A mit der entsprechenden Stelle in Matrix B angewendet. Daraufhin entsteht eine neue Matrix $A \vee B$. Das selbe gilt für $\wedge$.

Das boolsche Produkt zweier Null-Eins Matrizen wird definiert durch:

$$A \odot B = [c_{i,j}] \text{ wobei } c_{i,j} = (a_{i,1} \wedge b_{1,j}) \vee (a_{i,2} \wedge b_{2,j}) \vee \cdots \vee (a_{i,n} \wedge b_{n,j})$$

Was dann in einem Beispiel so aussieht:

$$\begin{aligned} A \odot B &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \odot \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \\ &= \begin{bmatrix} (1 \wedge 1) \vee (0 \wedge 0) & (1 \wedge 1) \vee (0 \wedge 1) & (1 \wedge 0) \vee (0 \wedge 1) \\ (0 \wedge 1) \vee (1 \wedge 0) & (0 \wedge 1) \vee (1 \wedge 1) & (0 \wedge 0) \vee (1 \wedge 1) \\ (1 \wedge 1) \vee (0 \wedge 0) & (1 \wedge 1) \vee (0 \wedge 1) & (1 \wedge 0) \vee (0 \wedge 1) \end{bmatrix} \\ &= \begin{bmatrix} 1 \vee 0 & 1 \vee 0 & 0 \vee 0 \\ 0 \vee 0 & 0 \vee 1 & 0 \vee 1 \\ 1 \vee 0 & 1 \vee 0 & 0 \vee 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix} \end{aligned}$$

Mathematische Induktion

Induktionsbeweis

1. Induktionsverankerung: Es wird gezeigt, dass P(1) wahr ist.
2. Induktionsschritt: Es wird gezeigt, dass die IMplikation $P(k) \rightarrow P(k+1)$ wahr ist $\forall k \geq 1$
   a. Induktionsvoraussetzung: Die Aussage P(k) ist wahr für ein bestimmtes k ≥ 1.
   b. Induktionsbehauptung: Die Aussage P(k+1) ist wahr
   c. Beweis des Induktionsschritts: Beweise, dass unter Annahme der Induktionsvoraussetzung die Induktionsbehauptung whar ist:

$$[P(1) \wedge \forall k(P(k) \rightarrow P(k+1))] \rightarrow \underbrace{\forall n P(n)}_{\text{kann nicht falsch sein}}$$

Sei $P(k)$ die Aussage "der k-te Dominostein fällt". Wann fallen alle Dominosteine? Wann gilt also $\forall k P(k)$? In der Induktionsverankerung müssen wir zeigen, dass $P(1)$ wahr ist. Im Induktionsschritt zeigen wir, dass falls der k-te Dominostein fällt (Induktionsvoraussetzung), muss auch der nächste Dominostein, also der $(k+1)$-te fallen (Induktionsbehauptung). Für ein beliebiges k ≥ 1, muss also gelten: $P(k) \rightarrow P(k+1)$.

Wenn wir z.B. folgendes mittels Induktion beweisen wollten,

$$\forall n \in \mathbb{N} \setminus \{0\} \quad \underbrace{(\sum_{i=n_0}^{n} i = \frac{n(n+1)}{2})}_{P(n)}$$

müssen wir zuerst zeigen, dass dies für das erste Element n_0 (hier: 1) in $n \in \mathbb{N} \setminus \{0\}$ wahr ist. Dies machen wir, in dem wir für $n = 1$ einsetzen, und beide Seiten berechnen. Wenn diese gleich sind, haben wir bewiesen, dass für $P(n_0)$ (oder auch $P(1)$) die Aussage stimmt.

$$IV) \quad P(1): \quad 1 = \frac{1(1+1)}{2}, \quad \text{ist wahr}$$

Nun bereiten wir den Induktionsschritt vor. Dazu benötigen wir einmal die Aussage, und schreiben diese dann für $n+1$ um.

$$IS) \quad P(n): \quad (\sum_{i=n_0}^{n} i = \frac{n(n+1)}{2}), \quad \text{sei wahr, für ein n}$$
$$P(n+1): \quad \sum_{i=n_0}^{n} i + (n+1) = \frac{n(n+1)}{2} + (n+1)$$

Jetzt müssen zeigen, dass wir für $n+1$ den gleichen Term erhalten, wie für n_0 in der Induktionsbehauptung. Dazu nehmen wir eine Seite und formen diese so um, dass wir den selben Term für $n+1$ erhalten.

Setzen wir für n direkt $n+1$ ein. Dann sehen wir, auf was wir etwa hinarbeiten müssen.

$$P(n+1): \frac{(n+1)(n+1+1)}{2} = \frac{(n+1)(n+2)}{2}$$

Und nun formen wir den Term der Induktionsbehauptung um:

$$= \frac{n(n+1)}{2} + (n+1) = \frac{n(n+1) + 2 \cdot (n+1)}{2} = \frac{(n+1)(n+2)}{2}$$

Somit haben wir gezeigt, dass diese Aussage für n_0 und für $n+1$, und somit für alle Werte für $n \in \mathbb{N} \setminus \{0\}$ stimmt.

Anhand einem anderen Beispiel

Hier wird weniger Schritt für Schritt erklärt, es wird jedoch konkreter auf das Beispiel eingegangen.

Wir wollen folgendes beweisen:

$$\forall n \in \mathbb{N} \setminus \{0\}(\sum_{i=n_o}^{n} 7^{i-1} = \frac{7^n - 1}{7 - 1})$$

Zuerst für n_0:

$$P(1) = 7^{1-1} = \frac{7^1 - 1}{7 - 1} \to 7^0 = \frac{6}{6} \to 1 = 1$$

Nun bestimmen wir die Erwartungsform für $P(n+1)$

$$\frac{7^{n+1} - 1}{7 - 1}$$

Und nun WICHTIG, die Induktionsbehauptung unbedingt aufstellen und aufschreiben, auch wenn die selbe Formel in dem zu beweisendem Term schon steht!

$$P(n) : \sum_{i=n_0}^{n} 7^{n-1} = \frac{7^n - 1}{7 - 1}, \text{sei wahr für ein n}$$

Und jetzt das umwandeln der Terme in die Erwartungsform. Hier ist wichtig zu sagen: Wir müssen den dazukommenden Summanden zur Reihe, der Funktion (dem rechten Term) genau gleich auch als Summanden hinzufügen. Da links in der Summe der nächste Summand für $n + 1 = 7^n$ ist, fügen wir dies auch rechts hinzu und versuchen dann auf den gewünschten Term zu kommen.

$$\frac{7^n - 1}{7 - 1} + 7^n = \frac{7^n - 1 + 7^n(7 - 1)}{7 - 1} = \frac{7^n - 1 + 7^n \cdot 7 - 7^n}{7 - 1} = \frac{7^{n+1} - 1}{7 - 1}$$

Nun ist es bewiesen. $P(n)$ ist wahr.

Rekursiv definierte Funktionen

Ist f eine Funktion mit Definitionsbereich $D(f) = \mathbb{N}$, für die $f(0)$ definiert ist und für die eine Vorschrift existiert, die den Wert $f(k)$ aus $f(k-1), f(k-2), ...f(1), f(0)$ berechnet, dann wird diese Funktion **rekursive** oder **induktive Funktion** genannt.

z.B die Fibonacci-Funktion ist eine rekursive Funktion. Denn für jedes f_{n+1} gilt $f_{n+1} = f_n \cdot f_{n-1}$

Rekursive Algorithmen

Ein Algorithmus ist rekursiv, wenn er ein Problem löst, inem er dazu ein (oder mehrere) gleiche, aber kleinere Probleme löst.

Folgende Algorithmen können gut rekursiv implementiert werden: Fakultät (n!), n-te Potenz (a^n), ggT (Euklidischer Algorithmus), Türme von Hanoi, Fibonacci Zahlen.

z.B. um Fakultät zu berechnen, können wir immer wieder das gleiche machen:

n	i	fact
3		1
	1	$1 \cdot 1 = 1$
	2	$1 \cdot 2 = 2$
	3	$2 \cdot 3 = 6$

Wir machen hier immer das gleiche: Die Funktion ruft sich so oft selber auf, bis sie einen fixen Wert hat (hier 1). Danach wird der Wert durch alle Funktionsaufrufe zurückgegeben und in jedem Schritt wird das vorherige Resultat für die neue Berechnung verwendet. Bis am Schluss der letzte (erste) Funktionsaufruf endet und das Resultat zurückgibt.

Auch sieht man hier, was passiert, wenn wir $0!$ aufrufen. Die Funktion ruft sich einmal auf, $f(0)$ was dann sofort 1 zurückgibt, da $n = 0$ dann auch immer $i = 0$.

Inferenzregeln

Bei **mathematischen Beweisen** geht es darum, mit gültigen Argumenten, die Richtigkeit einer mathematischen Aussage zu begründen. Ein **Argument** ist eine Folge von Aussagen, die mit einer **Folgerung** enden. **Gültig heisst**, dass die Folgerung (Konklusion) aus der Wahrheit der vorhergehenden Aussagen (den **Prämissen**) des Arguments folgt. Ein Argument ist also dann und nur dann gültig, wenn es unmöglich ist, dass alle Prämissen wahr sind, die Folgerung aber falsch ist.

Prämissen und Folgerungen

Prämissen sind Aussagen. Deren Richtigkeit wird verwendet um Folgerungen aus ihnen zu schliessen. Prämissen könnten so aussehen:

- Falls Sie ein aktuelels Passwort haben, dann können Sie sich ins Netzwerk einloggen (q)
- Sie haben ein aktuelles Passwort (p)

Demzufolge kann eine Folgerung aus diesen Prämissen gemacht werden:

- Sie können sich in Netzwerk einloggen ($p \rightarrow q$)

Das ganze Argument wird in Kurzform wie folgt geschrieben.

$$
\begin{array}{l}
p \rightarrow q \\
p \\
\therefore q
\end{array}
$$

> 💡 $\therefore$ ist das Symbol für "**demzufolge**" oder "**folglich**" (engl. "therefore")

Regeln

Modus ponens (Abtrennungsregel)

$$
\begin{array}{l}
p \\
p \rightarrow q \\
\therefore q
\end{array}
$$

$T \equiv [p \wedge (p \rightarrow q)] \rightarrow q$

Modus tollens (Aufhebender Modus)

$$
\begin{array}{l}
\neg q \\
p \rightarrow q \\
\therefore \neg p
\end{array}
$$

$T \equiv [\neg q \wedge (p \rightarrow q)] \rightarrow \neg p$

Hypothetischer Syllogismus

$$
\begin{array}{l}
p \rightarrow q \\
q \rightarrow r \\
\therefore p \rightarrow r
\end{array}
$$

$T \equiv [(p \rightarrow q) \wedge (q \rightarrow r)] \rightarrow (p \rightarrow r)$

Disjunktiver Syllogismus

$$
\begin{array}{l}
p \vee q \\
\neg p \\
\therefore q
\end{array}
$$

$T \equiv [(p \vee q) \wedge \neg p] \rightarrow q$

Addition

$$
\begin{array}{l}
p \\
\therefore p \vee q
\end{array}
$$

$T \equiv p \rightarrow (p \vee q)$

Simplifikation

$$
\begin{array}{l}
p \wedge q \\
\therefore p
\end{array}
$$

$T \equiv (p \wedge q) \rightarrow p$

Konjunktion

$$
\begin{array}{l}
p \\
q \\
\therefore p \wedge r
\end{array}
$$

$T \equiv [(p) \wedge (q)] \rightarrow (p \wedge q)$

Resolution

$$
\begin{array}{l}
p \vee q \\
\neg p \vee r \\
\therefore q \vee r
\end{array}
$$

$T \equiv [(p \vee q) \wedge (\neg p \vee r)] \rightarrow (q \vee r)$

Anwendungsbeispiel

Wir verwenden:

n = "Es ist heute Nachmittag sonnig"

k = "Es ist kälter als gestern"

s = "Wir gehen schwimmen"

f = "Wir machen eine Kanufahrt"

h = "Wir sind bis Sonnenuntergang zu Hause"

Dann lauten die Hypothesen:

- (H1) $\neg n \land k =$ "es ist heute Nachmittag nicht sonnig und es ist kälter als gester
- (H2) $s \to n =$ "wir gehen nur schwimmen, falls es sonnig ist
- (H3) $\neg s \to f =$ "wenn wir nicht schwimmen gehen, machen wir eine Kanufahrt"
- (H4) $f \to h =$ " wenn wir eine Kanufahrt machen, werden wir bis Sonnenuntergang zu Hause sein"

Um zu zeigen, dass aus diesen Hypothesen durch Anwendung der Schlussregeln h folgt, machen wir folgendes.

1. $\neg n \land k$	Hypothese (H1)
2. $\neg n$	Simplifikation unter Verwendung von (1)
3. $s \to n$	Hypothese (H2)
4. $\neg s$	Modus tollens unter Verwendung von (2) und (3)
5. $\neg s \to f$	Hypothese (H3)
6. f	Modus ponens unter Verwendung von (4) und (5)
7. $f \to h$	Hypothese (H4)
8. h	Modus ponens unter Verwendung von (6) und (7)

Dieses Problem hätte man auch mit einer Wahrheitstabelle lösen können. Allerdings hätte man durch die 5 Variablen $2^5 = 32$ Zeilen gehabt.

Schlussregeln für quantifizierte Aussagen

Universal instantiation

$$\begin{array}{l} \forall x P(x) \\ \therefore P(c) \end{array}$$

Universal generalization

$$\begin{array}{l} P(c), \text{for an arbitrary c} \\ \therefore \forall x P(x) \end{array}$$

Existential instantiation

$$\begin{array}{l} \exists x P(x) \\ \therefore P(c), \text{ for some element c} \end{array}$$

Existential generalization

$$\begin{array}{l} P(c), \text{ for some element c} \\ \therefore \exists P(x) \end{array}$$

Grundlagen des Zählens

Produktregel

Produktregel

Für kombinierte Entscheidungen.

$n = n_1 \cdot n_2$ n_1 und n_2

Angewendet, zur Berechnung der Möglichkeiten von zwei unabhängigen Ereignissen.

Beispiel Produktregel

Wie viele Nummernschilder lassen sich herstellen, wenn jedes Schild aus einer Folge von drei Buchstaben und drei Ziffern besteht: $AAA000, AAA001, ..., CDX228, ..., ZZZ999$

Für die 3 Buchstaben gibt es je 26 Optionen, also 26^3 verschiedene, dreistellige Buchstabenkombinationen. Für die drei Ziffern, gibt es 10^3 Arten, diese darzustellen.

Total also $26^3 + 10^3 = 17'576'000$ unterschiedliche Nummernschilder

> 💡 Ein Beispiel in dem die Produkt- und die Summenregel angewendet wird, ist im nächsten Kapitel "Summenregel" enthalten.

Summenregel

Summenregel

Für entweder/oder Entscheidungen.

$n = n_1 + n_2$ n_1 oder n_2

Angewendet, zur Berechnung der Möglichkeiten von Ereignissen die nicht gleichzeitig eintreten können.

Beispiel Summenregel

Ein Student kann seine Projektarbeiten aus drei Listen mit je 23, 15 und 19 Arbeiten auswählen. Aus wie vielen Projekte kann er auswählen?

$$|\text{Liste1} \cup \text{Liste2} \cup \text{Liste3}| = |\text{Liste1}| \cup |\text{Liste2}| \cup |\text{Liste3}| = 23 + 15 + 19 = 57$$

Der Student kann also aus **57** Projekten gesamthaft auswählen.

Beispiel Summen- und Produktregel

Man plant ein Wochenende und hat folgende Optionen

- Einen Tag draussen verbringen (Samstag oder Sonntag), mit Wahl zwischen Wandern oder Radfahren
- Beide Tage zu Hause bleiben, mit Wahl zwischen Buch lesen oder Film schauen.

Für die Optionen draussen: 2 Tage, 2 Aktivitäten $= 2 \cdot 2 = 4$ Möglichkeiten (**Produktregel**)

Für die Optionen Zuhause: 2 Tage, 2 Aktivitäten $= 2 \cdot 2 = 4$ Möglichkeiten (**Produktregel**)

Insgesamt also $4 + 4 = 8$ Möglichkeiten (**Summenregel**)

Das Einschluss-/Ausschlussprinzip

Für 2 beliebige Mengen A und B gilt:

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

Für 3 beliebige Mengen A, B und C gilt:

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

Beispiel Einschluss-/Ausschlussprinzip

Wie viele Bitstrings der Länge 8 starten mit einer 1, oder enden mit den beiden Bits 00?

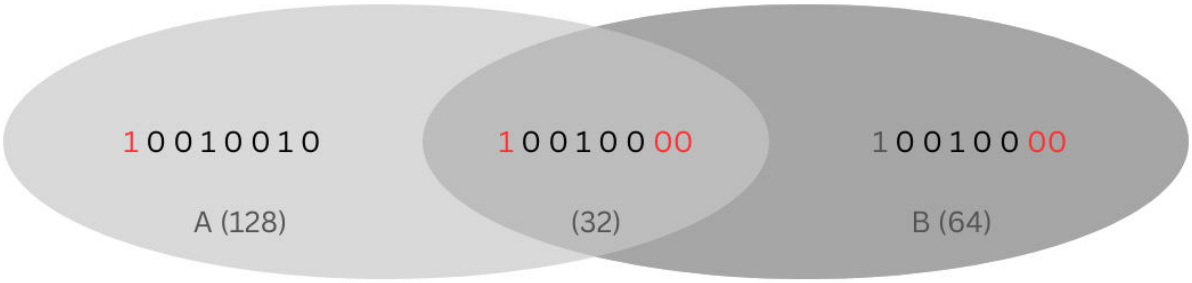
Wir verwenden dazu folgende Mengen:

- A = "Bitstrings der Länge 8, die mit 1 beginnen"
- B = "Bitstrings der Länge 8, die mit 00 enden"

Anzahl Möglichkeiten pro Stelle im Bitstring:

$$\bar{1} \cdot \bar{2} \cdot \bar{2} \cdot \bar{2} \cdot \bar{2} \cdot \bar{2} \cdot \bar{1} \cdot \bar{1} = 2^5 = |A \cap B|$$
$$\bar{1} \cdot \bar{2} \cdot \bar{2} \cdot \bar{2} \cdot \bar{2} \cdot \bar{2} \cdot \bar{2} \cdot \bar{2} = 2^7 = |A|$$
$$\bar{2} \cdot \bar{2} \cdot \bar{2} \cdot \bar{2} \cdot \bar{2} \cdot \bar{2} \cdot \bar{1} \cdot \bar{1} = 2^6 = |B|$$

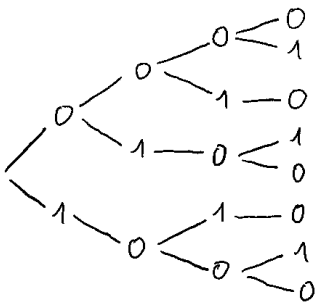
Wir wissen also, es gibt 2^7 Bitstrings, die mit einer Eins (1) beginnen. Das heisst 128 Bitstrings. Weiter gibt es $2^6 = 64$ Bitstrings, die mit zwei Nullen (00) enden. Auch weiss mann, dass es $2^5 = 32$ Bitstrings gibt, die mit einer Eins (1) starten und mit zwei Nullen (00) enden. Mit dem Ein/-Auschlussprinzip hat man $|A \cup B| = |A| + |B| - |A \cap B| = 128 + 64 - 32 = 160$ Bitstrings, die mit 1 beginnen **oder** mit 00 enden. A + B - **(A und B)**, weil "**oder**".



Baumdiagramme für Zählprobleme

Zählprobleme können oft mit Baumdiagrammen gelöst werden.

Wie viele Bitstrings der Länge 4 enthalten nicht zwei aufeinander folgende Einsen?



Wie man sieht hat man die folgenden Bitstrings:

- 0000
- 0001
- 0010
- 0101
- 0100
- 1010
- 1001
- 1000

⇒ Also **8** Bitstrings.

Das Schubfachprinzip (Pigeonhole Principle)

Wenn man $k+1$ Objekte auf k Schubfächer verteilen muss, dann gibt es wenigstens ein Schubfach mit mehr als einem Objekt

Falls man N Objekte auf k Schubfächer verteilt, dann gibt es wenigstens ein Schubfach, welches mindestens $\lceil N/k \rceil$ Objekte enthält.

An einem Beispiel: Unter 100 Personen, gibt es wenigstens $\lceil 100/12 \rceil = \lceil 8.33 \rceil = 9$, die im selben Monat Geburtstag haben.

Permutationen und Kombinationen

Permutationen

Eine Permutation von n verschiedenen Elementen ist eine geordnete Anordnung dieser n Elemente. Eine r-Permutation von n verschiedenen Elementen ist eine geordnete Anordnung von r der n Elemente.

Die geordnete Anordnung $(3,1,2)$ der Menge $S = \{1,2,3\}$ ist eine Permutation von S. Die geordnete Anordnung $(3,2)$ ist eine 2-Permutation von S.

3-Permutationen von S : $(1,2,3), (1,3,2), (2,1,3), (2,3,1), (3,1,2), (3,2,1)$

2-Permutationen von S : $(1,2), (1,3), (2,1), (2,3), (3,1), (3,2)$

Es gibt also $3 \cdot 2 \cdot 1 = 3! = 6$ 3-Permutationen und $3 \cdot 2 = 6$ 2-Permutationen von S

Anzahl Permutationen

Die Anzahl von r-Permutationen einer Menge von n Elementen ist:

$$P(n,r) = n(n-1)(n-2)\cdots(n-r+1) = \frac{n!}{(n-r)!}, \quad 0 \leq r \leq n \in \mathbb{N}$$

Anzahl n-Permutationen einer Menge von n Elementen ist $P(n,n) = n!$

Ein Beispiel

Auf wie viele Arten können die ersten drei Plätze bei einem Spiel mit 100 Teilnehmern ausgewählt werden?

Der erste Platz kann aus 100, der zweite aus 99 und der dritte noch aus 98 Teilnehmern ausgewählt werden. $P(100,3) = 100(100-1)(100-2) = 100 \cdot 99 \cdot 98 = \frac{100!}{(100-3)!} = 970'200$.

Kombinationen

Eine r-Kombination von n verschiedenen Elementen ist eine ungeordnete Auswahl von r dieser n Elemente. Sie ist also nichts anderes als eine Teilmenge mit r Elementen.

An einem einfachen Beispiel erklärt:

Für $S = \{1,2,3,4\}$ ist $\{1,3,4\}$ eine 3-Kombination von S. Dabei ist zu beachten, dass $\{3,1,4\}$ die selbe Kombination ist, da in Kombinationen die Anordnung keine Rolle spielt.

Anzahl r-Kombinationen

Die Anzahl von r-Kombinationen einer Menge von $n \geq 0$ Elementen ist gegeben durch:

$$C(n,r) = \frac{n!}{r!(n-r)!} = \binom{n}{r} = C(n,n-r), \text{ wobei } 0 \leq r \leq n \text{ gelten muss.}$$

Ein Beispiel

Wie viele Teams von 3 Personen lassen sich aus einer Gruppe von 5 Personen bilden?

Wie wir merken, ist hier die Reihenfolge der Mitglieder egal. Also handelt es sich hier um die 3-Kombination der Menge (5 Personen), die wir finden müssen: $C(5,3) = \frac{5!}{3! \cdot 2!} = 10$

Unterschied Permutation und Kombination

Wir sehen also, dass der Unterschied folgender ist:

- Eine Permutation ist eine Anordnung von Objekten in einer bestimmten Reihenfolge.
- Zum Beispiel: Die Anordnung der Zahlen [1,2,3] ist eine andere Permutation als [3,2,1], obwohl beide die gleichen Zahlen enthalten.

Auf der anderen Seite:

- Eine Kombination ist eine Auswahl von Objekten, bei der die Reihenfolge nicht berücksichtigt wird.
- Zum Beispiel: Die Auswahl der Zahlen [1,2,3] ist dieselbe Kombination wie [3,2,1].

Zusammenfassung Permutationen / Kombinationen

r-Permutationen von n Elementen ohne Wiederholung

Reihenfolge ist relevant.

$$\frac{n!}{(n-r)!}$$

r-Kombinationen von n Elementen ohne Wiederholung

*Reihenfolge ist **nicht** relevant.*

$$\frac{n!}{r!(n-r)!} = \binom{n}{r} = \binom{n}{n-r}$$

r-Permutationen von n Objekten mit Wiederholung

Reihenfolge ist relevant

$$n^r$$

r-Kombinationen von n Objekten mit Wiederholung

*Reihenfolge ist **nicht** relevant*

$$\frac{(n+r-1)!}{r!(n-1)!} = \binom{n+r-1}{r}$$

Binomialkoeffizient

Der Binomialkoeffizient ist folgendermassen definiert:

$$\binom{\alpha}{k} = \begin{cases} \frac{\alpha \cdot (\alpha-1) \cdots (\alpha-k+1)}{k!} & k > 0 \\ 1, & k = 0 \end{cases}$$

Für den Spezialfall, wo $\alpha = n \in \mathbb{N}_0$, also auch α eine natürliche Zahl inklusive Null ist, ist dies die Definition:

$$\binom{n}{k} = \begin{cases} \frac{n!}{k!(n-k)!} & k > 0 \\ 1, & k = 0 \end{cases}$$

Es gilt folgende Symmetrie der Binomialkoeffizienten:

$$\binom{n}{r} = C(n,r) = C(n,n-r) = \binom{n}{n-r}$$

Also gilt auch

$$\frac{n!}{(n-r)!(n-(n-r))!} = \frac{n!}{(n-4)!r!}$$

Beispiel:

$$\binom{10}{8} = \binom{10}{10-8} = \binom{10}{2} = \binom{10 \cdot 9}{1 \cdot 2} = 45 \quad \rightarrow \overline{8}$$

Das Pascalsche Dreieck und die Binomialkoeffizienten

$$\begin{array}{ccccccc} & & & n=0 & \binom{0}{0} & & \\ & & & n=1 & \binom{1}{0} & \binom{1}{1} & \\ & & & n=2 & \binom{2}{0} & \binom{2}{1} & \binom{2}{2} \\ & & n=3 & \binom{3}{0} & \binom{3}{1} & \binom{3}{2} & \binom{3}{3} \\ & & & r=0 & r=1 & r=2 & r=3 \end{array}$$

- Symmetrisch zur Mittelsenkrechten
- Aussen immer $= 1$
- Inneres Element gleich Summe der links und rechts darüber liegenden Elemente

Das Pascalsche Dreieck ist symmetrisch, denn für $n \in \mathbb{N}$ und $0 \leq k \leq n$ gilt:

$$\binom{n}{k} = \binom{n}{n-k}$$

Binomische Lehrsatz

Für $x, y \in \mathbb{R}$ (sogar in $\mathbb{C}$) und $n \in \mathbb{N}$ gilt:

$$\begin{aligned}(x+y)^n &= \sum_{k=0}^{n} \binom{n}{k} x^{n-k} y^k \\ &= \binom{n}{0} x^n y^0 + \binom{n}{1} x^{n-1} y^1 + \ldots + \binom{n}{n-1} x^1 y^{n-1} + \binom{n}{n} x^0 y^n\end{aligned}$$

Beispiel: Der Koeffizient von x^{13} in $(2x+y)^{51}$: $51 - k = 13, \quad k = 38$

$$(2x+y)^{51} = \binom{51}{0}(2x)^{51} + \binom{51}{1}(2x)^{50}y^1 + \ldots + \binom{51}{38}(2x)^{13}y^{38} + \ldots$$

$$\text{Also: } \underbrace{\binom{51}{38}2^{13}}_{\text{Koeffizient}} x^{13}y^{13} \qquad \binom{51}{38} \cdot 2^{13} = \binom{51}{13} \cdot 2^{13}$$

Folgerungen des binomischen Lehrsatz

Folgerung 1: Binomialtheorem für $(1+1)^n$

$$\sum_{k=0}^{n} \binom{n}{k} = 2^n = (1+1)^n = \sum_{k=0}^{n} \binom{n}{k} \cdot 1^{n-k} \cdot 1^k$$

Folgerung 2: Alternierende Summe der Binomialkoeffizienten

$$\sum_{k=0}^{n} (-1)^k \binom{n}{k} = 0 = (1+(-1))^n = \sum_{k=0}^{n} \binom{n}{k} \cdot 1^{n-k} \cdot (-1)^k$$

Folgerung 3: Binomialtheorem für $(1+2)^n$

$$\sum_{k=0}^{n} 2^k \binom{n}{k} = 3^n = (1+2)^n = \sum_{k=0}^{n} \binom{n}{k} \cdot 1^{n-k} \cdot 2^k$$

Folgerung 4: Die Vandermonde Folgerung

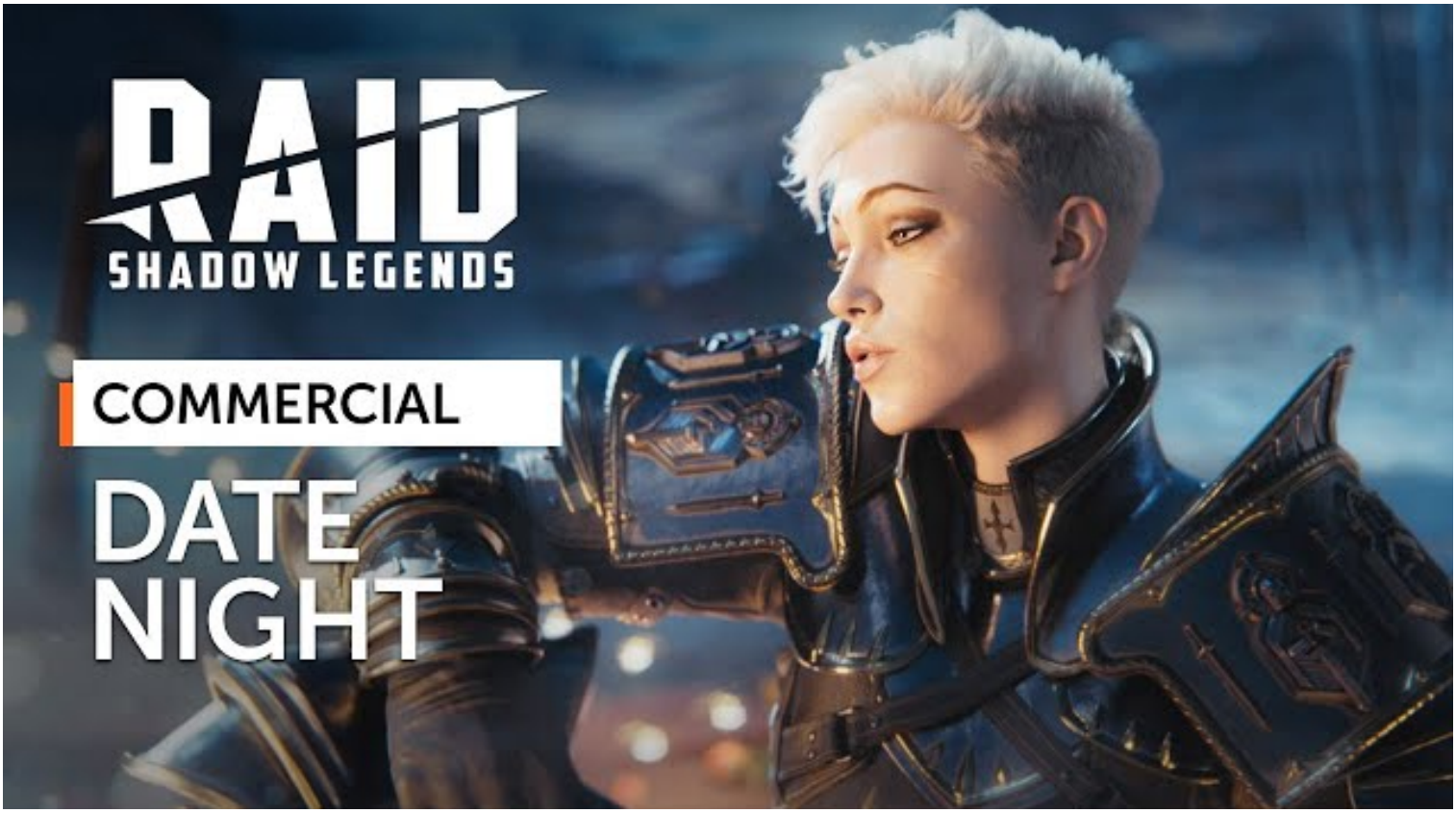
Nützlich bei der Lösung von Problemen mit überlappenden Mengen

$$\text{Für } n, m, k \in \mathbb{N} \text{ gilt} \qquad \binom{m+n}{k} = \sum_{j=0}^{k} \binom{m}{j} \binom{n}{k-j}$$

Short word from our sponsor

Today's summary is sponsored by Raid Shadow Legends,
one of the biggest mobile role-playing games of 2019 and it's totally free!
Currently almost 10 million users have joined Raid over the last six months,
and it's one of the most impressive games in its class with detailed models, environments and smooth 60 frames per second animations! All the champions in the game can be customized with unique gear that changes your strategic buffs and abilities! The dungeon bosses have some ridiculous skills of their own and figuring out the perfect party and strategy to overtake them's a lot of fun! Currently with over 300,000 reviews, Raid has almost a perfect score on the Play Store! The community is growing fast and the highly anticipated new faction wars feature is now live, you might even find my squad out there in the arena!
It's easier to start now than ever with rates program for new players you get a new daily login reward for the first 90 days that you play in the game! So what are you waiting for?

Good luck and I'll see you there!



Warscheinlichkeitstheorie

Definition (Wahrscheinlichkeit)

Jedem Ereignis $A \subset \Omega$ (Ω : Stichprobenraum) wird eine reelle Zahl $p(A) \in [0,1]$ zugeordnet. Sie entspricht der Wahrscheinlichkeit, dass das Ereignis A eintritt. Formal ist die Wahrscheinlichkeit eine Abbildung:

$$p : \underbrace{2^{\Omega}}_{P(\Omega)} \to [0,1], A \mapsto p(A)$$

mit den drei Eigenschaften (**Axiome** von Kolmogorov):

$$\begin{array}{ll} (1) & \forall A(0 \leq p(A) \leq 1), \\ (2) & p(\Omega) = 1, \\ (3) & \forall A \forall B(A \cap B = \emptyset \to p(A \cup B) = p(A) + p(B)) \end{array}$$

Gleichverteilung — Laplaceverteilung

Die Laplaceverteilung hat man dann, wenn alle möglichen Ereignisse eines Versuchs die selbe Eintrittswarscheinlichkeit haben.

$$p(x_i) = \frac{1}{|\Omega|} = \frac{1}{n}, \quad i = 1, 2, \ldots, n$$

Die Wahrscheinlichkeit eines Ereignisses A des endlichen Stichprobenraumes Ω von lauter gleich wahrscheinlichen Ereignissen ist gegeben durch:

$$p(A) = \frac{|A|}{|\Omega|} = \frac{\text{Anzahl günstige Fälle}}{\text{Anzahl mögliche Fälle}}$$

Zum Beispiel ist die Wahrscheinlichkeit bei einem Münzwurf Zahl zu erhalten $(A = 1, \Omega = 2)$ $p(A) = \frac{1}{2}$. Bei der Münze ist die Eintrittswahrscheinlichkeit von allen Elementarereignissen gleich gross. Somit handelt es sich um eine Laplaceverteilung.

Wichtige Regeln

Wahrscheinlichkeit des komplementären Ereignis

Falls A ein Ereignis des Strichprobenraumes Ω ist, dann gilt für das komplementäre Ereignis (oder Gegenereignis).

$$\overline{A}: \quad p(\overline{A}) = 1 - p(A)$$

Additionsregel

Falls A_1 und A_2 zwei Ereignisse des Strichprobenraumes Ω sind, dann gilt die Additionsregel:

$$p(A \vee B) = p(A) + p(B)$$

Wenn man dies mit überschneidenden Mengen hat. (Also, wenn A_1 oder A_2 eintreten soll, aber nicht beides gleichzeitig):

$$p(A_1 \cup A_2) = p(A_1) + p(A_2) - p(A_1 \cap A_2)$$

Für drei überschneidende Mengen hat man:

$$p(A_1 \cup A_2) = p(A_1) + p(A_2) + p(A_3) - p(A_1 \cap A_2) - p(A_1 \cap A_3) - p(A_2 \cap A_3) + p(A_1 \cap A_2 \cap A_3)$$

Wahrscheinlichkeit eines beliebigen Ereignisses

Die Wahrscheinlichkeit des Ereignisses A ist gleich der Summe der Wahrscheinlichkeiten der Ergebnisse s in A:

$$p(A) = \sum_{s \in A} p(s)$$

Beispiel: Ein gezinkter Würfel hat die Eigenschaft, dass die Zahl 3 doppelt so häufig wie jede andere Zahl fällt. Mit welcher Wahrscheinlichkeit würfelt man mit diesem Würfe eine ungerade Zahl?

Sei p die Wahrscheinlichkeit, irgend eine Zahl, ausser 3 zu werfen. Dann hat man:

$$p = p(1) = p(2) = p(4) = p(5) = p(6) = \frac{1}{2}p(3)$$

Die Summe der Wahrscheinlichkeiten muss 1 geben:

$$1 = \sum_{k=1}^{6} p(k) = 5p + 2p = 7p \quad \text{daraus folgt } p = \frac{1}{7}$$

und somit $p(1) = p(2) = p(4) = p(5) = p(6) = \frac{1}{7}$ und $p(3) = \frac{2}{7}$

Schliesslich hat man für die gesuchte Wahrscheinlichkeit:

$$p(\text{"ungerade Zahlen"}) = p(1) + p(3) + p(5) = \frac{1}{7} + \frac{2}{7} + \frac{1}{7} = \frac{4}{7} \approx 0.57$$

Bedingte Wahrscheinlichkeit

Sind A und B zwei Ereignisse mit $p(B) > 0$, dann bezeichnet man die Wahrscheinlichkeit von A unter der Voraussetzung B mit $p(A|B)$. *Beachte dass* $p(A|\Omega) = p(A)$.

Theorem:

Die Wahrscheinlichkeit für das Ereignis A unter der Voraussetzung, dass Ereignis B eingetreten ist, ist gegeben durch:

$$p(A|B) = \frac{p(A \cap B)}{p(B)}$$

Beispiel:

Beim dreimaligen Werfen einer fairen Münze wissen wir, dass beim ersten Mal Zahl geworfen wurde. Wie gross ist dann die Wahrscheinlichkeit, dass eine ungerade Anzahl Zahlen geworfen wurden?

$$\Omega = \{\,\mathrm{ZZZ}, \mathrm{ZZK}, \mathrm{ZKZ}, \mathrm{ZKK}, \mathrm{KZZ}, \mathrm{KZK}, \mathrm{KKZ}, \mathrm{KKK}\,\},$$
$$A = \{\,\mathrm{ZZZ}, \mathrm{ZKK}, \mathrm{KZK}, \mathrm{KKZ}\,\}, \quad B = \{\,\mathrm{ZZZ}, \mathrm{ZZK}, \mathrm{ZKZ}, \mathrm{ZKK}\,\}$$

Also hat man mit der obigen Formel:

$$p(A|B) = \frac{|A \cap B|}{|B|} = \frac{\frac{|A \cap B|}{|\Omega|}}{\frac{|B|}{|\Omega|}} = \frac{p(A \cap B)}{p(B)} = \frac{2/8}{4/8} = \frac{2}{4} = 0.5$$

Unabhängige Ereignisse

Man nennt zwei Ereignisse A und B genau dann unabhängig, falls $p(A \cap B) = p(A)p(B)$.

Also zwei Ereignisse A und B sind unabhängig, wenn die Warscheinlichkeit ihres gemeinsamen Auftretens $p(A \cap B)$ gleich dem Produkt ihrer einzelnen Wahrscheinlichkeiten $p(A) \cdot p(B)$ ist.

Folgende Gleichungen gelten für unabhängige Ereignisse:

$p(A \cap B) = p(A) \cdot p(B)$	Dies ist die Grunddefinition der Unabhängigkeit
$p(A\|B) = p(A)$	Dies bedeutet, dass die Wahrscheinlichkeit von A, gegeben B, gleich der Wahrscheinlichkeit von A allein ist. Das Existieren von B hat keinen Einfluss auf das Auftreten von A.
$p(B\|A) = p(B)$	Dies bedeutet, dass die Wahrscheinlichkeit von B, gegeben A, gleich der Wahrscheinlichkeit von B allein ist. Das Existieren von A hat keinen Einfluss auf das Auftreten von B.

Beim Prüfen der Unabhängigkeit von A und B gehen wir wie im folgenden Beispiel vor:

Ausgangslage

Ein Bitstring der Länge 4 wird zufällig erzeugt. Wir betrachten die beiden Ereignisse

- A = "String enthält mindestens zwei aufeinanderfolgende 0-en",
- B = "erstes Bit ist 0"

Bestimmen von $p(A|B)$ mit $|\Omega| = 2^4 = 16$ und

- $A = \{\, 0000, 0001, 0010, 0011, 0100, 1000, 1001, 1100 \,\}, \qquad |A| = 8$
- $B = \{\, 0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111 \,\} \qquad |B| = 8$

Unabhängigkeit beweisen (mithilfe der Gleichungen)

$$p(B) = \frac{|B|}{|\Omega|} = \frac{8}{16} = \frac{1}{2}, \qquad p(A \cap B) = \frac{|A \cap B|}{|\Omega|} = \frac{5}{16}$$
$$p(A|B) = \frac{p(A \cap B)}{p(B)} = \frac{5/16}{1/2} = \frac{5}{8} \neq p(A) = \frac{|A|}{|\Omega|} = \frac{8}{16} = \frac{1}{2}$$

Wir haben hier also $P(A|B) \neq P(A)$, somit sind A und B abhängig.

Totale Wahrscheinlichkeit und Satz von Bayes

Satz von der totalen Wahrscheinlichkeit

Angenommen, die Ereignisse $B_1, B_2, \ldots, B_n$ sind disjunkt und bilden eine vollständige Partition des Ergebnisraums, d.h., eines dieser B_i muss eintreten und keines der Ereignisse kann gleichzeitig eintreten. Dann kann die Wahrscheinlichkeit eines Ereignisses A ausgedrückt werden als:

$$p(A) = p(A \cap B_1) + p(A \cap B_2) + \cdots + p(A \cap B_n)$$

$$\text{oder auch: } \sum_{i=1}^{k} p(A \cap B_i)$$

Da die Ereignisse B_i disjunkt sind, kann dies auch ausgedrückt werden als:

$$p(A) = p(A|B_1)p(B_1) + p(A|B_2)p(B_2) + \cdots + p(A|B_n)p(B_n)$$

$$\text{oder auch: } \sum_{i=1}^{k} p(A|B_i) \cdot p(B_i)$$

Hierbei ist

- $p(A|B_i)$ die bedingte Wahrscheinlichkeit, dass Ereignis A eintritt, gegeben dass Ereignis B_i eingetreten ist.
- $p(B_i)$ die Wahrscheinlichkeit, dass Ereignis B_i eintritt.

Unter einer ***weiteren Voraussetzung C***, gilt:

$$p(A|C) = \frac{1}{p(C)} \left(p(A \cap B_1 \cap C) + p(A \cap B_2 \cap C) + \ldots + p(A \cap B_k \cap C)\right) =$$
$$p(A|B_1 \cap C) \cdot p(B_1|C) + p(A|B_2 \cap C) \cdot p(B_2|C) + \ldots + p(A|B_k \cap C) \cdot p(B_k|C)$$
$$\text{und somit:}$$
$$p(A|C) = \frac{1}{p(C)} \sum_{i=1}^{k} p(A \cap B_i \cap C) = \sum_{i=1}^{k} p(A|B_i \cap C) \cdot p(B_i|C)$$

Beispiel: Angenommen, eine Schule hat drei Klassen mit unterschiedlichen Anteilen an Schülern. Wenn wir die Wahrscheinlichkeit berechnen sollten, dass ein zufällig ausgewählter Schüler eine Brille trägt, können wir den Satz von der totalen Wahrscheinlichkeit anwenden, indem wir die Wahrscheinlichkeit, dass ein Schüler aus jeder Klasse eine Brille trägt, mit der Wahrscheinlichkeit gewichten, dass der Schüler aus dieser spezifischen Klasse kommt. Wenn **Klasse 1** einen Brille tragenden Anteil von 30%**, Klasse 2 von** 20%**, Klasse 3 von** 10% hat, und die Klassen 50%, 30%, 20% der Schülerschaft ausmachen, dann wäre die Gesamtwahrscheinlichkeit, dass ein zufällig ausgewählter Schüler eine Brille trägt:

$$p(\text{Brille}) = p(\text{Brille}|\text{Klasse 1}) \cdot p(\text{Klasse 1}) + p(\text{Brille}|\text{Klasse 2}) \cdot p(\text{Klasse 2}) + p(\text{Brille}|\text{Klasse 3}) \cdot p(\text{Klasse 3})$$

$$p(Brille) = 0.30 \cdot 0.50 + 0.20 \cdot 0.30 + 0.10 \cdot 0.20 = 0.23 = 23\%$$

Der Satz von Bayes

Der Satz von Bayes ermöglichst es uns, bedingte Wahrscheinlichkeiten anzupassen, wenn neue Informationen verfügbar werden. Er kann folgendermassen formuliert werden:

$$P(B|A) = \frac{P(A|B) \cdot P(B)}{P(A)}$$

Hierbei ist:

- $P(B|A)$ die Wahrscheinlichkeit von Ereignis B, gegeben das Ereignis A ist eingetreten.
- $P(A|B)$ die Wahrscheinlichkeit von Ereignis A, gegeben das Ereignis B ist eingetreten.
- $P(B)$ die Wahrscheinlichkeit von Ereignis B, bevor wir zusätzliche Informationen über das Eintreten von A haben
- $P(A)$ die Wahrscheinlichkeit von Ereignis A, die durch den Satz der totalen Wahrscheinlichkeit berechnet werden kann.

Beispiel: Wir haben eine Schule mit zwei Klassenräumen, und wir wissen, dass in einem der Klassenräume die Hälfte der Schüler blaue T-Shirts trägt, während im anderen Klassenraum nur ein Viertel der Schüler blaue T-Shirts trägt. Nehmen wir weiter an, dass die b**eiden Klassen gleich gross sind**. Wenn wir nun einen Schüler sehen, der das Schulgelände mit einem blauen T-Shirt verlässt, wie hoch ist die Wahrscheinlichkeit, dass der Schüler aus dem ersten Klassenraum kommt?

Zuerst definieren wir,

- dass B das Ereignis ist, dass ein Schüler aus dem ersten Klassenraum kommt.
- dass A das Ereignis ist, dass ein Schüler ein blaues T-Shirt trägt.

Gegeben:

- $P(B) = 0.5$, weil es zwei Klassenräume gibt und wir keine weiteren Informationen zur Klassengrösse haben
- $P(A|B) = 0.5$ (die Hälfte der Schüler im ersten Klassenraum trägt blaue T-Shirts).
- $P(A) = 0.5 \cdot 0.5 + 0.5 \cdot 0.25 = 0.375$ (die totale Wahrscheinlichkeit, dass ein Schüler ein blaues T-Shirt trägt, berechnet durch den Satz der totalen Wahrscheinlichkeit)

Nun können wir den Satz von Bayes anwenden, um $P(B|A)$ zu berechnen:

$$P(B|A) = \frac{P(A|B) \cdot P(B)}{P(A)} = \frac{0.5 \cdot 0.5}{0.375} = \frac{0.25}{0.375} = \frac{2}{3} \approx 66.67\%$$

Verteilungsfunktionen

Die Bernoulliverteilung

Die Bernoulliverteilung beschreibt die Binomialverteilung, wobei k nur die Werte $\{ 0, 1 \}$ annehmen kann. Mehr zur Binomialverteilung im nächsten Kapitel.

Binomialverteilung

Verwendet zur Berechnung kombinierter, unabhängiger Wahrscheinlichkeiten

Die Binomialverteilung bietet ein leistungsstarkes Werkzeug zur Berechnung der Wahrscheinlichkeiten von Kombinationen von Ereignissen bei einer festgelegten Anzahl von unabhängigen Versuchen. Die zugehörige Formel lautet:

$$P(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$$

Hierbei ist:

- $P(X = k)$ die Wahrscheinlichkeit, genau *k* Mal Erfolg in *n* unabhängigen Versuchen zu haben.
- $\binom{n}{k}$ der Binomialkoeffizient, der die Anzahl der Wege angibt, *k* Erfolge in *n* Versuchen anzuordnen.
- p die Wahrscheinlichkeit des Erfolgs bei einem einzelnen Versuch.
- $1-p$ die Wahrscheinlichkeit des Misserfolgs bei einem einzelnen Versuch.

Beispiel: Wie hoch ist die Wahrscheinlichkeit, dass 3 von 5 Kindern Jungen sind, wenn die Wahrscheinlichkeit, dass man einen Jungen kriegt $p(J) = 0.51$?

- $P(X = k)$ ist hier $P(3J)$
- $\binom{n}{k}$ ist hier $\binom{5}{3} = 10$, da n = 5 Versuche und k = 3 Erfolge
- $p = 0.51$ (Wahrscheinlichkeit für Erfolg : k)
- und $1 - p = 0.49$

Dies setzen wir in die Formel ein:

$$P(3J) = \binom{5}{3} 0.51^3 0.49^2 = 0.31849 \approx 31.85\%$$

Hypergeometrische Verteilung

Verwendet um Wahrscheinlichkeiten einer Stichprobe zu berechnen (ohne Zurücklegen)

$$p(k) = \frac{\binom{M}{k}\binom{N-M}{n-k}}{\binom{N}{n}}, \quad (k = 0, 1, 2, \ldots, n)$$

Wobei:

- N : Strichprobenraum (Anz. Objekte) (z.B. Wie viele Kugeln sind in Urne)
- M : Anz. Elemente im Stichprobenraum, welche eine Bedingung erfüllen $M(0 \leq M \leq N)$ (z.B. Wie viele rote Kugeln sind in Urne)
- n : Anzahl der Strichproben (z.B. Wie oft wird aus Urne gezogen)
- k : Anzahl der Erfolge (z.B. Bei wie vielen Zügen will man Erfolg (Kugel = rot) haben)

Beispiel: Eine Schachtel enthält 37 Teile. Von denen 5 defekt sind. Nun ziehen wir aus der Schachtel, ohne Zurücklegen 3 Teile. Wie gross ist die Wahrscheinlichkeit, dass von diesen 3 Teilen exakt 3 Teile defekt sind (also alle gezogenen defekt sind)?

$$p(k=3) = \frac{\binom{5}{3}\binom{37-5}{3-3}}{\binom{37}{3}} = 0.00128 \approx 0.13\%$$

Die Poissonverteilung

Verwendet zur Berechnung Wahrscheinlichkeiten von seltenen Ereignissen in einem festen Zeit- oder Raumintervall.

$$P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!} = \frac{\lambda^k}{k!}e^{-\lambda}, \qquad \lambda = n \cdot p$$

Wobei:

- λ : Der Erwartungswert der Verteilung, berechnet als $\lambda = n \cdot p$
  - n : die Anzahl der beobachteten Einheiten (z.B. 5000 Autos)
  - p : die Wahrscheinlichkeit des Ereignisses (W'keit das ein Auto rot $p = 0.01$)
- k : die Anzahl der Erfolge, die man berechnen will
- e : (Konstante) — die Basis des natürlichen Logarithmus

Beispiel: Die Wahrscheinlichkeit bei Grippeimpfung an Nebenwirkungen zu erkranken sei $p = 0.001$. Nun werden $n = 2000$ Personen geimpft. Wie gross ist die W'keit, dass mehr als 2 Personen an diesen Nebenwirkungen erkranken?

Hier muss man einen Trick anwenden. Wir berechnen die W'keit dafür, dass keine Person erkrankt ist, die W'keit, dass eine Person erkrankt und die W'keit, dass zwei Personen erkranken. Diese W'keiten ziehen wir von 1 (100%) ab:

$$\lambda = 2000 \cdot 0.001 = 2$$
$$p(>2) = 1 - p(0) - p(1) - p(2) = 1 - \frac{2^0 e^{-2}}{0!} - \frac{2^1 e^{-2}}{1!} - \frac{2^2 e^{-2}}{2!} = 0.3233$$

Somit ist die W'keit, dass mehr als 2 Personen an dieser Grippeimpfung erkranken bei 32.23%. Die W'keit dass mehr als 3 Personen erkranken, wäre schon bei $0.14287 = 14.29\%$. Man sieht hier gut, wie sich die Poissonverteilung verhält.

Zufallsvariablen, Wahrscheinlichkeitsverteilung

Eine Zufallsvariable X ist eine Abbildung vom Strichprobenraum Ω in die Menge der reellen Zahlen $\mathbb{R}$, wobei jedem Ergebnis $r \in S$ eine reelle Zahl $X(r)$ zugeordnet wird.

Die Wahrscheinlichkeitsverteilung einer Zufallsvariablen X auf einem Stichprobenraum S ist die Menge

$$\{\,(r, p(X=r)) \mid \forall r \in X(S)\,\}$$

wobei $p(X=r)$ die Wahrscheinlichkeit dafür ist, dass die Zufallsvariable X den Wert r annimmt. Die Wahrscheinlichkeitsverteilung von X wird üblicherweise dadurch spezifiziert, dass man $p(X=r), \forall r \in X(S)$ tabelliert.

Beispiel: Bestimmen der **Wahrscheinlichkeitsverteilung** der Zufallsvariable $X =$ "Augensumme beim Wurf zweier fairer Würfel"

r	2	3	4	5	6	7	8	9	10	11	12
P(X=r)	1/36	2/36	3/36	4/36	5/36	6/36	5/36	4/36	3/36	2/36	1/36

Erwartungswert einer Zufallsvariablen

Der Erwartungswert einer Zufallsvariablen ist der Wert der darüber aussagt, wie viele Versuche benötigt werden, um ein gewisses Ergebnis zu erhalten. Definiert durch:

$$E[X] = \sum_{s \in S} X(s) \cdot p(s) = \sum_{r \in X(S)} r \cdot p(X=r)$$

Es gelten folgende Regeln beim Rechnen mit Erwartungswerten:

$$\begin{array}{ll} (i) & E(X_1 + X_2 + \cdots + X_n) = E(X_1) + E(X_2) + \cdots + E(X_n) \\ (ii) & E(aX + b) = aE(X) + b \end{array}$$

Beispiel: Eine faire Münze wird so oft geworfen, bis ein Kopf (K), oder fünfmal Zahl (Z) geworfen wurde. Berechne die erwartete Anzahl X von Würfen (also $r = \text{Würfe} \Rightarrow X(r)$).

Zuerst muss man sich überlegen: Was sind die möglichen Ergebnisse?

$$\underbrace{K}_{1}, \underbrace{ZK}_{2}, \underbrace{ZZK}_{3}, \underbrace{ZZZK}_{4}, \underbrace{ZZZZK}_{5}, \underbrace{ZZZZZ}_{5}$$

Wir sehen sofort, dass wir für $(X = r, r = 5)$ zwei Ausgänge haben. Nur einer dieser kann geschehen, somit müssen wir später deren Wahrscheinlichkeiten addieren. Nun berechnet man für diese die Wahrscheinlichkeiten

$$\underbrace{\frac{1}{2}}_{r=1}, \quad \underbrace{(\frac{1}{2})^2}_{r=2}, \quad \underbrace{(\frac{1}{2})^3}_{r=3}, \quad \underbrace{(\frac{1}{2})^4}_{r=4}, \quad \underbrace{(\frac{1}{2})^5}_{r=5}, \quad \underbrace{(\frac{1}{2})^5}_{r=5}$$

Diese Wahrscheinlichkeiten verwenden wir in der Formel (wobei wir die Wahrscheinlichkeiten für $r = 5$ noch miteinander addieren müssen, da wir zwei Mögliche Ereignisse für dies haben, von welchen nur das eine **oder** das andere eintreten kann)

$$E(X) = 1 \cdot \frac{1}{2} + 2 \cdot \frac{1}{4} + 3 \cdot \frac{1}{8} + 4 \cdot \frac{1}{16} + 5(\frac{1}{16} + \frac{1}{16}) = \frac{31}{16} \approx 1.94$$

Durchschnittliche Komplexität

Die durchschnittliche Komplexität eines Algorithmus wird folgendermassen berechnet:

$$E(X) = \sum_{j=1}^{n} p(a_j)X(a_j)$$

wobei:

- $p(a_j)$ angibt, wie wahrscheinlich die Eingabe a_j ist.
- $X(a_j)$ angibt, wieviel Rechenzeit (oder Speicher) die Eingabe a_j erfordert.

Beispiel: Beim Roulette kann die Kugel am Ende bei Zahlen 1 bis 36, bei 0 oder bei 00 (in Amerika) stehen bleiben. Es sind verschiedene Wetten möglich. Wenn man auf Schwarz wettet und einen Dollar einsetzt. Wie hoch ist der erwartete Gewinn?

Mit einer W'keit von 18/38 gewinnt man einen Dollar, mit eine rW'keit von 20(38 verlieren wir den Dollar, d.h. man hat:

$$\text{Erwarteter Gewinn} = \frac{18}{38}(+1) + \frac{20}{38}(-1) = -\frac{1}{19} \approx -0.053$$

Varianz von Zufallsvariablen

Unabhängige Zufallsvariablen

Man nennt die beiden Zufallsvariablen X_1 und X_2 auf dem Strichprobenraum S unabhängig, falls:

$$\forall r_1 \in X_1(S) \forall r_2 \in X_2(S) \left[\underbrace{p(X_1 = r_1 \wedge X_2 = r_2)}_{\text{Verbundwahrscheinlichkeit}} = \underbrace{p(X_1 = r_1) \cdot p(X_2 = r_2)}_{\text{Produkt der Randwahrscheinlichkeit}} \right]$$

Varianz einer Zufallsvariablen

Die Varianz einer Zufallsvariablen X über dem Strichprobenraum S ist definiert durch

$$V(X) = \sum_{s \in S} (X(s) - E(X))^2 \cdot p(s) = \sum_{r \in X(S)} (r - E(X))^2 \cdot p(X = r)$$

$$\text{Ist } X \text{ eine Zufallsvariable. Dann gilt: } V(X) = E(X^2) - [E(X)]^2$$

Es gelten folgende Regeln beim Rechnen mit Varianzen:

$$V(X_1 + X_2 + \cdots + X_n) = V(X_1) + V(X_2) + \cdots + V(X_n)$$

Beispiel: Bestimmen der Varianz der Zufallsvariablen $X =$ "Augensumme beim Wurf zweier fairer Würfel".

Wir hatten $E(X) = 7$. Also:

$$V(X) = (2-7)^2 \frac{1}{36} + (3-7)^2 \frac{2}{36} + \cdots + (12-7)^2 \frac{1}{36} \approx 5.834 \quad \text{oder:}$$

$$V(X) = 2^2 \cdot \frac{1}{36} + 3^2 \cdot \frac{2}{36} \ldots 12^2 \frac{1}{36} - 7^2 \approx 5.834$$

Anwendung von Wahrscheinlichkeits-Konzepten

1. **Anzahl Möglichkeiten:** $\binom{n}{k}$
   Anzuwenden, wenn die Anzahl Möglichkeiten (Permutationen) berechnet werden sollen — aus n Versuchen, in denen k Versuche günstig ausfallen.
2. **Laplace-Wahrscheinlichkeit:** $P(E) = \frac{\text{Anz. günstiger Ereignisse}}{\text{Anz. möglicher Ereignisse}}$
   Anzuwenden, wenn alle Ergebnisse gleich wahrscheinlich sind, wie beim Würfeln.
3. **Additionstheorem:** $P(A \cap B) = P(A) + P(B) - P(A \cap B)$.
   Anzuwenden, um die Wahrscheinlichkeit zu finden, dass mindestens eines von zwei Ereignissen eintritt.
4. **Bedingte Wahrscheinlichkeit:** $P(A|B) = \frac{P(A \cap B)}{P(B)}$.
   Wenn ein Ereignis abhängig von einem anderen Ereignis ist (z.B. "Wenn Ereignis B passiert, wie wahrscheinlich ist dann Ereignis A?").
5. **Totale Wahrscheinlichkeit:** $P(A) = \sum P(A|B_i) \cdot P(B_i)$.

   Anzuwenden, um die Gesamtwahrscheinlichkeit eines Ereignisses zu berechnen, das über mehrere unterschiedliche Wege eintreten kann.
6. **Satz von Bayes:** $P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$.
   Anzuwenden, um Hypothesen zu testen oder Schlussfolgerungen auf Basis neuer Informationen zu ziehen.
7. **Bernoulli-Verteilung:** $P(X = k) = p^k(1-p)^1 - k \text{ für } k = \{0, 1\}$

   Anzuwenden für das Berechnen der Wahrscheinlichkeit eines Ereignisses, das nur zwei Ausgänge hat und nur einmal stattfindet. z.B. das Wefen einer Münze.
8. **Binomialverteilung:** $P(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$.
   Anzuwenden, wenn berechnet werden soll, wie Wahrscheinlich es ist, dass ein Ereignis mit zwei möglichen Ausgängen (wie Erfolg oder Miserfolg) in einer bestimmten Anzahl von unabhängigen Wiederholungen auftritt, wie z.B: das mehrfache Werfen einer Münze, wo n = Anzahl Versuche, k = Anzahl gewünschter Erfolge.
9. **Hypergeometrische Verteilung:** $P(X = k) = \frac{\binom{M}{k}\binom{N-M}{n-k}}{\binom{N}{n}}$
   Für Situationen, in denen aus einer begrenzten Menge ohne Zurücklegen gezogen wird, z.B. die Wahrscheinlichkeit, bestimmte Karten in einem Kartenspiel zu ziehen.
10. **Poissonverteilung**: $P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$
    Geeignet, um die Wahrscheinlichkeit von seltenen Ereignissen in einem festen Zeit- oder Raumintervall zu berechnen, wie z.B. die Anzahl der Autos, die in einer Stunde über eine Brücke fahren. z.B. konkret: Wenn 1% der Autos rot sind, wie wahrscheinlich dass von 5000 Autos genau 3 rot sind?
11. **Erwartungswert**: $E[X] = \sum xP(X = x)$
    Wenn der Durchschnittswert von Versuchen über einen längeren Zeitraum hinweg berechnet werden soll. z.B. die durchschnittliche Anzahl von Würfeln, die du werfen musst, um eine Sechs zu bekommen.
12. **Varianz**: $Var(X) = E[(X - E[X])^2)]$
    Wenn berechnet werden soll, wie stark die Ergebnisse einer Zufallsvarbiablen schwanken können, wie z.B. die Streuung der Ergebnisse bei mehrfachen Würfeln.

Rekursionsbeziehungen

Rekursionsbeziehungen werden gebraucht um rekursiv zu rechnen. Eine Rekursionsbeziehung der Folge $\{ a_n \}$ ist eine Beziehung der Form:

$$a_n = f(a_{n-1}, a_{n-2}, \dots, a_2, a_1), \qquad \forall n \geq n_0, n_0 \in \mathbb{N}^+$$

Eine Lösung dieser Rekursionsbeziehung ist eine Folge, welche diese Relation erfüllt.

Beispiel: Gegeben sei die Rekursionsbeziehung $a_n = 2an - 1 - a_{n-2}$ für alle $n = 2, 3, 4, \dots$. Sind nachfolgende Zahlenfolgen Lösungen dieser Rekursion?

$$\begin{array}{lll} (i) & a_n = 3n & 3n \stackrel{?}{=} 2 \cdot (3(n-1)) - (3(n-2)) = 6n - 6 - 3n + 6 \\ (ii) & a_n = 2^n & 2^n \stackrel{?}{=} 2 \cdot 2^{n-1} - 2^{n-2} \neq 2^n - 2^{n-1} \\ (iii) & a_n = 5 & 5 \stackrel{?}{=} 2 \cdot 5 - 5 = 5 \end{array}$$

Lösen von Rekursionsbeziehungen

Eigenschaften von Rekursionsbeziehungen

Die Rekursionsbeziehung gelte ab $n_0 \in \mathbb{N}^+, \forall n \geq n_0$:

$$F(a_n, a_{n-1}, \dots, a_2, a_1) = r(n)$$

- Sie ist **linear**, falls F eine lineare Funktion ihrer Variablen $a_n, a_{n-1}, a_{n-2}, \dots, a_2, a_1$ ist. Typischer Weise ist F von der Form:

$$F(a_n, a_{n-1}, \dots, a_2, a_1) = a_n - c_1 a_{n-1} - c_2 a_{n-2} - \dots - c_k a_{n-k}$$

- Sie ist **homogen**, falls die rechte Seite verschwindet, das heisst falls $r(n) = 0, \forall n$.
- Sie ist **k-ter Ordnung**, falls F höchstens von Gliedern ab a_{n-k} abhängt, aber keinesfalls von früheren Gliedern.

Lineare Rekursionsbeziehungen

Eine lineare, homogene Rekursionsbeziehung k-ter Ordnung mit konstanten Koeffizienten ist eine Beziehung der Form.

$$a_n - c_1 a_{n-1} - c_2 a_{n-2} - \dots - c_k a_{n-k} = 0$$

Wobei für die Koeffizienten gilt: $c_1, c_2 \dots, c_k \in \mathbb{R}$ und $c_k \neq 0$

Die **allgemeine Lösung** einer **linearen, inhomogenen Rekursionsbeziehung** ist die Summe der allgemeinen Lösung der zugehörigen homogenen RB und der partikulären Lösung der inhomogenen Rekursionsbeziehung.

Theoretisches Beispiel:
Bestimme die allgemeine Lösung
$\{ a_n^{(h)} \}$ der zugehörigen homogenen Rekursionsbeziehungen (RHS Null setzen):

$$a_n - c_1 a_{n-1} - c_2 a_{n-2} - \dots - c_k a_{n-k} = 0$$

Bestimme eine (einzige) partikuläre Lösung $\{ a_n^{(p)} \}$ der inhomogenen Rekursionsbeziehungen:

$$a_n - c_1 a_{n-1} - c_2 a_{n-2} - \dots - c_k a_{n-k} = r(n)$$

Dann ist die allgemeine Lösung der inhomogenen Rekursionsbeziehungen die Summe dieser beiden Lösungen:

$$\{ a_n \} = \{ a_n^{(h)} \} + \{ a_n^{(p)} \}$$

Praktisches Beispiel: Setzt man in die lineare, homogene Rekursionsbeziehung

$a_n - a_{n-1} - a_{n-2} = 0$ den Ansatz $a_n = r^n$ ein, erhält man:

$$r^n - r^{n-1} - r^{n-2} = 0 \Leftrightarrow r^{n-2}(r^2 - r - 1) = 0$$

Nach Division durch $r^{n-2} \neq 0$ erhält man die charakteristische Gleichung

$$r^2 - r - 1 = 0$$

Die Lösungen dieser quadratischen Gleichung sind:

$$r_{1,2} = \frac{-(-1) \pm \sqrt{(-1)^2 - 4 \cdot 1 \cdot (-1)}}{2 \cdot 1}$$
$$r_1 = \frac{1}{2}(1 + \sqrt{5}), \qquad r_2 = \frac{1}{2}(1 - \sqrt{5})$$

Erzeugende Funktion

Erzeugende Funktionen erstellen Folgen. Die erzeugende Funktion der Folge $a_0, a_1, \ldots, a_k, \ldots (a_k \in \mathbb{R})$ ist die unendliche Reihe:

$$G(x) = a_0 x^0 + a_1 x^1 + a_2 x^2 + a_3 x^3 + \cdots = \sum_{k=0}^{\infty} a_k x^k$$

Beispiele für erzeugende Funktionen:

- $3, 3, 3, \ldots \quad \rightsquigarrow G(x) = 3x^0 + 3x^1 + 3x^2 + 3x^3 + \cdots = \sum_{k=0}^{\infty} 3x^k$
- $1, 2, 3, \ldots \quad \rightsquigarrow G(x) = 1x^0 + 2x^1 + 3x^2 + \cdots = \sum_{k=0}^{\infty} (k+1)x^k$
- $1, 1, 1, 1, 1 \quad \rightsquigarrow G(x) = 1x^0 + 1x^1 + 1x^2 + 1x^3 + 1x^4 = \sum_{k=0}^{4} x^k$

Erweitertes Ein-/Auschlussprinzip und Anwendungen

Für beliebige Mengen A und B gilt

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

Beispiel: Wieviele positive natürliche Zahlen nicht grösser als 1000 sind durch 7 oder 11 teilbar?

$$\lfloor \frac{1000}{7} \rfloor + \lfloor \frac{1000}{11} \rfloor - \lfloor \frac{1000}{77} \rfloor = 142 + 90 - 12 = 220$$

Also $\text{Anz. } /7 + \text{Anz. } /11$ - $\text{Anz. } (/7 \ \& \ /11)$.

Bei mehr als zwei Mengen sieht der Prozess wie folgt aus:

$$|A_1 \cup A_2 \cup A_3| = |A_1| + |A_2| + |A_3| - |A_1 \cap A_2| - |A_1 \cap A_3||A_2 \cap A_3| + |A_1 \cap A_2 \cap A_3|$$

Derangement

Ein Derangement ist eine Permutation, die kein Objekt am selben Platz lässt. Die Permutation 21453 ist ein Derangement von 12345. Die Anzahl Derangements bei einer Menge von n Elementen ist:

$$D_n = n![1 - \frac{1}{1!} + \frac{1}{2!} - \frac{1}{3!} + \cdots + (-1)^n \frac{1}{n!}]$$

Alternative Form: Einschluss-/Auschlussprinzip

Wenn man mit vielen Mengen arbeitet, wir die herkömmliche Form des Ein-/Auschlussprinzips recht mühsam. Dann kann man aber diese Form verwenden:

$$|N - P_1 \cup P_2 \cup P_3 \cup \cdots \cup P_n|$$

Wobei

- N : Die Gesamtmenge (Universalmenge) aller Möglichen Werte
- P_i : Die Teilmengen im Universum

Beispiel: Auf wie viele Arten kann man 5 Jobs 4 verschiedenen Angestellten zuordnen, wenn jeder mind. einen Job kriegen soll?

$$\_ \_ \_ \_ \_ = J_1 \cdot J_2 \cdot J_3 \cdot J_4 \cdot J_5 = 4 \cdot 4 \cdot 4 \cdot 4 \cdot 4 = 4^5 = 1024, \text{ ohne Bedingung}$$

Dies ist die Universalmenge, die Menge aller Möglichkeiten, ihne irgendwelche Bedingungen. P_i : Person i hat keinen Job.

$$\begin{aligned} & 4^5 - |P_1 \cup P_2 \cup P_3 \cup P_4| \\ & = 4^5[\binom{4}{1} \cdot 3^5 - \binom{4}{2} \cdot 2^5 + \binom{4}{3} 1^5 - \binom{4}{4} \cdot 0] \\ & = 1024 - 972 + 192 - 4 + 0 \\ & = 240 \end{aligned}$$

Divison mit Rest und Kongruenz modulo n

Für beliebige $a, n \in \mathbb{Z}$mit $n \neq 0$ existieren eindeutig bestimmte Zahlen $q \in \mathbb{Z}$ (genannt Quotient) und $r \in \mathbb{Z}$ (genannt Rest) mit

$$a = q \cdot n + r \qquad \text{und} \qquad 0 \leq r < |n|$$

Man nennt a den Dividend und n den Divisor. Oft schreibt man auch $r = R_n(a) = a \bmod n$ (Rest von a bei Division durch n). Also z.B. $13 = 2 \cdot 5 + 3 \rightsquigarrow R_5(13) = 3 = 13/5 = 2 \text{ R } 3$.

Wenn $a, b \in \mathbb{Z}$ und $n \in \mathbb{Z}^+$. Dann ist a kongruent zu $b \bmod n$, falls n ein Teiler von $a - b$ ist, kurz: $n|(a - b)$. Wir schreiben dann kurz:

$$a \equiv b \pmod{n}$$

Natürlich gilt dann auch $b \equiv a \pmod{n}$.

Lösung linearer Diophantischer Gleichungen

Für $n, n_1, n_2 \in \mathbb{Z}$ hat die lineare Diophantische Gleichung

$$n_1 \cdot x + n_2 \cdot y = n$$

genau dann ganzzahlige Lösungen, $x, y \in \mathbb{Z}$ falls $ggT(n_1, n_2)|n$, also wenn $ggT(n_1, n_2)$ ein Teiler von n ist. In den Beispielen und Aufgaben wird in diesem Modul dies beschränkt auf ganzzahlige Lösungen von linearen Diophantischen Gleichungen $n_1 \cdot x + n_2 \cdot y = 1$. Also existieren Lösungen, wenn $ggT(n_1, n_2)|1$. Wenn n_1 und n_2 teilerfremd sind, dann hat die Gleichung $n_1x + n_2y = 1$ Lösungen!

Beispiel: Die Gleichung $3x + 7y = 1$ hat wegen $ggT(3, 7) = 1 \text{ und } 1|1$ ganzzahlige Lösungen.

Zur Lösung nach x und y benötigt man den erweiterten euklid'schen Algorithmus, der nicht nur den ggT berechnet, sondern auch Werte für x und y sodass $n_1x + n_2y = ggT(n_1, n_2)$.

Modulare Inverse

Das modulare Inverse einer Zahl a ist die Zahl b, falls:

$$(a \cdot b) \bmod m = 1 \qquad \text{oder} \qquad a \cdot b \equiv 1 \pmod{m}$$

Das modulare Inverse existiert jedoch nur, wenn a und m teilerfremd sind, das heisst sie haben keinen gemeinsamen Teiler ausser 1. Also $ggT(a, m) = 1$. Wenn wir mittels Euklidischem Algorithmus x und y bestimmen, ist $ax + bx = ggT(a, b)$ und somit x das modulare inverse.

Das positive modulare inverse

Wenn das modulware inverse beim berechnen einen negativen Wert ergibt, kann man solange $+m$ addieren, bis man einen positiven Wert erhält. z.B. das modulare inverse von $2 \mod 19$:

Das modulare inverse ist -9. Da wir aber teilweise weiterrechen möchten und dies mit einem positivem Wert einfacher ist, addieren wir nun solange das modulare (hier: 19), bis wir eine positive Zahl erhalten: $-9 + 19 = 10$. Hier müssen wir dies nur einmal dazu addieren. Unser positives modulares inverses ist also 10.

Lösung mithilfe dem erweitertem Euklidischem Algorithmus

Beispiel: Finde $x, y \in \mathbb{Z}$ mit $211 \cdot x + 13 \cdot y = 1$.

Euklidischen Algorithmus mit Zahlen $n_1 = 211$ und $n_2 = 13$ durchführen: $211 \equiv 3 \mod 13$

$$\begin{aligned} 211 \bmod 13 &= 3 \\ 13 \bmod 3 &= 1 \\ 3 \bmod 1 &= 0 \end{aligned} \qquad \begin{aligned} 211 &= 16 \cdot 13 + 3 \\ 13 &= 4 \cdot 3 + 1 \\ 3 &= 3 \cdot 1 + 0 \end{aligned}$$

Somit ist der $ggT(211, 13) = 1$. Jetzt löst man die letzte Gleichung vor $\cdots = 0$ nach 1 auf un dnutzt die restlichen Gleichungen in umgekehrter Reihenfolge, um störende Terme zu eliminieren.

$$\begin{aligned} 1 &= 13 - 4 \cdot 3 \\ 1 &= 13 - 4 \cdot (211 - 16 \cdot 13) \\ 1 &= 13 - 4 \cdot 211 + 64 \cdot 13 \\ 1 &= \underbrace{-4}_{x} \cdot 211 + \underbrace{65}_{y} \cdot 13 \end{aligned}$$

Einfacher, kann man dies mit einer Tabelle lösen:

> ⚠ **ACHTUNG**! Wenn $a < b$ vertauscht man diese. Da man ja ein $a > b$ haben will. ABER, da man a und b in der Tabelle dann vertauscht hat, müsste man dann auch x und y vertauschen. Also das gesuchte modulare inverse, wenn wir die Tabelle gleich belassen, wäre bei einer Vertauschung von a und b neu das **y**, und **nicht** das x.

i (Schritt)	a	b	q (wie oft b in a)	r (Rest)	x	y
1	211	13	16	3	-4	65
2	13	3	4	1 = ggT	1	-4
3	3	1	3	0	0	1

Wobei

- $a_i = b_{i-1}$
- $b_i = r_{i-1}$
- $q_i = \left\lfloor \frac{a_i}{b_i} \right\rfloor$
- $r_i = a \bmod b$
- $x_i = y_{i+1}$, für das letzte x gilt $x = 0$
- $y_i = x_{i+1} - q_i \cdot y_{i+1}$, für das letzte y gilt $y = 1$

Modulares Inverses für Primzahlen als Modulo

Wenn die Zahl für module eine Primzahl ist, kann man das modulare inverse wie folgt berechnen:

$n^{-1} = R_m(n^{m-2}) = n^{m-2} \mod m$

Wobei:

- m: modulo (Primzahl)
- n: Basis für modulares Inverses

Der chinesische Restsatz (Sun Tsu Suan-Ching)

Welche Zahl(en) x ergeben bei der Division durch 5 den Rest 3, bei der Division durch 7 den Rest 2 und bei der Division durch 9 den Rest 4? Diese Frage beantwortet der Satz von **Sun Tsu Suan-Ching**.

Seien $m_1, m_2 \dots m_k \in \mathbb{N}^+$ paarweise teilerfremde Zahlen und $m := m_1 \cdot m_2 \cdots m_k$. Dann besitzt das System von k simultanen Kongruenzen eine eindeutige Lösung $x(\mod m)$.

$$\begin{aligned} x &\equiv r_1 \pmod{m_1} \\ x &\equiv r_2 \pmod{m_2} \\ &\vdots \\ x &\equiv r_k \pmod{m_k} \end{aligned}$$

Gehen wir von nun an von einem Beispiel mit 3 m's aus.

Schritt 1: In einem ersten Schritt definiert man alle a's und $m's$. Dann errechnet man M:

$$M = m_1 \cdot m_2 \cdot m_3$$

Schritt 2: Danach berechnet man die einzelnen M_i's, wobei $M_i = \frac{M}{m_i}$. Also z.B.

$$M_1 = m_2 \cdot m_3, \quad M_2 = m_1 \cdot m_3, \quad M_3 = m_1 \cdot m_2$$

Schritt 3: Man berechnet für jedes der M_i's die modularen inversen y_i:

$$M_i \cdot y_i \equiv 1 (\mod m_i)$$

Und hat somit dann das modulare inverse (hier: y_1, y_2, y_3) jedes m's.

Schritt 4: Man berechnet die Lösung mittels

$$x = \sum_{i=1}^{k} r_i \cdot M_i \cdot y_i$$

Dies sieht dann für ein Beispiel mit drei $m's$, also $k = 3$, so aus:

$$x = a_1 \cdot y_1 \cdot M_1 + a_2 \cdot y_2 \cdot M_2 + a_3 \cdot y_3 \cdot M_3$$

Jetzt hat man also die Lösung x. Diese ist aber nicht unbedingt die kleinste Lösung des chinesischen Restsatzes. Es kann sein, dass wir nun solange M von x subtrahieren müssen, bis wir die kleinstmögliche Lösung für $x > 0$ erhalten. Für eine Lösung wie $x = 881, M = 210$ muss man also $881 \mod 210 = 41$ rechnen. **41** ist dann die kleinste Lösung.

Die Eulersche ϕ-Funktion

Folgende Mengen sind in der Kryptologie wichtig:

$$\mathbb{Z}_n := \{\, 0, 1, 2, \ldots, n-1 \,\}$$
$$\mathbb{Z}_n^* := \{\, x \in \mathbb{Z}_n \mid x > 0 \text{ und } ggT(x, n) = 1 \,\} \text{ mit } |\mathbb{Z}_n^*| := \text{Anz. Elemente in } \mathbb{Z}_n^*$$

Die Eulersche ϕ-Funktion ist gegeben durch:

$$\phi : \mathbb{N} \to \mathbb{N}, n \mapsto |\mathbb{Z}_n^*| =: \phi(n)$$

sie ordnet jeder natürlichen Zahl n die Anzahl der zu ihr teilerfremden natürlichen Zahlen zu, die kleiner als n sind.

Beispiel: Bestimme $\phi(4)$: Man definiere dazu auch $\mathbb{Z}_i^*$ für $i = 4$:

$$ggT(0,4) = 4,\ ggT(1,4) = 1,\ ggT(2,4) = 2,\ ggT(3,4) = 1$$
$$\phi(4) = |\mathbb{Z}_4^*| = |\{\, 1, 3 \,\}| = 2 \qquad \mathbb{Z}_4 = \{\, 0, 1, 2, 3 \,\}$$

Eigenschaften der Eulerschen ϕ-Funktion

$$\phi(p) = p - 1$$
$$\phi(p \cdot q) = (p-1) \cdot (q-1)$$
$$\phi(m) = (p_1 - 1) \cdot p_1^{r_1 - 1} \cdot (p_2 - 1) \cdot p_2^{r_2 - 1} \ldots (p_n - 1) \cdot p_n^{r_n - 1}$$
$$\phi(m \cdot n) = \phi(m) \cdot \phi(n)$$

Wobei

- p und q zwei verschiedene Primzahlen sind

Beispiel: Angenommen, man möchte $\phi(60)$ berechnen.

Die Primfaktorzerlegung von 60 ist $2^2 \cdot 3^1 \cdot 5^1$.

Die Phi-Funktion für eine Potenz einer Primzahl p^k ist $\phi(p^k) = p^k - p^{k-1}$. Wir werden also diese Regel auf jeden Primfaktor an:

$$\phi(2^2) = 2^2 - 2^{2-1} = 4 - 2 = 2$$
$$\phi(3^1) = 3^1 - 3^{1-1} = 3 - 1 = 2$$
$$\phi(5^1) = 5^1 - 5^{1-1} = 5 - 1 = 4$$

Nun multipliziert man diese Werte, um $\phi(60)$ zu erhalten:

$$\phi(60) = \phi(2^2) \cdot \phi(3^1) \cdot \phi(5^1)$$
$$\phi(60) = 2 \cdot 2 \cdot 4 = 16$$

Also gibt es 16 positive ganze Zahlen, die teilerfremd zu 60 sind und kleiner als 60 sind.

Der kleine Satz von Fermat

Sei p eine Primzahl und m eine nichtnegative ganze Zahl. Dann gilt

$$m^p \bmod p = m \bmod p$$

Rechnen in Restesystemen

Relationen und Äquivalenzrelationen

Eine Relation R auf einer Menge A ist eine Teilmenge von $A \times A$.

Beispiel: Sei A die Menge aller Landpunkte eines virtuellen Planeten. Dann ist

$A \times A = \{ (x, y) \mid x \in A \land y \in A \}$

Wir definieren die Relation R auf A wie folgt:

$R = \{ (x, y) \in A \times A \mid y \text{ lässt sich von x aus trockenen Fusses erreichen} \}$

Somit gilt $(x, y) \in \mathbb{R}$ falls x und y auf der selben Insel (Kontinent) liegen.

Zahlenbeispiel: Für ein $n \in \mathbb{N} \setminus \{ 0 \}$ betrachten wir die folgende Relation auf $\mathbb{Z}$

$$R = \{ (x, y) \in \mathbb{Z} \times \mathbb{Z} \mid y \equiv x \bmod n \}$$

Nehmen wir als Beispiel $n = 11$. Dann hat man beispielsweise $(1, 1) \in R, (1, 12) \in R, (-2, -13) \in R$ aber $(2, 3) \notin R$.

Eigenschaften: Relationen

Eine Relation R auf der Menge A heisst **reflexiv**, falls $\forall x \in A((x, x) \in R)$.

Beispiel: Wenn A die Menge der Landpunkte eines virtuellen Planeten ist, dann ist $(x, y) \in R$ für Punkte $x$und y, die auf derselben Insel liegen.

Eigenschaften: Reflexive Relationen

Eine Relation R ist reflexiv, wenn jedes Element zu sich selbst in Relation steht.

Beispiel: Jeder Punkt auf einer Insel kann sich selbst erreichen.

Eigenschaften: Symmetrische Relationen

Eine Relation R ist symmetrisch, wenn aus $(x, y) \in R$ folgt, dass $(y, x) \in R$.

Beispiel: Wenn ein Punkt y von einem Punkt x aus erreichbar ist, ist auch x von y aus erreichbar.

Eigenschaften: Transitive Relationen

Eine Relation R ist transitiv, wenn aus $(x, y) \in R$ und $(y, z) \in R$ folgt, dass $(x, z) \in R$.

Beispiel: Wenn ein Punkt y von x aus und z von y aus erreichbar ist, dann ist z auch von x erreichbar.

Eigenschaften: Äquivalenzrelationen

Eine Relation ist eine Äquivalenzrelation, wenn sie reflexiv, symmetrisch und transitiv ist.

Die oben genannten Eigenschaften auf den Landpunkten eines virtuellen Planeten bilden eine Äquivalentrelation.

Restklassen

Eine Restklasse bezeichnet eine Menge von Zahlen, die alle denselben Rest bei Division durch eine bestimmte Zahl n haben. Formal wird die Restklasse von $r \bmod n$ definiert als:

$$[r] = \{ x \in \mathbb{Z} \mid x \equiv r \bmod n \}$$

Das bedeutetm dass $[r]$ alle Zahlen x enthält, die bei Division durch n den Rest r ergeben. Zum Beispiel, wenn $n = 3$, dann gibt es die Restklassen $[0], [1], \text{ und } [2]$. Die Restklasse $[1]$ enthält alle Zahlen, die bei Division durch 3 den Rest 1 ergeben, wie zum Beispiel $-5, -2, 1, 4, 7$ usw. In jeder Restklasse gibt es genau einen der Reste in $Z_n = \{ 0, 1, 2, \ldots, n - 1 \}$, die bei der Division durch n auftreten können.

Modulare Rechenoperationen

Sei $n \geq 2$. Wir führen auf der Menge $\mathbb{Z}_n = \{ 0, 1, 2, \ldots, n-1 \}$ eine Addition $\oplus_n$ (Addition modulo n) und eine Multiplikation $\odot_n$ (Multiplikation modulo n) ein. Für $a, b \in \mathbb{Z}_n$ sei:

$$a \oplus_n b = a + b \bmod n = R_n(a + b)$$
$$a \odot_n b = a \cdot b \bmod n = R_n(a \cdot b)$$

Man kann dies auch noch mit weiteren Operationen auf $\mathbb{Z}_n$ einführen, z.B:

$$a \ominus_n b = a - b \bmod n = R_n(a - b)$$

Rechenregeln

Kommutativgesetz bei der Addition:

$a \oplus_n b = b \oplus_n a$

Kommutativgesetz bei der Multiplikation:

$a \odot_n b = b \odot_n a$

Neutralelement der Addition:

$a \oplus_n 0 = a$

Neutralelement der Multiplikation:

$a \odot_n 1 = a$

Distributivgesetz:

$a \odot_n (b \oplus_n c) = (a \odot_n b) \oplus_n (a \odot_n c)$

Square and Multiply

Der Square and Multiply Algorithmus wird verwendet, um effizient hohe Potenzwerte modulo zu rechnen. Dies in folgenden Schritten:

Beispiel: $5^{21} \bmod 11$

Schritt 1: Exponent binär schreiben:

$$21 = (10101)_2$$

Schritt 2: Für 1 QM einsetzen und für 0 Q einsetzen, das erste QM streichen:

$$(10101)_2 \rightarrow QMQQMQQM \rightarrow QQMQQM$$

Schritt 3: Gemäss Q/QM-String Quadrieren oder Quadrieren und Multiplizieren.

$$5 \xrightarrow{Q} 25 \equiv 3 \xrightarrow{Q} 9 \xrightarrow{M} 45 \equiv 1 \xrightarrow{Q} 1 \xrightarrow{Q} 1 \xrightarrow{M} 5$$

Schritt 3 genauer:

1. Man quadriert 5 (die Basis in unserem Beispiel), was zu 25 wird.
2. Man rechnet nun 25 modulo 11, was zu 3 wird.
3. Die 3 quadriert man dann, was 9 ergibt.
4. Diese 9 multipliziert man dann mit der Basis 5, was 45 ergibt.
5. Die 45 modulo 11 ergibt dann 1
6. Die 1 quadriert man, was wieder 1 gibt, und dann nochmal das gleiche mit dieser 1.
7. Zuletzt wird die 1 nochmals mit der Basis (hier: 5) multiplizert, was den Finalwert 5 ergibt.

Nullteiler

Existiert zu einem $a \in \mathbb{Z}_n$ mit $a \neq 0$ ein $b \in \mathbb{Z}_n$ mit $b \neq 0$ so dass $a \odot_n b = 0$ gilt, so heisst a ein **Nullteiler** von $\mathbb{Z}_n$.

Für $\mathbb{Z}$ gilt, dass: $a \cdot b = 0 \rightarrow a = 0 \vee b = 0$

Bemerkungen:

- Nullteiler gibt es in $\mathbb{Z}$ oder $\mathbb{R}$ (mit der gewöhnlichen Multiplikation) nicht.
- Falls es in $\mathbb{Z}_n$ Nullteiler gibt, kann in $\mathbb{Z}_n$ nicht jede Gleichung $a \odot_n x = b$ gelöst werden.
  Also
  $a \odot_n b = 0 \nrightarrow a = 0 \vee b = 0$

Sei $n = p$ eine Primzahl. Dann gilt:

- $\mathbb{Z}_p$ ist nullteilerfrei
- für alle $a, b \in \mathbb{Z}_p$ besistzt die folgende Gleichung genau eine Lösung $x (x = b \cdot a^{-1} = a' - 1 \cdot b)$.

$$a \odot_p x = b \qquad \text{bzw.} \quad a \cdot x \equiv b \bmod p$$

Multiplikativ Inverse Elemente

In der Menge der ganzen Zahlen $\mathbb{Z}$ hat fast kein Element ein multiplikatives Inverses. Die Gleichung $x \cdot y = 1$ hat in $\mathbb{Z}$ nur die Lösungen $1 \cdot 1 = 1$ und $(-1) \cdot (-1) = 1$.

In der Menge $\mathbb{R}$ ist jedes Element bis auf 0 invertierbar, denn $x \cdot \frac{1}{x} = 1$.

Erstaunlicherweise haben in $\mathbb{Z}_n$ viele Elemente ein multiplikativ Inverses.

Das Theorem

Sei n eine natürliche Zahl und $a \in \mathbb{Z}_n$. Dann hat a genau dann ein Inverses bezüglich der Multiplikation $\odot_n$ in $\mathbb{Z}_n$ (also der Multiplikation modulo n), wenn $ggT(a, n) = 1$ gilt.

Beispiel:

In $\mathbb{Z}_3 = \{ 0, 1, 2 \}$ gilt $\mathbb{Z}_3^* = \{ 1, 2 \}$, denn

- $1 \odot_3 1 = 1$ also $1^{-1} = 1$
- $2 \odot_3 2 = 1$ also $2^{-1} = 2$

$\odot_3$	**0**	**1**	**2**
0	0	0	0
1	0	1	2
2	0	2	1

Multiplikative Inverse können mithilfe des Euklidischen Algorithmus einfach bestimmt werden.

Primitives Element

Ein Element aus z aus $\mathbb{Z}_p^*$ wird als primitives Element bezeichnet, wenn jede Zahl in $\mathbb{Z}_p^*$ durch Potenzieren von z dargestellt werden kann. Formal ausgedrückt, z ist ein primitves Element, wenn die Potenzen von z, das heisst $z^1, z^2, z^3, \ldots, z^{p-1}$, modulo p alle unterschiedlichen Elemente von $\mathbb{Z}_p^*$ ergeben.

Zum Beispiel ist die Zahl 2 ein primitives Element im Restklassenring $\mathbb{Z}_5^*$, weil die Potenzen von 2, also $2^1, 2^2, 2^3$, und 2^4, modulo 5 alle unterschiedlichen nicht-null Elemente von $\mathbb{Z}_5^*$ erzeugen. In diesem Fall:

$$\begin{aligned} 2^1 &\equiv 2 \pmod 5 \\ 2^2 &\equiv 4 \pmod 5 \\ 2^3 &\equiv 3 \pmod 5 \\ 2^4 &\equiv 1 \pmod 5 \end{aligned}$$

Die Elemente $\{ 2, 4, 3, 1 \}$ sind alle unterschiedlich und repräsentieren alle Elemente von $\mathbb{Z}_5^*$ ausser der Null. Da die Potenzen von 2 die gesamte Menge abdecken, ist 2 ein primitves Element.

Einwegfunktionen

Einwegfunktionen sind Funktionen, die einfach auszuführen, aber schwer (oder besser: praktisch unmöglich) zu invertieren sind.

Beispiele einiger Einwegfunktionen (nach heutigem Wissensstand). Die Zahlen p und q seien dabei stets verschiedene Primzahlen.

- Quadrieren modulo $n = pq$: $x \mapsto x^2 \bmod n$
- Potenzieren modulo $n = pq$: $x \mapsto x^e \bmod n$
- Diskrete Exponentialfunktion modulo p: $k \mapsto b^k \bmod p$

Modulare Quadratwurzeln (Quadratreste und Nichtreste)

Modulare Quadratwurzeln (und modulare Logarithmen) sind in der Kryptographie von besonderem Interesse (Einwegfunktionen).

Sei $a \in \mathbb{Z}_n^*, n \geq 2$. Eine Lösung $x \in \mathbb{Z}_n^*$ (falls sie existiert) der Gleichung

$$x^2 = a \bmod n \qquad \text{bzw. } x \odot_n x = a$$

heisst modulare Quadratwurzel von a modulo n. In diesem Fall nennt man a quadratischer Rest (QR) modulo n. Andernfalls heisst a quadratischer Nichtrest (NR).

Berechnung modularer Quadratwurzeln

Beispiel: Für $n = 7$ und $\mathbb{Z}_7^* = \{1, 2, 3, 4, 5, 6\}$

Am besten macht man dafür eine Tabelle, und berechnet die modularen Quadratwurzeln in dem man für jedes x die Gleichung $x^2 = x \odot_7 x$ löst.

x	1	2	3	4	5	6
$x^2 = x \odot_7 x$	1	4	2	2	4	1

Etwas genauer:

- Für $x = 1$: Wir rechnen $1 \cdot 1$, was 1 gibt. Modulo 7 ändert nichts am Ergebnis, da der Rest 1 bleibt.
- Für $x = 2$: Wir rechnen $2 \cdot 2$, was 4 gibt. Modulo 7 ändert nichts am Ergebnis, da der Rest 4 bleibt.
- Für $x = 3$: Wir rechnen $3 \cdot 3$, was 9 gibt. Modulo 7 ändert hier das Ergebnis, da $9 \bmod 7 = 2$. Somit ist 2 hier unser Resultat.

Nun muss man noch folgendes machen:

QR/NR?	QR	QR	NR	QR	NR	NR
a	1	2	3	4	5	6
$\sqrt{a} \bmod 7$	1, 6	3, 4	-	2, 5	-	-

Etwas genauer:

- Für $a = 1$: WIr schauen in unserer Tabelle von vorher für welche x wir den Rest 1 erhalten haben. Dies trifft für $x = 1, 6$ zu. Somit ist $\sqrt{1} \bmod 7 = 1, 6$. Somit ist für $a = 1$ ein quadratischer Rest (QR) vorhanden.
- Für $a = 2$: Wir schauen in unserer Tabelle von vorher für welche x wir den Rest 2 erhalten haben. Dies trifft für $x = 3, 4$ zu. Somit ist $\sqrt{2} \bmod 7 = 3, 4$. Somit ist für $a = 2$ ein quadratischer Rest (QR) vorhanden.
- Für $a = 3$: Wir schauen in unserer Tabelle von vorher für welche x wir den Rest **3** erhalten haben. Dies trifft für keines der x zu. Somit ist für $a = 3$ kein quadratischer Rest vorhanden und es handelt sich um einen quadratischen Nichtrest (NR).

Beispiel: für n = 21 (keine Primzahl)

21 besteht aus der Primfaktorzerlegung von $7 * 3$.

Wenn wir hier die Quadratwurzeln von 1 berechnen wollen machen wir folgendes Tabelle:

x	1	2	4	5	8	10	11	13	16	17	19	20
$x \odot_{21} x$	1	4	16	4	1	16	16	1	4	16	9	1

Wir sehen hier, dass einige Werte für x fehlen. Da quadratische Reste und quadratische Nichtreste nur für teilerfremde x gelten, **dürfen wir die vielfachen von $7, 3$ nicht berechnen!** Somit sind die quadratischen Reste nur **1, 4, 16** und die quadratischen Wurzeln für z.B. 1 sind 1, 8, 13 und 20.

Modulare Quadratwurzeln — Das Euler-Kriterium

Mit Hilfe des Euler-Kriteriums kann man schnell überprüfen, ob eine bestimmte Zahl ein quadratisches Residum modulo eine Primzahl ist.

Theorem

Sei p eine ungerade Primzahl. Dann ist $a < p$ ein quadratischer Rest modulo p genau dann, wenn

$$a^{\frac{p-1}{2}} \equiv 1 \bmod p$$

Beispiel: Sei $p = 11$. Wir zeige nun, dass 3 ein quadratisches Residuum modulo 11 ist. Nach dem Euler-Kriterium hat man

$$3^{\frac{11-1}{2}} = 3^5 = 3^3 \cdot 3^2 = 27 \cdot 9 \equiv 5 \cdot 9 = 45 \equiv 1 \bmod 11$$

Tatsächlich ist $5^2 = 25 \equiv 3 \bmod 11$ und auch $(11-5)^2 = 6^2 = 36 \equiv 3 \bmod 11$.

Sei $n = p \cdot q$ das Produkt zweier verschiedener Primzahlen p und q (beide $\neq 2$). Dann ist das Berechnen von Quadratwurzeln modulo n genau so schwierig, wie das Faktorisieren von n.

Der diskrete Logarithmus

Die diskrete Exponentialfunktion zur Basis b modulo p ist definiert (und effizient berechenbar) durch:

$$\exp_b(k) = b^k \bmod p$$

Unter des Problem des diskreten Logarithmus versteht man:

Finde zu gegebenem $y \in \mathbb{Z}_p^*$ ein $k \in \mathbb{N}$ (falls es existiert), so dass die folgende Gleichung erfüllt ist:

$$y = b^k \bmod p$$

Man schreibt auch: $k = log_b(y) \bmod p$

Die diskrete Exponentialfunktion scheint eine Einwegfunkton zu sein, denn $k \longrightarrow b^k \bmod p$ ist einfach, $y \longrightarrow log_b(y) \bmod p$ ist schwierig auszuführen.

Beispiel: Finden Sie eine Lösung k der Gleichung $7 = 3^k \bmod 17$.

Dies bedeutet, wir brauchen eine Lösung für $k = log_3(7) \bmod 17$ wodurch $k = 11$ folgt.

Diffie-Hellman Schlüsselvereinbarung (SB-Version)

Können zwei Personen A und B ein Geheimnis (z.B. einen Schlüssel) vereinbaren, obwohl jemand ihre Kommunikation vollständig überwacht? Falls es Einwegfunktionen gibt, geht das.

Schritt 1:

Wähle zwei (allg. bekannte) natürliche Zahlen p (prim) und $g < p$ (Erzeugende von $\mathbb{Z}_p^*$)

Schritt 2:

A wählt eine (nur ihm bekannte) Zufallszahl a < p, berechnet $\alpha = g^a \bmod p$ und sendet α über einen öffentlichen Kanal an B.

B wählt eine (nur ihm bekannte) Zufallszahl b < p, berechnet $\beta = g^b \bmod p$ und sendet β über einen öffentlichen Kanal an A.

Schritt 3:

A berechnet $\beta^a \bmod p = g^{b \cdot a} \bmod p$ und B berechnet $\alpha^b \bmod p = g^{b \cdot a} \bmod p$

Schritt 4:

Beide haben den gemeinsamen Schlüssel $K = \beta^a \bmod p = \alpha^b \bmod p$ und ein Angreifer kann aus der Kenntnis von α und β (sowie natürlich p und g) den Schlüssel K praktisch (wegen dem diskreten Logarithmusproblem) nicht rekonstruieren.

An einem Bespiel: Vereinbaren eines Schlüsels mit p = 17, g = 5.

Alice

Zufallszahl von 1-17 wählen.

$$2 \longleftarrow_R \mathbb{Z}_{17}^*$$

α berechnen und an Bob senden:

$$\alpha = 5^2 \bmod 17 = 8$$

Alice berechnet gemeinsamen Schlüssel K

$$K = \beta^a \bmod p = 10^2 \bmod 17$$
$$K = 15$$

Bob

Zufallszahl von 1-17 wählen

$$7 \longleftarrow_R \mathbb{Z}_{17}^*$$

β berechnen und an Alice senden:

$$\beta = 5^7 \bmod 17 = 10$$

Bob berechnet gemeinsamen Schlüssel K

$$K = \alpha^b \bmod p = 8^7 \bmod 17$$
$$K = 15$$

Symmetrische Verschlüsselung

Ein symmetrischer Verschlüsselungsalgorithmus besteht aus

- einer /Verschlüsselungs-) Funktion f mit zwei Variabeln
- dem Schlüssel k
- dem Klartext m
- dem Geheimtext c, der sich wie folgt ergibt: $c = f(k, m)$

Die Verschlüsselungsfunktion f muss umkehrbar sein, das heisst, es muss eine Entschlüsselungsfunktion f^* geben, die die Wrikung von f neutralisiert. Also muss gelten:

$$m = f^*(k, c) = f^*(k, f(k, m))$$

Caesar-Chiffre

Bei der Caesar-Chiffre handelt es sich um eine Verschlüsselungsverfahren, bei welchem jeder Buchstabe um eine bestimmte Anzahl Stellen verschoben wird.

$$c = f(k, m) = m + k \bmod 26$$
$$m = f^*(k, c) = c - k \bmod 26$$

Diese Rechenoperationen gelten, nachdem man die Buchstaben mit Zahlen 0 - 25 ersetzt hat.

Um die Nachricht “informatik” mit dem Schlüssel $k = 3$ zu verschlüsseln, und dann wieder entschlüsseln, gehen wir wie folgt vor:

- “i” ist im Alphabet an 9. Stelle, also rechnen wir $9 + k = 9 + 3 = 12$. Also nehmen wir anstelle i den Buchstaben an 12. Stelle: “L”.

Dies machen wir für jeden Buchstaben im Wort “informatik” und erhalten dann “LQIRUPDWLN”.

Um dies zu entschlüsseln machen wir genau das umgekehrte also f^*.

Schlüsselwortchiffre

Die Schlüsselwortchiffre geht gleich vor wie die Caesar-Chiffre, nur ist k anstelle einer Zahl, ein Wort. Das Alphabet wird also um das Wort k (wobei jeder Buchstabe im Wort nur einmal vorkommt in k) verschoben.

Vigenère-Chiffre

Auch bei der Viginère-Chiffre ist das Geheimnis ein ganzes Wort (oder kann auch mehr sein, etwas wie ein Passwort). z.B. nehmen wir das Wort CAESAR. Dabei werden oft Buchstaben weggelassen, damit sie nur einmal vorkommen, da dies sicherer gegen Häufigkeitsanalysen sind.

Wir verschlüsseln dann einen Text wie “heute ist freitag” mit diesem Wort, wie wir es bei der Caesar-Chiffre gemacht haben. Nur verschlüsselt man jeden Buchstaben in der Nachricht mit der Reihe nach mit den Buchstaben im Wort CAESAR (k). Also das “h” mit “c”, dann das “e” mit “a”, dann das “u” mit “e”, usw. Dann entsteht die verschlüsselte Nachricht: “jeyle zut jjezvak”. Natürlich lässt man oft die Leerzeichen weg, damit nicht bez. der Wortlänge eine Häufigkeitsanalyse gemacht werden kann, was dann “jeylezutjjezvak” wäre.

Kerckhoffsches Prinzip

Das Kerckhoffs’sche Prinzip (1883) sagt: Man muss grundsätzlich davon ausgehen, dass f und f^* allgemein bekannt sind. “Hence, no security through obscurity!” Ein sicheres Verschlüsselungsschema (M, C, K, f, f*) muss folgende Angriffe überstehen:

- **Ciphertext-Only-Attacke:** Der Angreifer kennt eine begrenze Anzahl von Geheimtexten (wobon man eigentlich immer ausgehen kann).
- **Known-Plaintext-Attacke**: Der Angreifer kennt eine begrenzte Anzahl von Geheimtexten mit den zugehörigen Klartexten (was z.B. vorkomm,w enn man die selbe Meldung verschlüsselt und unverschlüsselt übermittelt).
- **Chosen-Plaintext-Attacke:** Der Angreifer hat die Möglichkeit zu einem (von ihm ausgewählten) Klartext an den zugehörigen Geheimtext zu gelangen (Angreifer kennt das Verschlüsselungsverfahren).

Perfekte Sicherheit

Für perfekte Sicherheit im Sinne von Shannon muss die Verteilung der Klartexte unabhängig von der Verteilung der Geheimtexte sein. Das bedeutet, dass das Wissen über den Geheimtext einem Angreifer keine Informationen über den Klartext gibt. In mathematischen Begriffen ausgedrückt bedeutet das:

$p(m|c) = p(m)$

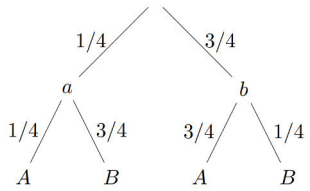
für alle Klartexte m und Geheimtexte c:

- $p(m|c)$ ist die bedingte Wahrscheinlichkeit, dass der Klartext m ist, gegeben den Geheimtext c
- $p(m)$ ist die Wahrscheinlichkeit des Klartextes ohne jegliche Kenntnis des Geheimtextes.

Um also zu überprüfen ob ein System perfekte Sicherheit bietet, können wir für jeden Geheimtext die bedingte Wahrscheinlichkeit des Klartextes berechnen und sie mit der Wahrscheinlichkeit des Klartextes ohne Kenntnis des Geheimtextes vergleichen.

Beispiel: Gegeben sei die Klartextmenge $M = \{ a, b \}$, die Geheimtextmenge $C = \{ A, B \}$ und die Schlüsselmenge $K = \{ 0, 1 \}$, sowie die Wahrscheinlichkeitsverteilungen auf der Klartextmenge $p(a) = 1/4, p(b) = 3/4$ und auf der Schlüsselmenge $p(0) = 1/4, p(1) = 3/4$. Die Verschlüsselungsfunktion f sei wie folgt definiert:

$$
\begin{aligned}
f(0, a) &= A \\
f(0, b) &= B \\
f(1, a) &= B \\
f(1, b) &= A
\end{aligned}
$$



Um die perfekte Sicherheit zu bestimmen, müssen wir hier$p(A), p(B), p(A|a), p(A|b), p(B|a)$ und $p(B|b)$ bestimmen. Man kann dann den Satz von Bayes verwenden um $p(a|A)$ zu berechnen und mit $p(a)$ zu vergleichen. Man findet dann heraus, dass dies nicht perfekt sicher ist, da:

$$p(A) = p(a) \cdot P(A|a) + P(b) \cdot P(A|b) = \frac{1}{4} \cdot \frac{1}{4} + \frac{3}{4} \cdot \frac{3}{4} = \frac{10}{16}$$

$$P(B) = 1 - P(A) = \frac{6}{16}$$

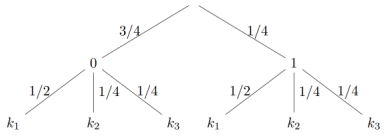
$$P(a|A) = \frac{P(a \cap A)}{P(A)} = \frac{P(a) \cdot P(A|a)}{P(A)} = \frac{\frac{1}{4} \cdot \frac{1}{4}}{\frac{10}{16}} = \frac{1}{16} \cdot \frac{16}{10} = \frac{1}{10} < P(a) = \frac{1}{4}$$

Anderes Beispiel:

Hier ist $M = \{ 0, 1 \}, G = \{ 00, 01, 10, 11 \}, k = \{ k_1, k_2, k_3 \}$

Die W'keitverteilung: $p(0) = \frac{3}{4}, p(1) = \frac{1}{2}, p(k_1) = \frac{1}{2}, p(k_2) = \frac{1}{4}, p(k_3) = \frac{1}{4}$

$$
\begin{aligned}
f(k_1, 0) &= 01 \qquad & f(k_1, 1) &= 10 \\
f(k_2, 0) &= 10 \qquad & f(k_2, 1) &= 11 \\
f(k_3, 0) &= 11 \qquad & f(k_3, 1) &= 00
\end{aligned}
$$



Wir überprüfen hier jetzt also ob $p(m|c) = p(m)$. Wir können dies in diesem Beispiel recht einfach für $m = 0$ tun. Die Wahrscheinlichkeit, dass $m = 0$ ist wie dem Graphen zu entnehmen ist $3/4$. Somit müssen wir nun überprüfen, ob $p(0|00)$ ebenfalls $3/4$ ist. Da in diesem Fall $p(0|00)$ gar nicht eintreten kann, da es für $m = 0$ kein $c = 00$ gibt, ist die W'keit $p(0|00) = 0$. Und somit ist $p(0|00) \neq p(0), \quad \textbf{weil:} \quad 0 \neq 3/4$. Somit ist diese Funktion nicht perfekt sicher.

Block- und Stromchiffren

Die Algorithmen zur symmetrischen Verschlüsselung von Daten unterteilen sich typischerweise in Blockchiffren und Stromchiffren.

- Bei der Blockchiffre wird die Nachricht in Blöcke $m_1, m_2, m_3, \ldots$ fester Länge (z.B. 64 Bit bei **DES**[2] oder 128 Bit bei **AES**[3] eingeteilt und jeder Block wird mit dem Schlüssel k verschlüsselt: $c_i = f(k, m_i)$ für $i = 1, 2, 3, \ldots$
  Beispiele sind AES (Advanced Encryption Standard), DES (Data Encryption Standard) und IDEA (International Data Encryption Algorithm) und viele weitere (Blow-/Twofish, etc.)
- Bei der Stromchiffre wird die Nachricht zeichenweise (z.B. bit- oder byteweise) verschlüsselt, wobei sich in der Regel der aktuell verwendete Schlüsselstrom von Zeichen zu Zeichen ändert.
  Ein typisches Beispiel ist das One-Time-Pad (OTP), dt. Einmalverschlüsselungsverfahren.

Der One-Time-Pad (OTP) — Pseudozufallsbitstrom

Verschlüsseln: $c_i = m_i \oplus k_i, i = 1, 2, \ldots$

Entschlüsseln: $m_i = c_i \oplus k_i, i = 1, 2, \ldots$

Das One-Time-Pad funktioniert folgendermassen:

Wenn die beiden Bits (Klartext und Schlüssel) gleich sind (beide 0 oder beide 1), ist das Ergebnis 0. Sind die unterschiedlich (eines ist 0 und das andere 1), ist das Ergebnis 1. Bei der Verwendung eines Pseudozufallszahlengenerators (PRNG) zur Erzeugung des Schlüssels entsteht ein Pseudozufallsbitstrom.

Aufteilen eines Geheimnisses auf 3 Personen

1. **Erzeugen von zufälligen Bitstrings**: Es werden zuerst zwei zufällige Bitstrings c_1 und c_2 erzeugt. Diese Bitstrings werden jeweils den ersten beiden Personen gegeben.
2. **Berechnen des dritten Bitstrings**: Der dritte Bitstrings (c_3) wird berechnet, indem das ursprüngliche Geheimnis (m) mit dem XOR der ersten beiden Bitstrings ($c_1 \oplus c_2$) verknüpft wird. Der dritte Bitstrings wird der dritten Person gegeben.
3. Rekonstruktion des Geheimnisses: Das ursprüngliche Gehemnis kann rekonstruiert werden, indem man alle drei Bitstrings miteinander verknüpft: $m = c_1 \oplus c_2 \oplus c_3$.

Die XOR-Operation sorgt dafür, dass das Gehemnis nur dann rekonstruiert werden kann, wenn alle drei Bitstrings bekannt sind. Jeder Bitstrings für sich genommen gibt keine Information über das ursprüngliche Geheimnis preis. Das stellt also sicher, dass das Geheimnis nur dann zugänglich ist, wenn alle drei Personen zusammenarbeiten.

Aber Achtung! *In Beispielen wo nur 2 der drei Bitstrings benötigt werden, z.B. wenn das Gehemnis m als der y-Achsenabschnitt einer beliebigen Geraden durch den Punkt $(0, m)$ betrachtet wird, und jede der drei Personen einen zufälligen Punkt auf dieser Geraden erhält (x, m), reichen auch nur 2 Personen um das Gehemnis m (die Gerade) zu rekonstruieren, da man nur 2 Punkte auf einer Geraden kennen muss, um die Gerade zu berechnen.*

Asymmetrische Verschlüsselung

- Jedem Teilnehmer T wird ein privater Schlüssel $d = d_T$ (geheim) zum Entschlüsseln (decryption) und ein öffentlicher Schlüssel $e = e_T$ zum Verschlüsseln zugeordnet.
- Der Verschlüsselungs-Algorithmus (Funktion) f_e zum öffentlichen Schlüssel e ordnet jedem Klartext m den Geheimtext $c = f_e(m)$ zu. Jeder kann also verschlüsseln.
- Umgekehrt ordnet der Entschlüsselungs-Algorithmus f_d zum privaten Schlüssel d dem Geheimtext c den Klartext $m' = f_d(c)$ zu. Nur wer den zum öffentlichen Schlüssel e passenden geheimen Schlüssel d hat, kann entschlüsseln.

Folgende Bedingungen müssen erfüllt sein:

- Korrekte Entschlüsselung: $m' = f_d(c) = f_d(f_e(m)) = m$
- Public-Key-Eigenschaft: Es ist praktisch unmöglich aus der Kenntnis von e (bzw. f_e) auf d (bzw. f_d) zu schliessen.

Vorgehen beim Nachricht übermitteln mittels asymmetrischer Verschlüsselung

1. Ermitteln des öffentlichen Schlüssels $e = e_T$ von T (Suche in einem elektronischen "Telefonbuch", z.B. keyserver)
2. Die Funktion f_e wird auf die zu verschlüsselnde Nachricht m angewendet und man erhält den Ciphertext $c = f_e(m)$
3. c wird (über einen öffentlichen Kanal) an T gesendet
4. Nur T kann mit seinem privaten Schlüssel $d = d_T$ die Nachricht entschlüsseln:

$$m = f_{d_T}(c) = f_{d_T}(f_{e_T}(m))$$

Der RSA-Algorithmus ist der Prototyp für die Public-Key-Kryptographie.

Der Satz von Euler — Verallgemeinerung des kleinen Satzes von Fermat

Seien $a, n \in \mathbb{N}$ mit $ggT(a, n) = 1$ und $\phi(n)$ die eulersche ϕ-Funktion (d.h. die Anzahl der zu n teilerfremden Reste modulo n). Dann gilt:

$$a^{\phi(n)} \equiv 1 \mod n$$

Für eine Primzahl n, ist dies wegen $\phi(n) = n - 1$ der kleine Satz von Fermat: $a^{n-1} \equiv 1 \mod n$.

Ist n das Produkt zweier verschiedener Primzahlen p und q, dann gilt $\phi(n) = (p-1)(q-1)$. Dann hat man zum Beispiel: $\phi(10) = \phi(2 \cdot 5) = (2-1)(5-1) = 4$.

> Im Anhang ist der Zusammenhang vom Satz von Euler und der asymmetrischen Verschlüsselung in einem einfachen Beispiel erklärt.

RSA-Algorithmus

Um eine RSA-Verschlüsselung zu knacken, müsste man einen schnellen Algorithmus kennen, der riesige Zahlen in ihre Primfaktoren zerlegen kann, oder $\phi(n)$ schnell berechnen kann, was fast unmöglich ist, wenn man den privaten Schlüssel nicht hat.

Hier wird der RSA-Algorithmus in 3 Schritten erklärt.

Schritt 1 — Schlüsselerzeugung (key generation)

1. Wähle zwei grosse (mind 300-600 stellige) Primzahlen p und q
2. Berechne $n = p \cdot q$
3. Berechne $\phi(n) = (p-1)(q-1)$
4. Wähle eine zu $\phi(n)$ teilerfremde Zahl e, d. h. überprüfe ob $ggT(\phi(n), e) = 1$.
5. Bestimme ein modulares Inverses modulo $\phi(n)$ zu e, d.h. eine Zahl d mit

   $e \cdot d = 1 + k\phi(n) \quad \text{bzw.} \quad e \cdot d \equiv 1 \mod \phi(n)$
   für eine natürliche Zahl k (erweiterter Euklidischer Algorithmus).
   also:
   $ex + \phi(n)y = 1 = ggT(e, \phi(n))$, wobei $d = x$
6. Privater (geheimer) Schlüssel: d (auch p, q und $\phi(n)$ müssen geheim gehalten werden)
7. Öffentlicher Schlüssel: (n, e)

Schritt 2 — Verschlüsseln (encryption)

Verschlüsseln eines Klartextes $m < n$ mit dem öffentlichen Schlüssel (n, e):

$$f_e(m) = m^e \mod n = c$$

Schritt 3 — Entschlüsseln (decryption)

Entschlüsseln eines Geheimtextes c mit dem privaten (geheimen) Schlüssel d:

$$f_d(c) = c^d \mod n = m'$$

wobei natürlich $m' = m$ sein sollte!

*An einem **Beispiel** gezeigt*:

1. Schlüsselerzeugung
   a. Wähle p = 307, q = 859 und $n = p \cdot q = 263'713$
   b. $\phi(n) = \phi(253713) = (p-1)(q-1) = 262'548$
   c. $e = 1721$ ist zu $\phi(263'713) = 262'548$ teilerfremd
   d. Bestimmung von $d = 1373$ (modulares Inverses zu $ex + \phi(n)y = 1 \Rightarrow 1721x + 262713y = 1$, was dann bedeutet $x = 1373 = d$)
   e. Geheimer Schlüssel: d = 1373
   f. Öffentlicher Schlüssel: n = 263'713 und e = 1721
2. Verschlüsseln von m = 138: $c = f_e(m) = 138^{1721} \mod 263'713 = 75'125$
3. Entschlüsseln von c = 75'125: $f_d(c) = 75'125^{1373} \mod 263'713 = 138$

Hier wird zum Beispiel der Square-And-Multiply Algorithmus verwendet, um die hohen Potenzen zu berechnen.

Knacken von RSA mit kleinen Werten

Öffentliche Werte sind e und n

1. Eve kann n faktorisieren: $14'803 = 131 \cdot 113$
   Dann folgt
   $\phi(14'803) = 130 \cdot 112 = 14560$
2. Eve kann $\phi(n)$ berechnen: 14560
   Dann auch
   $p = 131$ und $q = 113$:

   $$n = p \cdot q, \phi(n) = (p-1)(q-1)$$

   Somit hat man das Gleichungssystem

   $$\begin{aligned} n &= p \cdot q \\ \phi(n) &= (p-1)(q-1) \end{aligned} \qquad \begin{aligned} 14803 &= p \cdot q \\ 14560 &= (p-1)(q-1) \end{aligned}$$

   Wenn wir nach q auflösen erhalten wir also

   $$\begin{aligned} q &\overset{I}{=} \frac{14803}{p} \text{ und } 14560 \overset{II}{=} p \cdot q - p - q + 1 \\ 14560 &= 14803 - p\frac{14803}{p} + 1 \\ 14560p &= 14803p - p^2 - 14803 + p \\ p^2 - 244p + 14803 &= 0 \\ p &= \frac{244 \pm \sqrt{244^2 - 4 \cdot 1 \cdot 14803}}{2 \cdot 1} = \frac{244 \pm \sqrt{324}}{2} \\ p &= \frac{244 + 18}{2} = 131, \quad q = \frac{244 - 18}{2} = 113 \end{aligned}$$

   Mit p und q könnte man jetzt d selbst berechnen und so c entschlüsseln.

RSA-Verfahren mittels CRT (Chinesischer-Restsatz)

Um die Entschlüsselung zu optimieren, kann für grosse Werte das CRT (der Chinesische Restsatz) verwendet werden. Dies folgendermassen:

1. **Schlüsselgenerierung**

   Nehmen wir an, p = 61 und q = 53, dann ist $n = p \cdot q = 3233$.

   Bedeuted: $\phi(n) = (p-1)(q-1) = 60 \cdot 52 = 3120$.

   Wählen wir $e = 17$ (welches teilerfremd zu 3120 ist), dann berechnen wir d, das modulare inverse von e bezüglich $\phi(n)$:

   $d \cdot e \equiv 1 \mod \phi(n) \Rightarrow ex + \phi(n)y = 1, \quad \text{nach } x = d \text{ auflösen}$

   also: $17x + 3120y = 1, \quad x = 2753 = d$

   Also haben wir: p = 61, q = 53, n = 3233, e = 17, d = 2753

2. **Verschlüsselung**

   Eine Nachricht m, sagen wir $m = 123$, wird verschlüsselt zu c mit dem öffentlichen Schlüssel (n, e) durch:

   $c = m^e \mod n$
   $c = 123^{17} \mod 3233 = 855$

3. **Vorbereiten der CRT Kongruenzen**

   $v_1 = c^d \mod p \quad \rightarrow 855^{2753} \mod 61 = 1$
   $v_2 = c^d \mod q \quad \rightarrow 855^{2753} \mod 53 = 17$

   Da d gross ist, verwenden wir Square-And-Multiply, um d effizienter zu berechnen.

4. **Verwendung des CRT fürs entschlüsseln von m**

   Jetzt da v_1 und v_2 berechnet sind, gilt es mithilfe des CRT folgende Kongruenzen zu berechnen:

   $m \equiv v_1 \mod p \quad \rightarrow m \equiv 1 \mod 61$
   $m \equiv v_2 \mod q \quad \rightarrow m \equiv 17 \mod 53$
   $m = 123$

Anwendung von Zahlentheorie-Konzepten

- **Rechenoperatoren**: Die Rechenoperatoren sind die gleichen, nur dass am Ende noch modulo gerechnet wird.

  $x \oplus_m y = (x + y) \bmod m$, $x \odot_m y = (x \cdot y) \bmod m$, $x \ominus_m y = (x - y) \bmod m$

- **Negativer Modulo Wert:** Ist ein Wert negativ und man soll modulo damit rechnen, kann man solange m der Zahl dazuaddieren, bis sie positiv ist.

  $a \bmod m = (a + xm) \bmod m$ also zum Beispiel: $-3 \bmod 8 = (-3 + 8) \bmod 8 \equiv 5$

- **Modular Invertierbare Werte multiplikativer Mengen**: Bei einer Multiplikation einer Gruppe Z_n^*wobei n eine Primzahl ist, ist jedes Element ausser Null invertierbar. Wenn n keine Primzahl ist, sind die invertierbaren Elemente in Z_n die, die teilerfremd zu n sind (ggT = 1). Die Anzahl invertierbarer Werte ist $\phi(n) = (p-1)(q-1)$.

- **Multiplikative Inverse bestimmen**: Die multiplikativen Inversen invertierbarer Werte in einer Menge Z_n sind diejenigen, die bei der Operation $a \odot_n b \equiv 1$ ergeben.

- **Quadratische Reste (QR) und quadratische Nichtreste (NR)**: Ein Element einer Menge $\mathbb{Z}_n$ ist ein QR wenn es mindestens ein x gibt, sodass $x^2 \equiv a \mod n$. Der Quadratrest wird mithilfe einer Tabelle berechnet für alle $x \in \mathbb{Z}_n^*$ mit der Funktion $x \odot_n x$. Die untenstehenden Werte wenn in die Gleichung eingesetzt, $\forall x$ teilerfremd zu n, sind die QR's.

- Q**uadratwurzeln in** $\mathbb{Z}_n$: In einer Menge $\mathbb{Z}_n$ ist eine Zahl a eine Quadratwurzel, wenn ein x existiert, sodass $x^2 \equiv a \mod n$. z.B. ist 4 eine Quadratwurzel in $\mathbb{Z}_{15}$, weil $4^2 \equiv 1 \mod 15$, aber 8 hat keine Quadratwurzel, da kein x existiert, für das $x^2 \equiv 8 \mod 15$. Dies wird auch mithilfe der Tabelle für QR/NR sowie einer Zusatztabelle mit der Formel $\sqrt{a} \mod n$ gelöst.

Short word from our other sponsor: NordVPN

This summary is sponsored by NordVPN.

Staying safe online is an ever growing difficulty and you could be exploited by hackers.

NordVPN allows you to change your IP address, making you harder to track, securing your privacy. Check out the following link to get 3 months for free if you purchase a 1 year or 2 year plan!

https://ref.nordvpn.com/AVNJcFJOusp

Definition Graphen und Isomorphie

Definition Graph

Ein Graph ist eine netzartige Struktur, um mathematische Modelle in Natur und Technik, z.B. für Strassennetze, Computernetze, elektrische Schaltungen, etc. darzustellen.

Graphen bestehen aus zwei verschiedenen Mengen von Objekten:

- **Knoten**: Orte im Netz
- **Kanten**: Verbindungen zwischen Knoten (z.B. Übertragungsleitungen)

Typische Aufgabe: Kürzester Weg zwischen Knoten.

Ungerichtete Graphen

Ungerichtete Graphen bestehen aus:

- einer Knotenmenge $V = V(G)$ (engl. vertex) mit der Anzahl Knoten $n := |V| < \infty$
- einer Kantenmenge $E = E(G)$ (engl. edge) mitder Anzahl Kanten $m := |E| < \infty$

wobei jeder Kante $e \in E$ zwei (nicht notwendigerweise verschiedene) Knoten aus V zugeordnet sind.

Schreibweise: $e = \{ u, v \} = \{ v, u \}$

Die Knoten u und v heissen dann Endknoten der Kante e. Eine Kante der Form $\{ v, v \}$ heisst Schlinge (oder Loop) und zwei verschiedene Kanten der Form $e = \{ u, v \}$ und $f = \{ u, v \}$ heissen parallel.

Ein Graph, der weder Schlingen noch parallele Kanten besitzt, heisst schlichter Graph.

Knoten- oder Eckengrad

Eine wichtige Definition ist der Grad eines Knotens. Der Grad $deg(v)$ eines Knotens oder einer Ecke $v \in V$ ist gleich der Anzahl Kanten von G, die v enthalten. Schlingen werden deoppelt gezählt.

$$\deg(v) = 0 \longrightarrow v \text{ heisst isolierte Ecke}$$
$$\deg(v) = 1 \longrightarrow v \text{ heisst Endecke}$$

Addiert man die Grade aller Knoten (Ecken) eines Graphen, zählt man alle Kanten des Graphen doppelt. Es gilt also der **Satz (Handshaking Lemma):**

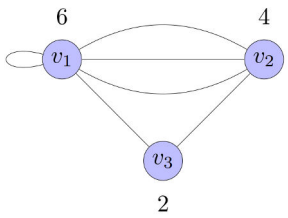
$$\sum_{v \in V} \deg(v) = 2 \cdot |E|$$

Gradliste, minimaler und maximaler Knotengrad

Der Maximalgrad $\triangle(G)$ ist Maximum der Grade aller Knoten von G. Der Minimalgrad $\delta(G)$ dessen Minimum.

Der Maximalgrad: $\triangle(G) = \underbrace{\max}_{v \in V(G)} \deg(v)$

Der Minimalgrad: $\delta(G) = \underbrace{\min}_{v \in V(G)} \deg(v)$



Wichtig zu beachten iost, dass Schlingen doppelt gezählt werden!

Die Gradliste von G sieht wie folgt aus: $(6, 4, 2)$

Hier ist

$$\triangle(G) = 6, \quad \delta(G) = 2$$

Definition isomorphe Graphen

Zwei Graphen $G = (V, E)$ und $G'(V', E')$ heissen isomorph, wenn es eine Bijektion $f : V \to V'$ gibt, die die Kanten erhält, das bedeutet:

$$\{ u, v \} \in E \quad \iff \quad \{ f(u), f(v) \} \in E'$$

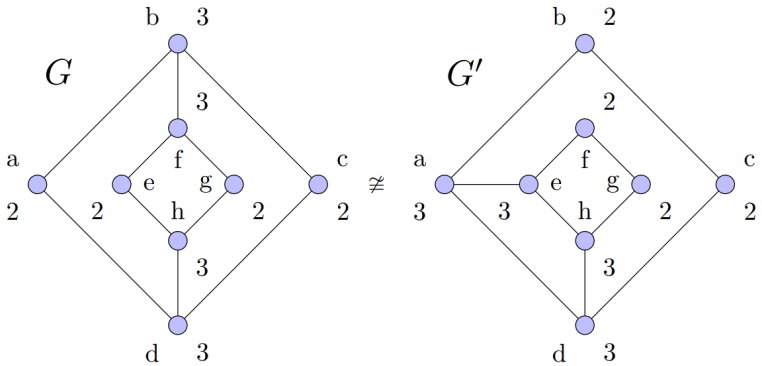
Es gibt also eine bijektive Abbildung, welche die Nachbarschaftsbeziehung beibehält: sind also zwei Knoten u und v in G durch eine Kante verbunden, dann sind auch die Bilder dieser beiden Knoten $f(u)$ und $f(v)$ in G' durch eine Kante verbunden.

***Kurz und einfach**: Isomorphe Graphen sind zwei Graphen, die in ihrer Struktur identisch sind, auch wenn sie unterschiedlich aussehen. Sie haben die gleichen Knoten, die gleichen Beziehungen zwischen den Knoten und sind somit isomorph.*

Isomorphe Graphen haben also gleich viele Knoten und Kanten und deren Gradlisten sind identisch. Die Umkehrung gilt aber nicht!

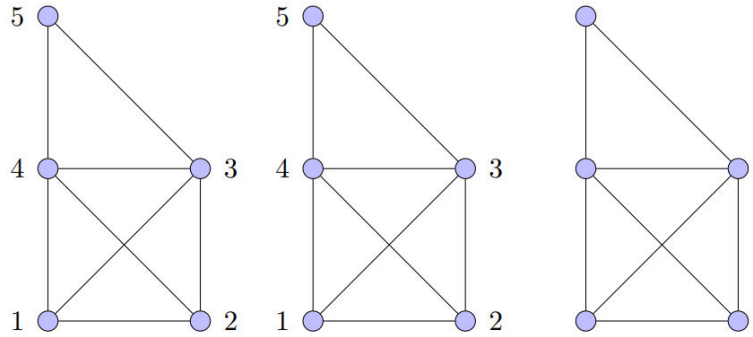
Wenn isomorphe Graphen identifiziert werden, erhält man **Isomorphieklassen** von Graphen. Die Bezeichnung der Ecken spielt in diesen Klassen keine Rolle mehr und man spricht dann auch von **unmarkierten Graphen**.

Gegenbeispiel für zwei nicht isomorphe Graphen mit gleichen Gradlisten:

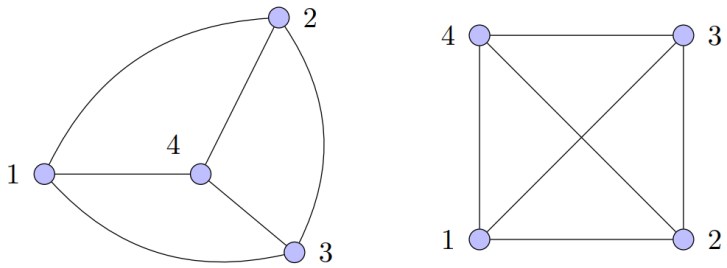


Obwohl die Gradlisten $G = G' = (3, 3, 3, 3, 2, 2, 2, 2)$ sind, sind die Beziehungen nicht gleich und die Graphen G und G' somit nicht isomorph.

Zum Beispiel sind folgende Graphen verschieden, gehören aber zur selben Isomorphieklasse, die durch den unmarkierten Graphen (ganz rechts) repräsentiert wird:



Ein Beispiel: Zeigen SIe, dass die beiden Graphen $G(V, E)$ und $G'(V', E')$ isomorph sind. Geben Sie dazu die bijektive Abbildung $f : V \to V'$ an und kontrollieren Sie für jede Kante $\{ u, v \} \in E$, dass $\{ f(u), f(v) \} \in E'$.



Wobei der linke Graph G sei und der rechte Graph G':

G	G'
1	1
2	3
3	2
4	4

$$\begin{array}{ll} \{1,2\} \to \{1,3\} & \{1,2\} \to \{1,3\} \\ \{1,3\} \to \{1,2\} & \{1,3\} \to \{1,2\} \\ \{1,4\} \to \{1,4\} & \{1,4\} \to \{1,4\} \\ \{2,3\} \to \{3,2\} & \{2,3\} \to \{3,2\} \\ \{2,4\} \to \{3,2\} & \{2,4\} \to \{3,2\} \\ \{3,4\} \to \{2,4\} & \{3,4\} \to \{2,4\} \end{array}$$

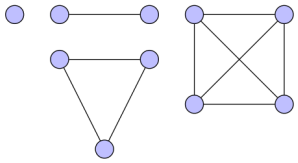
Wichtige Graphen

Vollständige Graphen

Ein vollständiger Graph K_n mit n Knoten besitzt zwischen je zwei Knoten stets genau eine Kante.

Ein vollständiger Graph K_n mit n Knoten besitzt genau:

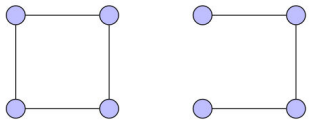
$$m = \binom{n}{2} = \frac{n(n-1)}{2} \quad \text{Kanten.}$$



Wege und Kreise

Der **Weg** P_n (engl. path) besitzt die Knotenmenge $\{1, 2, \ldots, n\}$ und die Kantenmenge $\{\{1,2\}, \{2,3\}, \ldots, \{n-1, n\}\}$.

Ein **Kreis** C_n (engl. cycle) besitzt die Knotenmenge $\{1, 2, \ldots, n\}$ und die Kantenmenge $\{\{1,2\}, \{2,3\}, \ldots, \{n-1, n\}, \{n, 1\}\}$



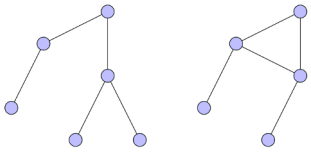
Links: Kreis, Rechts: Weg

Ein **Kreis** ist also *ein geschlossener Weg*, d.h. der letzte Knoten ist gleich dem ersten Knoten.

Bäume

Ein Baum (Bezeichnung T_n (engl. tree) für einen Baum mit n Knoten) ist ein zusammenhängender Graph, der keinen Kreis enthält! Man kann leicht zeigen, dass ein Graph genau dann ein Baum ist, wenn es zwischen zwei Knoten genau einen Weg gibt.

Im rechten der Graphen, kann man schnell erkennen, dass dies kein Baum ist.

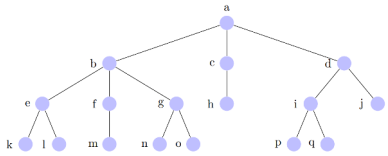


Links: Baum, Rechts: kein Baum

Dies, da dieser einen Kreis hat, somit gibt es nicht für jeden Knoten nur einen Weg sondern mehrere Wege. Der Linke Graph hingegen ist eine gültige Baumstruktur.

Ein **Wurzelbaum** ist ein Baum in dem ein Knoten als Wurzel fest gewählt wurde. Die Höhe eines Knotens ist dann die Länge des Weges von der Wurzel zu diesem Knoten.

a ist hier die **Wurzel**; c ist der **Vorgänger** oder **Vater** von h; g ist der **Nachfolder** oder **Sohn** von b; e, f, und g sind **Brüder**; Knoten mit Söhnen heissen **innere Knoten** und Knoten ohne Söhne heissen **Blätter;** a befindet sich auf der **Höhe 0**; b, c, d befinden sich auf der Höhe 1, usw...



Ein Baum mit 17 Knoten und der Höhe 3 (0 bis und mit 3)

Ein Wurzelbaum heisst **m-facher Baum**, wenn jede innere Ecke höchstens m Söhne hat. Ein voller m-facher Baum hat stets genau m Söhne an jeder innerer Ecke. **Ein 2-facher Baum heisst Binärbaum.**

Eigenschaften von Bäumen:

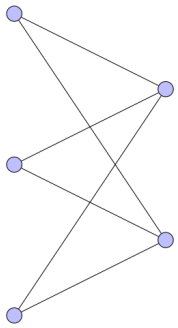
- Ein Baum mit n Knoten hat $n-1$ Kanten.
- Ein voller m-facher Baum mit i inneren Knoten hat $n = m \cdot i + 1$ Knoten.
- Ein m-facher Baum der Höhe h hat höchstens m^h Blätter.

Vollständige bipartite Graphen

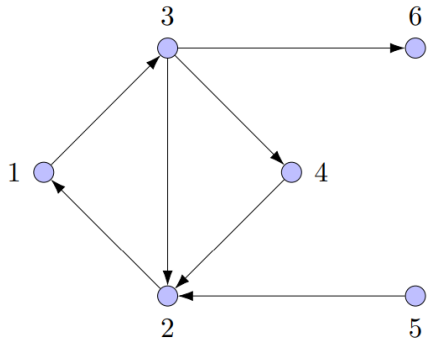
Ein vollständiger bipartiter Graph $G = (V, E)$ besitzt eine Knotenmenge $V = U \cap W$ mit $U \cap W = \emptyset$, so dass folgendes gilt:

- es gibt keine Kante zwischen Knoten aus U
- es gibt keine Kante zwischen Knoten aus W
- jeder Knoten aus U ist mit dem Knoten aus W durch genau eine Kante verbunden

Bezeichnung $K_{p,q}$ falls $|U| = p$ und $|W| = q$



Page-Rank Algorithmus (verbesserte Variante)



Dies entspricht einer Eigenwertgleichung:

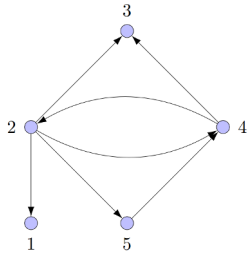
$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/3 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/3 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} PR_1 \\ PR_2 \\ PR_3 \\ PR_4 \\ PR_5 \\ PR_6 \end{bmatrix} = \begin{bmatrix} PR_1 \\ PR_2 \\ PR_3 \\ PR_4 \\ PR_5 \\ PR_6 \end{bmatrix}$$

$$PR_i = d \cdot \sum_{\text{j hat Link auf i}} \frac{PR_j}{C_j} + (1-d) \cdot \frac{1}{N}, \quad i = 1, 2, \ldots, N$$

Page-Rank Algorithmus (verbessert) (an einem Beispiel)

Folgende Aufgabenstellung:

Berechnen Sie die PageRank's $PR_1, \ldots, PR_5$ für das folgende Netzwerk mit dem Dämpfungsfaktor d = 4/5 (80% folgt man den Links, 20% ruft man zufällig eine der 5 Seiten auf). Stellen Sie dazu zunächst das lineare Gleichungssystem zur Bestimmung der PageRanks auf. Zur Lösung des Systems kann ein CAS, Python, etc verwendet werden.



Zuerst erstellen wir die Matrix, bei der wir die Wahrscheinlichkeiten über die Links beschreiben:

Wir schauen auf einen Knoten x und wie viele Kanten k_x von diesem Knoten aus führen. Bei dem Knoten 2 zum Beispiel, haben wir 4 Kanten, die vom Knoten wegführen. Also ist in der Matrix für jede Beziehung von diesem Knoten aus die Wahrscheinlichkeit $1/k_x$ also hier $1/4$ zu ergänzen. Dies macht man für die ganze Matrix und erhält folgendes Resultat:

$$A = \begin{bmatrix} 0 & 1/4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/2 & 0 \\ 0 & 1/4 & 0 & 1/2 & 0 \\ 0 & 1/4 & 0 & 0 & 1 \\ 0 & 1/4 & 0 & 0 & 0 \end{bmatrix} \qquad r_0 = \begin{bmatrix} 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \end{bmatrix}$$

Mithilfe dieser Matrix A und dem Dämpfungsfaktor d erstellen wir das Gleichungssystem, in dem wir folgendes berechnen: $d \cdot A + (1-d) \cdot r_0$, wobei r_0 der Eigenvektor ist. Man erhält dann:

$$\begin{aligned} PR_1 &= \frac{4}{5}(\frac{1}{4}PR_2) + \frac{1}{5} \cdot \frac{1}{5} \\ PR_2 &= \frac{4}{5}(\frac{1}{2}PR_4) + \frac{1}{5} \cdot \frac{1}{5} \\ PR_3 &= \frac{4}{5}(\frac{1}{4}PR_2 + \frac{1}{2}PR_4) + \frac{1}{5} \cdot \frac{1}{5} \\ PR_4 &= \frac{4}{5}(\frac{1}{4}PR_2 + PR_5) + \frac{1}{5} \cdot \frac{1}{5} \\ PR_5 &= \frac{4}{5}(\frac{1}{4}PR_2) + \frac{1}{5} \cdot \frac{1}{5} \end{aligned}$$

So könnten wir dies jetzt schon lösen. Teilweise jedoch, wollen wir das Ganze in der Matrizenschreibweise:

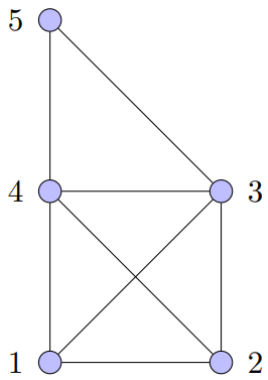
$$\begin{bmatrix} 0 & 4/20 & 0 & 0 & 0 \\ 0 & 0 & 0 & 8/20 & 0 \\ 0 & 4/20 & 0 & 8/20 & 0 \\ 0 & 4/20 & 0 & 0 & 16/20 \\ 0 & 4/20 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} PR_1 \\ PR_2 \\ PR_3 \\ PR_4 \\ PR_5 \end{bmatrix} + \frac{1}{5} \begin{bmatrix} 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \end{bmatrix}$$

Graphen und Matrizen

Adjazenzmatrix

Die Adjazenzmatrix $A(G)$ des Graphen G mit n Knoten ist die $n \times n$ Matrix mit den Elementen:

$$A_{ij} := \text{Anz. der Kanten zwischen Knoten i und Knoten j}$$



$$A(G) = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

Wenn der Graph G ungerichtet ist, dann ist $A(G)$ symmetrisch.

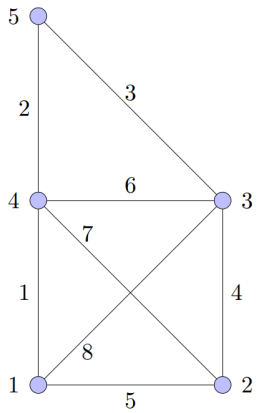
Die Adjazenzmatrix verrät uns an der Stelle A_{ij} wie viele Kanten es zwischen i und j gibt: $A_{2,3} = 1$.

In diesem einfachen Beispiel ist der Wert in der Matrix immer 1 oder 0, da wir entweder keine oder eine Kante zwischen zwei Knoten haben und nicht mehr. Dies kann aber auch anders sein.

Inzidenzmatrix

Die Inzidenzmatrix $B(G)$ ist die $n \times m$ Matrix mit den Komponenten:

$$B_{ij} := x = \begin{cases} 1 & \text{falls Knoten i auf Kante j liegt} \\ 0 & \text{sonst} \end{cases}$$

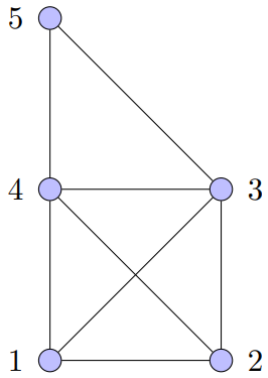


$$A(G) = \begin{bmatrix} \overbrace{\begin{matrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{matrix}}^{Kanten} \end{bmatrix}$$

Diese Matrix zeigt uns für jede Kante (Horizontal) eine 1, wenn sie mit dem Punkt (Vertikal) verbunden ist und sonst eine 0.

Gradmatrix

Die Gradmatrix $D(X)$ des Graphen G ist die $n \times n$ Diagonalmatrix, deren Diagonaleinträge die Grade der entsprechenden Knoten von G sind.



$$D(G) = \begin{bmatrix} 3 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 2 \end{bmatrix}$$

Wir sehen zum Beispiel für den Knoten 1, in der Matrix an Position (1,1), haben wir den Grad 3. Dies ist so für jeden Knoten hinterlegt. Der Rest der Matrix bleibt leer (0).

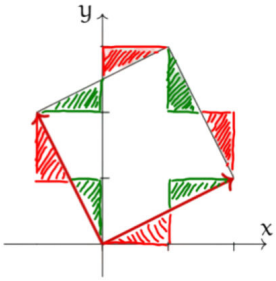
Admittanzmatrix (Laplace-Matrix)

Die Differenz aus der Gradmatrix $D(G)$ und Adjazenzmatrix $A(G)$ eines Graphen G:

$$L(G) = D(G) - A(G)$$

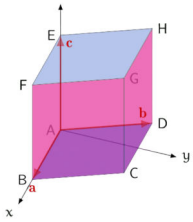
heisst Admittanzmatrix (oder Laplace-Matrix).

Determinante einer Matrix



Bei einer 2x2 Matrix ist die Determinante die Fläche des durch die beiden Vektoren aufgespannten Paralellogramms.

Bei einer 3x3 Matrix entspricht die Determinante des Volumens des Parallelepipeds.



Determinante einer **zweireihigen** Matrix:

$$A = \begin{bmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{bmatrix}, \qquad det(A) = A_{1,1}A_{2,2} - A_{1,2}A_{2,1}$$

Determinante einer **dreireihigen** Matrix:

$$A = \begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} \\ A_{2,1} & A_{2,2} & A_{2,3} \\ A_{3,1} & A_{3,2} & A_{3,3} \end{bmatrix}, \qquad \det(A) = A_{1,1}A_{2,2}A_{3,3} + A_{1,2}A_{2,3}A_{3,1} + A_{1,3}A_{2,1}A_{3,2} - A_{3,1}A_{2,2}A_{1,3} - A_{3,2}A_{2,3}A_{1,1} - A_{3,3}A_{2,1}A_{1,2}$$

Länge von Wegen und Kreisen

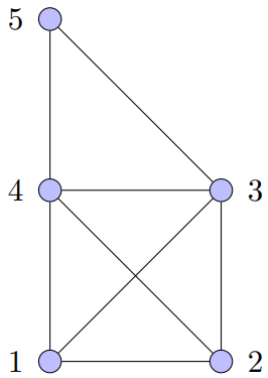
Ein Weg in einem Graphen ist der Pfad von einem Knoten zu einem anderen Knoten über alle Kanten des Graphen, ohne dabei die gleiche Kante 2-Mal zu besuchen.

Ein Kreis in einem Graphen hingegen, ist der Weg von einem Knoten über andere Knoten zurück zu sich selbst, dabei muss jede Kante durchlaufen werden ohne dabei Kanten 2-Mal zu besuchen.

Anzahl Wege zwischen Knoten

Um die Anzahl Wege in einem Graphen von einem Knoten i zu einem Knoten j zu berechnen, erstellen wir die Adjazenzmatrix dieses Graphen. Wenn wir alle Wege der Länge n vom Knoten i zum Knoten j berechnen wollen, müssen wir die Matrix n-Mal mit sich selbst multiplizieren. In der resultierenden Matrix an der Stelle $A^n_{i,j}$ befindet sich die Anzahl Wege der Länge n von i zu j.

Zum Beispiel die Anzahl Wege Länge $n = 3$ von Knoten 1 (i) zum Knoten 3 (j)?

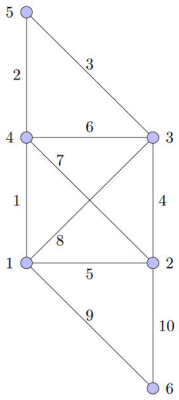


$$A(G) = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix} \quad A^3 = \begin{bmatrix} 6 & 7 & 9 & 9 & 4 \\ 7 & 6 & 9 & 9 & 4 \\ 9 & 9 & 8 & 9 & 7 \\ 9 & 9 & 9 & 8 & 7 \\ 4 & 4 & 7 & 7 & 2 \end{bmatrix}$$

Also ist die Antwort: **9**

Es führen 9 Wege der Länge 3 vom Knoten 1 nach Knoten 3

Eulerweg und Eulerkreise



Eulerweg

Ein Eulerweg (bzw. ein Eulerkreis) in einem Graphen G ist ein Weg (bzw. ein Kreis), der jede Kante von G genau einmal durchläuft.

In dieser Abbildung links ist kein Eulerweg möglich, weil:

Mehr als zwei Knoten haben einen ungeraden Grad.

Mit einem Euler-Weg wäre man alle Kanten durchlaufen, endet aber nicht beim Startknoten.

Eulerkreis

Ein möglicher Eulerkreis in diesem Graphen wäre:

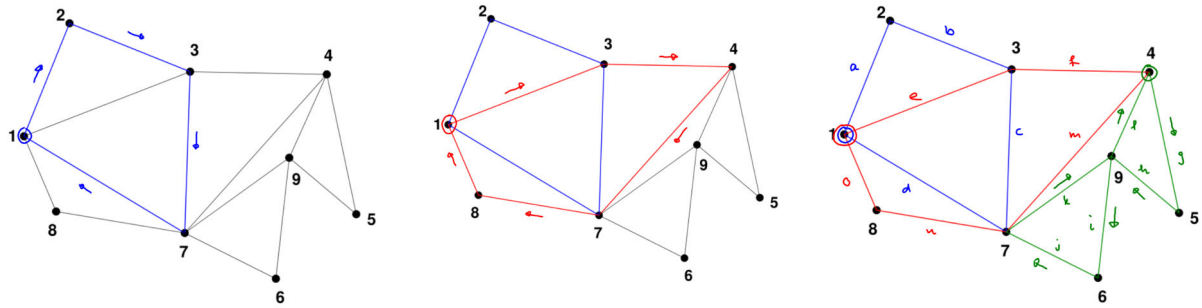
$3, 2, 6, 1, 2, 4, 1, 3, 4, 5, 3$

Mit diesem Weg startet und endet man bei 3. Somit ist dies ein Eulerkreis.

***Satz**: Jeder zusammenhängende Graph mit mindestens zwei Ecken hat genau dann einen Eulerkreis, wenn alle Knoten einen geraden Grad haben.*

Konstruktion von Eulerkreisen

1. Wähle einen beliebigen Knoten a von G, damit hat a einen ungeraden (Rest)grad.
2. Hänge immer neue Kanten an (aber keine doppelt!).
3. Jeder solche Weg muss in a enden, denn jeder Knoten (bis auf a) hat geraden Grad, kann also wieder verlassen werden.
   a. 1. Möglichkeit: Jede Kante wurde durchlaufen: Fertig.
   b. 2. Möglichkeit: Nicht jede Kante wurde durchlaufen. Somit muss man einen Knoten auf dem restlichen Graphen wählen, der besucht wurde, jedoch noch Kanten hat, die in keinem Weg (Teilkreis) verwendet wurden. Von dort aus macht man einen eigenen Kreis.
4. Nachdem alle Teilwege bestimmt sind, geht es darum, diese zusammenzusetzen. Sobald man einem Knoten begegnet, der einer der Startpunkte der neuer Wege ist, geht man zuerst diesen Weg durch, bevor man den vorherigen Weg fortsetzt. So durchläuft man jeder seiner Teilkreise und somit den ganzen Graphen.

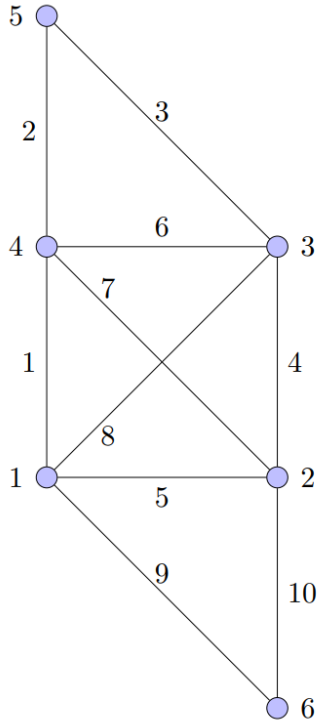


Hier würde man jetzt den Eulerkreis folgendermassen bilden:

$1, 2, 3, 7, 1, 3, 4, 5, 9, 6, 7, 9, 4, 7, 8$

Hamiltonweg und Hamiltonkreis

Ein Hamiltonweg (bzw. Hamiltonkreis) in einem Graphen G ist ein einfacher Weg (bzw. ein einfacher Kreis), der jeden Knoten von G genau einmal durchläuft.



Der Hamiltonweg

Der Hamiltonweg besucht alle Knoten, ohne dabei einen doppelt zu besuchen, und ohne dabei am Startpunkt zu enden.

$1, 6, 2, 4, 3, 5$ ist ein Hamiltonweg.

Jeder Knoten kommt genau einmal vor, der Startknoten ist 1 und der Endknoten ist 5.

Der Hamiltonkreis

Der Hamiltonkreis besucht alle Knoten, ohne dabei einen doppelt zu besuchen, mit der Bedingung, dass der Kreis dort endet wo er auch startet (wie bereits bekannt aus Kreisen in Graphen generell):

$1, 6, 2, 3, 5, 4, 1$ ist ein Hamiltonkreis.

Jeder Knoten kommt genau einmal vor (ausser Start-/Endknoten) und die Bedingung, dass kein Knoten doppelt besucht wird stimmt auch.

Planare Graphen

Ein Graph ist planar, wenn er sich in der Ebene ohne Kantenkreuzung zeichnen lässt. Wenn ein Graph also ohne Kantenkreuzung gezeichnet werden KANN, ist er planar.

Typisch trifft man dies bei z.B. dem Versorungsproblem an. Wobei man z.B. Strom, Wasser und Gas zu verschiedenen Gebäuden bringen muss, ohne, dass sich die Leitungen schneiden.

Dieses Problem ist für grosse Graphen sehr schwierig zu lösen, da man ausprobieren muss.

Satz von Kuratovsky

Ein Graph ist genau dann nicht planar, wenn er einen Untergraphen vom Typ $K_{3,3}$ oder K_5 enthält. Somit sind sicher $K_{3,3}$ oder K_5 nicht planar!

Der Satz sagt also: G nicht planar $\iff$ G enthält $K_{3,3}$ oder K_5 als Untergraphen

Eulerscher Polyedersatz

Der Eulersche Polyedersatz beschreibt die Beziehung zwischen Knoten, Kanten und Flächen in allen konvexen Polyeder und auch für Polyedernetze dessen Graphen planar sind (sich also kreuzungsfrei auf einer Ebene darstellen lassen).

Der Satz lautet:

$$\begin{gathered}\text{Knoten} - \text{Kanten} + \text{Flächen} = |V| - |E| + |F| = 2 \\ \text{und somit auch: } |E| = |V| + |F| - 2\end{gathered}$$

> ⚠ **ACHTUNG**! Auch die Fläche ausserhalb der Kanten des Graphen wird als Fläche gezählt!

Eulers Antwort zum Utilities Problem (Versorungsproblem):

In jedem planaren Graphen gilt:

$$\begin{aligned} 2 \cdot |E| &= \text{Flächen mit 1 Kante} \cdot 1 && (1) \\ &+ \text{Flächen mit 2 Kanten} \cdot 2 && (2) \\ &\vdots && (3) \end{aligned}$$

Also zusammengefasst:

$$\sum_{i \geq K} \text{Flächen mit i Kanten} \cdot i$$

Färbungen

Definition

Unter einer Färbung (der Ecken) eines Graphen $G = (V, E)$ versteht man eine Abbildung $c : V \longrightarrow C$ so, dass $c(u) \neq c(v)$, falls $\{ u, v \} \in E$.

Die kleinste Zahl der Farben, die eine Färbung von G ermöglicht, wird chromatische Zahl von G, kurz $\chi(G)$, genannt.

Es gelten die folgenden beiden Abschätzungen für die chromatische Zahl eines Graphen G:

$$1 \leq \chi(G) \leq \triangle(G) + 1$$

Der Algorithmus

Das Färben von Graphen wird wie folgt gemacht: Als Farben wählen wir die Zahlen 1 bis $\triangle(G) + 1$.

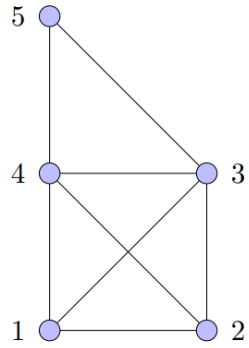
- Wir färben die Knoten $v_1, v_2, v_3, \ldots$ nacheinander, zunächst v_1 mit der Farbe 1, ...
- Wenn wir zu irgendeinem Knoten v_i kommen, färben wir ihn mit der kleinsten noch erlaubten Farbe.
- Wieviele Farben können verboten sein? Schlimmstenfalls
  - ist v_1 eine Ecke maximalen Grades $\triangle(G)$
  - alle $\triangle(G)$ Nachbarn von v_i sind bereits gefärbt
  - alle $\triangle(G)$ Nachbarn von v_i sind verschieden gefärbt.
- Dann sind $\triangle(G)$ Farben verboten, aber $\triangle(G) + 1$ genügen!

Bemerkungen

Der beschriebene Algorithmus ist ein Greedy-Algorithmus. Das bedeutet, dass der Algorithmus immer das Nächstbeste nimmt ("frisst"), ohne dabei einen globalen Plan für "Nahrungsaufnahme" zu haben.

Theorem

Ein ebener Graph (planarer Graph) kann mit vier Farben gefärbt werden, d.h. die chromatische Zahl eines ebenen (planaren) Graphen ist nicht grösser als vier.



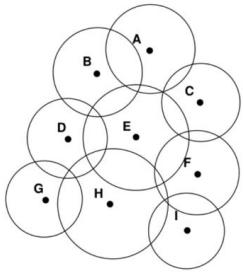
Die Punkte 1, 2, 3, 4 müssen alle verschiedene Farben tragen, da jeder mit den drei restlichen verbunden ist, also gilt $\chi(G) \leq 4$. Ausserdem stellt das linke Bild sicher eine Färbung mit 4 Farben dar.

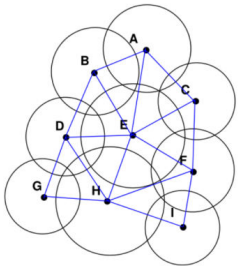
$\chi(G) = 4, \quad \triangle(G) = 4$

Beispielanwendung: Frequenzplanung in Funknetzen

Aus dem Funknetz mit den Masten A bis I, sind auch die überlappenden Zonen eingezeichnet. Hierbei wollen wir jetzt die Frequenzen so verteilen, dass sich die Funknetze nicht gegenseitig stören.

Dafür können wir aus den Funkmasten einen Graphen zeichnen.





Dieser Graph sieht dann so aus (links).

Hier bestimmen wir jetzt den chromatischen Wert

1. Maximalgrad von G : $\chi(G) \leq \triangle(G) = 6$
2. Weil ein k_3 in G enthalten ist: $\chi(G) \geq 3$

Somit ist $3 \leq \chi(G) \leq 7$

Die genaue Berechnung ist nicht so einfach.

Das chromatische Polynom

Sei $G = (V, E)$ ein Graph. Das chromatische Polynom $P(G, x)$ gibt für jedes $x \in \mathbb{N}$ die Anzahl der zulässigen Färbungen (d.h. Knoten auf einer Kante müssen verschieden gefärbt sein) von G mit höchstens x Farben.

Mit dem chromatischen Polynom kann man die Anzahl der möglichen Färbungen eines Graphen berechnen.

- Das chromatische Polynom $P(G, x)$ ist eine Funktion, die für einen Graphen G definiert wird. Das Polynom gibt an, wie viele verschiedene Arten es gibt, die Knoten des Graphen mit x Farben zu färben, so dass keine zwei benachbarte KNoten die gleiche Farbe haben.
- x : Dies ist eine Variable, die für die Anzahl der Farben steht, die wir zur Verfügung haben.
- $P(G, 2), P(G, 3), P(G, n)$: Diese sind Beispiele für das chromatische Polynom, wobei die Zahl nach dem G die Anzahl der Farben angibt. $P(G, 2)$ bedeutet zum Beispiel, dass wir berechnen, wie viele Möglichkeiten es gibt, den Graphen G mit genau zwei Farben zu färben, sodass benachbarte Knoten unterschiedliche Farben haben.

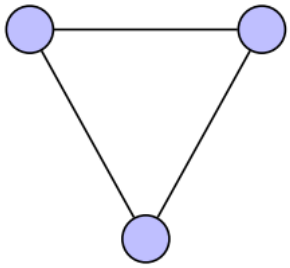
Beispiele für einen Graphen mit 2 Knoten:

- Für $P(G, 2) = 2^2 = 4$ Möglichkeiten
- Für $P(G, 3) = 3^2 = 9$ Möglichkeiten

Sei G der Graph mit n Knoten und der leeren Kantenmenge, dann gilt $P(G, x) = x^n$

Sei K_n der vollständige Graph mit n Knoten. Dann gilt $P(K_n, x) = x \cdot (x-1) \dots (x-n+1)$
Dies, weil ja jeder Knoten mit jedem Verbunden ist, und somit jedesmal eine Farbe von x abgezogen wird.

Beispiel: Für das chromatische Polynom von K_3 gilt $P(K_3, x) = x \cdot (x-1) \cdot (x-2)$



- $P(K_3, 1) = 0$ bedeutet, dass der Graph mit einer Farbe nicht gefärbt werden kann,
- $P(K_2, 1) = 0$ bedeutet, dass der Graph auch mit zwei Farben nicht gefärbt werden kann,
- $P(K_3, 3) = 3 \cdot 2 \cdot 1 = 6$ bedeutet, dass K_3 mit 3 Farben auf 6 verschiedene Arten gefärbt werden kann.

Chromatisches Polynom eines Baumes

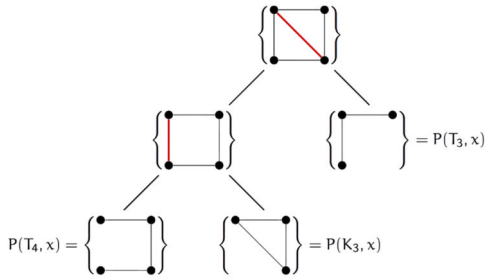
Sei T_n ein Baum mit n Knoten. Dann gilt $P(T_n, x) = x \cdot (x-1)^{n-1}$

Dekompositionsgleichung

Satz (Dekompositionsgleichung), es gilt:

$$P(G-e,x) = P(G,x) + P(G_e,x) \text{ oder } P(G,x) = P(G-e.x) - P(G_e,x)$$

Mithilfe der Dekompositionsgleichung erstellt man das chromatische Polynom eines komplexen Graphen. Dies macht man, in dem man den komplexen Graphen in Untergraphen unterteilt, wobei man in jedem Schritt Kanten entfernt und Knoten verschmilzt. Am besten gezeigt an folgendem Beispiel:



Schritt 1: Wir wollen im Originalgraphen die rote Kante, nennen wir sie e, zuerst entfernen.

Dazu entfernt man einmal die Kante komplett (links). Dieser Teilgraph heisst nun: $G' = G - e$

Zusätzlich verschmelzen wir noch die Knoten dieser Kante e zu einem Knoten (rechts). Knoten von unten rechts und oben links werden zu einem → zu dem oben links. Dieser Teilgraph wie man sieht hat nun eine Baumstruktur. Für diesen Graphen können wir das chromatische Polynom bereits aufstellen. Dies mithilfe der Formel für chromatische Polynome in Baumstrukturen:

$P(T_n, x) = x \cdot (x-1)^{n-1}$, hier also: $x \cdot (x-1)^{3-1} = x(x-1)^2$

Schritt 2: Wir wollen für G', da wir diesen noch nicht berechnen können, dies nochmals vereinfachen, in dem wir für die Kante f (rote Kante in 2. Reihe) das gleiche nochmals machen wie bei e.

Dazu entfernt man einmal die Kante komplett (links). Dieser Teilgraph, $G' - f$, ist nun ebenfalls eine Baumstruktur und somit berechenbar durch:

$P(T_n, x) = x \cdot (x-1)^{n-1}$, hier also: $x \cdot (x-1)^{4-1} = x(x-1)^3$

Zusätzlich verschmelzen wir die Knoten der Kante f auch noch (rechts). Hier erhalten wir einen Kreis. Auch für diesen können wir das chromatische Polynom berechnen mittels:

$P(K_n, x) = x \cdot (x-1) \cdot \cdots \cdot (x-n+1)$, hier also $x(x-1)(x-2)$

n sind hier die Anzahl Knoten im Teilgraphen

Schritt 3: Wir haben jetzt alle Teilgraphen und deren chromatische Polynome bestimmt.

Für $P(G,x)$ haben wir jetzt also $P(T_4,x) - P(K_3,x) - P(T_3,x)$ und setzen jetzt die einzelnen Komponenten für diese Teilgraphen ein, was dann folgendes ergibt:

$$\begin{aligned} P(G,x) &= P(T_4,x) - P(K_3,x) - P(T_3,x) \\ &= x(x-1)^3 - x(x-1)(x-2) - x(x-1)^2 \\ &= x(x-1)\left[(x-1)^2 - (x-2) - (x-1)\right] \\ &= x(x-1)\left[x^2 - 2x + 1 - x + 2 - x + 1\right] \\ &= x(x-1)\left[x^2 - 4x + 4\right] \\ &= x(x-1)(x-2)^2. \end{aligned}$$

Das chromatische Polynom liefert somit einen systematische Möglichkeit zur Berechnung der chroamtischen Zahl eines Graphen: $\chi(G) = \min\{\, x \in \mathbb{N} \mid P(G,x) > 0 \,\}$

Nun hat man also für den obigen Graphen G:

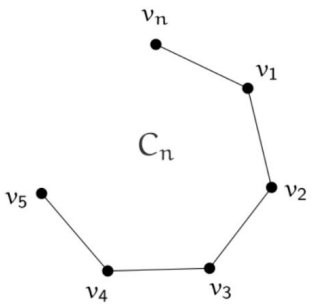
- $P(G,2) = 2(2-1)(2-2)^2 = 0$, somit keine Möglichkeit den Graphen mit nur 2 Farben zu Färben
- $P(G,3) = 3(3-1)(3-2)^2 = 6$, somit 6 Möglichkeiten zur Färbung von G mit 3 Farben

Somit weiss man auch, dass $\chi(G) = 3$.

Chromatische Polynom eines Kreises

Das chromatische Polynom $P(C_n,x)$ eines Kreises C_n mit $n \geq 1$ Knoten ist gegeben durch

$P(C_n,x) = (x-1)^n + (-1)^n(x-1)$



Nach obigem Satz gilt:

$$\begin{aligned} P(C_3,x) &= (x-1)^3 + (-1)^3(x-1) \\ &= (x-1)\left[(x-1)^2 - 1\right] \\ &= (x-1)\left[x^2 - 2x + 1 - 1\right] \\ &= (x-1)\left[x^2 - 2x\right] \\ &= x(x-1)(x-2) \end{aligned}$$

Das ist gerade das chromatische Polynom von K_3 des obigen Beispiels. Was zu erwarten war, weil ja $C_3 = K_3$

Kreise mit Schlingen

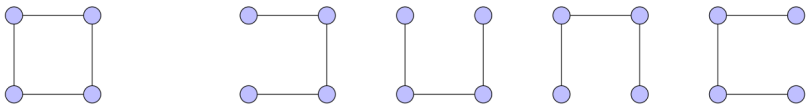
- Die chromatische Zahl von C_1 ist nicht definiert, denn C_1 lässt sich nicht färben. Ganz allgemein lässt sich ein Graph mit Schlingen nicht färben!
- Dies ist generell so, das zwei benachbarte Knoten unterschiedlich gefärbt werden müssen. Da der Knoten einer Schlinge zu sich selbst benachbart ist, kann dieser nicht 2 Farben gleichzeitig haben und somit auch nicht gefärbt werden.
- Das Einfügen einer parallelen Kante ändert die Färbung eines Graphen nicht.
- Wir färben ausschlisslich schlichte Graphen, d.h. Graphen ohne Schlingen und parallele Kanten. Falls der Graph nicht schlicht ist, entfernen wir vor der Färbung alle Schlingen und parallelen Kanten.

Gerüste / Spannbäume

Ein Gerüst (oder auch Spannbaum genannt) eines Graphen $G = (V, E)$ ist ein zusammenhängender kreisfreier Untergraph (Baum), der alle Knoten aus V enthält.

Wie viele Grüste hat ein Graph?

- Ein Baum besitzt genau ein Gerüst
- Für einen Kreis mit n Kanten erhalten wir jedes Gerüst durch Entfernen von genau einer Kante. Folglich besitzt ein Kreis mit n Kanten genau n Gerüste.



Graph (links), Gerüste des Graphen (rechts)

Es ist nicht direkt einfach die Anzahl aller Gerüste $t(G)$ zu bestimmen, aber es ist einfach einen rekursiven Algorithmus zur Bestimmung von $t(G)$ anzugeben.

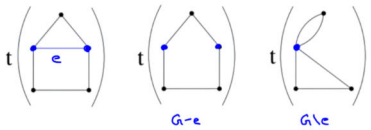
Dazu braucht man weitere zwei Graphenoperationen:

- $G - e$ der Graph, der aus G durch Weglassen der Kante e entsteht
- G/e der Graph, der aus G durch Zusammenziehen der Kante e und Weglassen aller Schlingen entsteht. Parallele Kanten werden nicht enfernt!

Mit diesen Operationen gilt folgendes Theorem:

Für die Anzahl Gerüste $t(G)$ des Graphen G gilt:

$$t(G) = t(G - e) + t(G/e)$$



Auch hier wird die Dekompositionsgleichung solange angewendet, bis nur noch Kreise oder Bäume verbleiben, und dann wird dies in die Gleichung eingesetzt.

Anwendung: Bäume, Spiele und Strategien

Ein Spiel kann unter folgenden Bedingungen durch enen Baum dargestellt werden:

- Es sind nur endlich viele Spielstellungen möglich
- Es gibt keine Züge, die zu bereits vorgekommenen Stellungen führen.
- Es handelt sich um ein Spiel mit vollständiger Information, d.h. die Spieler kennen zu jedem Zeitpunkt die vollständige Spielsituation
- Die Spielzüge sind nicht vom Zufall bestimmt.

Ein wichtiger Satz der Spieltheorie besagt:

Wenn sich beide Spieler rational verhalten, d.h. das für sie jeweils beste tun, dann ist der Ausgang eines solchen Spieles immer derselbe.

Minimax-Algorithmus

Beim Minimax-Algorithmus arbeitet man sich in einem Baum eines Spiels von den Blättern zur Wurzel. Dabei überlegt man sich immer, welcher Spieler wann dran ist und welche Optionen für ihn die günstigste ist.

Für den optimalen Spielverlauf, kann man sich bei jedem Entscheidungspunkt überlegen, welcher dem Idealfall entspricht. Die anderen kann man dann verwerfen. So bestimmt man den optimalen Spielverlauf.

Ist das gewünschte Ergebnis am Schluss in der Wurzel:

- gut für Spieler 1, hat dieser eine Gewinnstrategie
- gut für Spieler 2, hat dieser eine Gewinnstrategie
- gleich einem Wert der für ein Untentschieden spricht, können beide Spieler ein Unentschieden erzwingen.

Graphalgorithmen

Problem: Ein Netzwerk hat (vollständiger Graph K_{100}) genau $\frac{100 \cdot 99}{2} = 4950$ Kanten und zwischen den Rechnern X und Y gibt es mindestens $98! = 10^{154}$ verschiedene Wege! Würde jemand pro Sekunde einen Weg anschauen können, würde das Betrachten aller Wege mindestens

$\frac{10^{154}}{3600 \cdot 24 \cdot 365} = 3.2 \cdot 10^{146}$ Jahre dauern.

Ziel: Wir wollen eine verbesserte Strategie kennenlernen, die das Problem des Auffindens eines kürzesten Weges effizienter löst als Durchprobieren aller Wege.

Gewichtete Graphen

Ein gewichteter Graph $G = (V, E, w)$ wird durch einen zusammenhängenden Graphen G und eine Gewichtsfunktion w beschrieben:

$$w : E \longrightarrow (0, \infty), \{ u, v \} \mapsto w(u, v)$$

Sie ordnet jeder Kante $e = \{ u, v \}$ ihr Gewicht $w(\{ u, v \})$ zu. Aus Bequemlichkeit schreibt man dafür einfach $w(u, v)$.

Länge oder Abstand in gewichteten Graphen

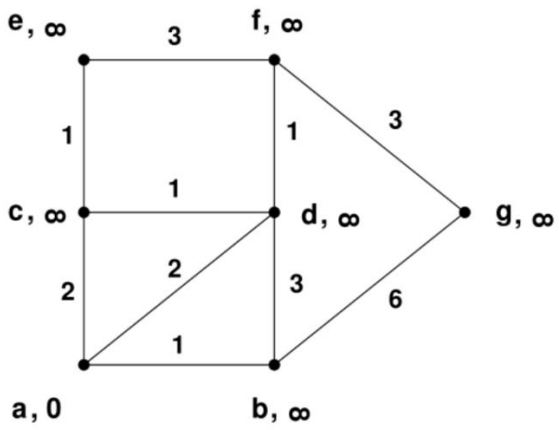
Die Länge oder das Gewicht eines Weges in G ist die Summe der Gewichte sämtlicher Kanten des Weges: $w(u_0, u_1, u_n) = w(u_0, u_1) + w(u_1, u_2) + \cdots + w(u_{n-1}, u_n)$

Der Abstand $d(u, v)$ zwischen zwei Knoten $u, v \in V$ ist das Minimum der Längen aller Wege von u nach v. Ein Weg der Länge $d(u, v)$ von u nach v wird kürzester Weg genannt (es kann mehrere kürzeste Wege geben).

Um den Abstand zwischen zwei Punkten zu bestimmen: $\binom{\text{Knoten}}{2} = \text{Anz. Abstände}$

Der Algorithmus von Dijkstra

Man geht vom Startpunkt a aus, und setzt initial für alle Längen unendlich, da man sie noch nicht kennt. Nun geht man wie folgt vor:

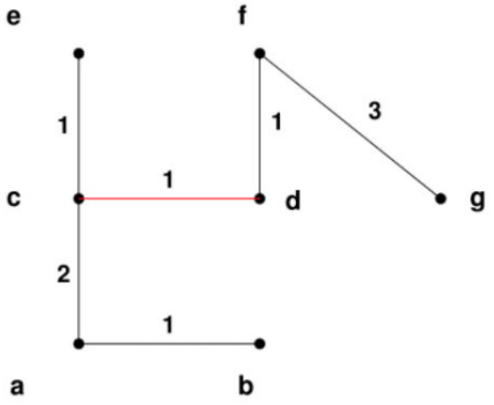


- Man überprüft beim aktuellen Knoten alle Nachbarsknoten, und schreibt die Länge zu ihnen in der Tabelle auf
- Nun geht man zum nächsten (billigsten) Knoten (des ganzen Graphen) und macht dort das gleiche. Wenn wir einen kürzeren Weg zu einem bereits bekannten Knoten finden, ersetzen wir diesen, sonst nicht.
- Dies macht man, bis man alle Knoten durch hat.

Dabei entsteht dann folgende Tabelle:

$L(a)$	0						
L(b) p(b)	∞	1 a					
L(c) p(c)	∞	2 a	2 a				
L(d) p(d)	∞	2 a	2 a	2 a			
L(e) p(e)	∞	∞	∞	∞	3 c		
L(f) p(f)	∞	∞	∞	∞	3 d	3 d	
L(g) p(g)	∞	∞	7 b	7 b	7 b	7 b	6 b
S	a	b	c	d	e	f	g

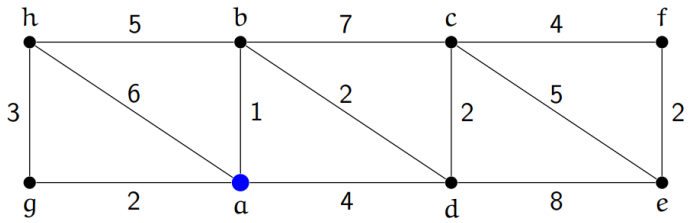
Und daraus dann folgender Spannbaum der kürzesten Wege:



Hier sieht man jetzt nur die kürzesten Wege zu jedem Knoten vom Startknoten a aus.

Der Algorithmus von Prim

Der Algorithmus von Prim findet den minimalen Spannbaum in einem zusammenhängenden, gewichteten Graphen, indem er schrittweise die kürzesten Verbindungen zu bereits angeschlossenen Knoten hinzufügt. Dabei beginnt er mit einem zufälligen Knoten und wählt in jedem Schritt die kürzeste Kante, die einen neuen Knoten mit dem wachsenden Baum verbindet.



Ablauf des Algorithmus von Prim

1. Initialisiert man den Algorithmus mit dem Startknoten und der noch leeren Kantenmenge

Diese Schritte werden nun wiederholt:

2. Man wählt eine Kante mit minimalem Gewicht aus dem aktuellen Graphen (nicht nur von diesem Knoten sondern im ganzen Graphen der geringste) aus.
3. Man fügt der Menge S den Knoten hinzu
4. Man fügt der Menge T die Kante hinzu.

Somit hat man am Schluss den leichtesten Spannbaum aller möglichen Spannbäume von G

In diesem Beispiel

Initialisierung: $S := a, \quad T := \emptyset$

1. Von a aus ist die Kante zu b die billigste
Also:
$S := \{a, b\}, T := \{\{a, b\}\}$

$w(T) = 1$

2. Von b aus ist die Kante zu d die billigste:

$S := \{a, b, d\},$
$T := \{\{a, b\}, \{b, d\}\}$
$w(T) = 1 + 2$

...

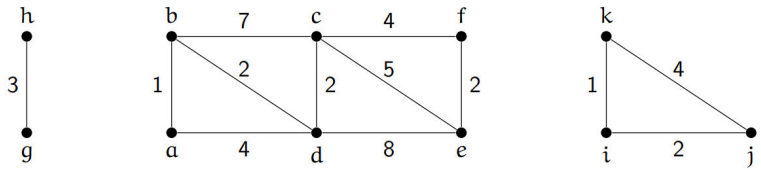
7. Zuletzt ist die Kante von f zu e die billigste:

$S := \{a, b, d, g, c, h, f, e\}$
$T :=$
$\{\{a, b\}, \{b, d\}, \{a, g\}, \{d, c\}, \{g, h\}, \{c, f\}, \{f, e\}\}$
$w(T) = 1 + 2 + 2 + 2 + 3 + 4 + 2 = 16$

Algorithmus von Kruskal

Der Algorithmus von Kruskal berechnet einen sogenannten minimalen aufspannenden Wald T von G und die Menge R der Zusammenhangskomponenten von G.

Ein aufspannender Wald von G ist ein Teilgraph T von G ohne Kreis, der alle Knoten von G enthält. Ein aufspannender Wald T heisst minimal, wenn $w(T) \leq w(T')$ für jeden aufspannenden Wald T' gilt.



Ablauf des Algorithmus

1. Initialisierung von R und T, R sind dabei alle Knoten

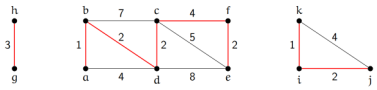
Ab hier Wiederholung:

2. Wähle die Kante in G mit minimaler Gewichtung (unabhängig vom Startknoten, es geht rein um die Kanten)
3. In R wird wird der Knoten mit der Vereinung der Knoten ersetzt.
4. Füge die Kante der Menge T hinzu.

Wiederholung ende!

Für das rechte Beispiel:

Der Spannbaum in diesem Beispiel sieht am Ende so aus:



In diesem Beispiel:

1. Initialisierung:

$$R := \{ a, b, c, d, e, f, g, h, i, j, k \}$$
$$T := \emptyset$$

2. Kleinste Kante a, b (oder i, k die Reihenfolge ist egal)

$$R := \{ \{ a, b \}, c, d, e, f, g, h, i, j, k \}$$
$$T := \{ \{ a, b \} \}$$
$$w(T) = 1$$

3. Kleinste Kante i, k:

$$R := \{ \{ a, b \}, c, d, e, f, g, h, \{ i, k \}, j \}$$
$$T := \{ \{ a, b \}, \{ i, k \} \}$$
$$w(T) = 1 + 1$$

4. Wir wählen wieder die kürzeste noch verbleibende Kante e, f (auch hier könnte man jetzt zuerst i, j nehmen, aber Reihenfolge egal)

...

8. Die letzte verbleibende Kante, die keinen Kreis im Spannbaum erzeugt ist c, f:

$$R := \{ \{ a, b, c, d, e, f \}, \{ g, h \}, \{ i, k, j \} \}$$
$$T := \{ \{ a, b \}, \{ i, k \}, \{ e, f \},$$
$$\{ c, d \}, \{ b, d \}, \{ i, j \}, \{ g, h \}, \{ c, f \} \}$$
$$w(T) = 1 + 1 + 2 + 2 + 2 + 2 + 3 + 4$$

Satz von Kirchhoff

Der Satz wird verwendet für die Berechnung der Spannbäume / Gerüste eines Graphen. Der Satz von Kirchhof lautet:

$$T(G) = |det(L^*)|$$

Wobei:

- T: Dies ist die gesuchte Anzahl der Spannbäume / Gerüste im Graphen
- L^*: Das ist eine modifizierte Laplace-(Admittanz-)Matrix des Graphen. Man erhält L^*, indem man eine Zeile und eine Spalte (üblicherweise die letzte oder die erste) aus der vollständigen Laplace-Matrix L des Graphen entfernt.
  Die Laplace-Matrix selbst wird gebildet durch die Gradmatrix
  D und die Adjazenzmatrix A mit der Formel $L = D - A$
- $|det(L^*)|$: Der Betragswert (Positivwert) der Determinante der modifizierten Laplace-Matrix L^*.

Anhang

Im Anhang befinden sich Vorlagen, Informationen zu Eingaben im TR, und sonstige hilfreiche Elemente die nicht direkt mit einem Thema zusammenhängen oder einfach nur eine Hilfestellung zur Lösung von Aufgaben dienen sollte.

Wahrheitstabellen (2, 3)

p	q
w	w
w	f
f	w
f	f

p	q	r
w	w	w
w	w	f
w	f	w
w	f	f
f	w	w
f	w	f
f	f	w
f	f	f

TI-30X Pro MathPrint Functions

Function	Shortcut
Binomial Koeffizient	`x` ncr `y`
lcm (kgV)	math + 2
gcd (ggT)	math + 3
Primzahl Faktorisierung	math + 4
Summe berechnen	math + > + 8
Bruch ↔ Dezimal	2nd + $\approx$
Gleichung lösen	2nd + sin (num-solve)
Polynomische Gleichung lösen	2nd + cos (poly-solve)
In bestehenden Term einsetzen	2nd + `delete` (insert)

Auch das Rechnen mit Matrizen funktioniert mit diesem Taschenrechner, ist jedoch recht komplex zusammenzufassen. Dies sollte man vorher einige Male ausprobieren!

Primzahlen (1-100)

i	1	2	3	4	5	6	7	8	9	10
01-10	2	3	5	7	11	13	17	19	23	29
11-20	31	37	41	43	47	53	59	61	67	71
21-30	73	79	83	89	97	101	103	107	109	113
31-40	127	131	137	139	149	151	157	163	167	173
41-50	179	181	191	193	197	199	211	223	227	229
51-60	233	239	241	251	257	263	269	271	277	281
61-70	283	293	307	311	313	317	331	337	347	349
71-80	353	359	367	373	379	383	389	397	401	409
81-90	419	421	431	433	439	443	449	457	461	463
91-100	467	479	487	491	499	503	509	521	523	541

Asymmetrische Verschlüsselung und der Satz von Euler — Zusammenhang erklärt

Die Eulersche φ-Funktion sagt dir, wie viele Zahlen kleiner als *n* teilerfremd zu *n* sind. Also wie viele Zahlen nicht die gleichen Teiler wie *n* haben, außer der 1. Das ist so, als würdest du in einer großen Gruppe von Menschen nach denen suchen, die keine gemeinsamen Hobbys mit dir haben, außer einem ganz bestimmten.

Was auf der Folie "verallgemeinert" wird, ist der kleine Satz von Fermat, der nur für Primzahlen gilt, und sagt, dass, wenn du eine Zahl *a* nimmst und sie *n*−1 Mal mit sich selbst multiplizierst (wenn *n* eine Primzahl ist), das Ergebnis immer modulo *n* gleich 1 ist. Der Satz von Euler erweitert diese Regel auf alle Zahlen *n*, nicht nur Primzahlen, und benutzt dafür die φ-Funktion.

Hier ist ein einfaches Beispiel:

Nehmen wir an, *n* ist 10 und *a* ist 3. Die φ-Funktion von 10 ist 4, weil es vier Zahlen gibt, die kleiner als 10 und teilerfremd zu 10 sind: 1, 3, 7 und 9.

Jetzt sagen wir, wir "verschlüsseln" unsere Zahl 3, indem wir sie $\varphi(n)$-mal mit sich selbst multiplizieren. Also:

$3^{\phi(10)} = 3^4$

Das Ergebnis ist:

$3^4 = 81$

Wenn wir 81 modulo 10 nehmen, erhalten wir 1.

Das ist das Ergebnis, das der Satz von Euler vorhersagt. Egal welche Zahl a wir nehmen, wenn wir sie $\phi(n)$-mal mit sich selbst multiplizieren, erhalten wir modulo n immer 1.

Das Problem bei der Entschlüsselung, wenn man n nicht hat, dass man die ϕ-Funktion von n nicht berechnen kann. Ohne n weisst du nicht, wie oft du a mit sich selbst multiplizieren musst, um beim Entschlüsseln wieder auf 1 zu kommen. Es ist, als hättest du einen Tresor mit einem Zahlenschloss, aber du kennst die Kombination nicht. Ohne die richtige Kombination (in diesem Fall die ϕ-Funktion und n), kannst du den Tresor nicht öffnen und die geheime Nachricht nicht lesen.