Algorithmen und Datenstrukturen

Theoretische Grundlagen der Informatik

Prof. Dr. rer. nat. Jörg Striegnitz

Fachhochschule Aachen
striegnitz@fh-aachen.de

19. April 2024

Algorithmen und Datenstrukturen

Abzählende Kombinatorik

Prof. Dr. rer. nat. Jörg Striegnitz

Fachhochschule Aachen
striegnitz@fh-aachen.de

19. April 2024

Standortbestimmung

Standort

- **Wir haben gelernt**
  - wie man auch komplexere Daten durch Wörter (Sprachen) beschreiben kann
  - dass Σ_{Bool} als Alphabet ausreichend ist (Homomorphismen)
- **Weitere Fragen bezüglich der Kodierung von Daten**
  - Wie viele Wörter enthält eine Sprache, wenn Sie nicht unendlich ist?
    [Wie viele Wörter hat z.B. $L_{Hamil}(G) = \{w \mid w \text{ kodiert Hamiltonschen Kreis in } G\}$]
- **Noch zu klären**
  - Wie kodieren wir algorithmische Probleme mithilfe von Sprachen?
    [Das sind Probleme, die wir durch Algorithmen lösen wollen]
- **Offen gebliebene Fragen zur Kodierung von Daten**
  - Gibt es eine zu einem Datum eine kürzeste Darstellung als Wort über einem beliebigem Alphabet?
    [Diese Frage hatten wir vertagt]

Motivation

- Unter der Größe (**Mächtigkeit / Kardinalität**) $|L|$ einer endlichen Sprache L verstehen wir die Anzahl ihrer Wörter (Elemente)
  Den Begriff der Mächtigkeit werden wir auf unendliche Mengen ausdehnen

- Offenbar ist es wichtig, die Größe einer Sprache zu kennen, um z.B. Aussagen über den Aufwand zur Lösung eines Problems machen zu können
  [→ Komplexitätstheorie, später]

- Werkzeug hierzu: **Abzählende Kombinatorik**

Ziehen von Elementen aus einer Menge

Zentrale Frage

Wie viele Möglichkeiten gibt es, k Elemente aus einer n-elementigen Menge zu ziehen?

- Man unterscheidet
  - Ziehen mit und ohne Zurücklegen
  - Beachtung / Nicht-Beachtung der Reihenfolge der gezogenen Elemente (geordnet / ungeordnet)
    - geordnet: Ergebnis der Ziehung als Folge
    - ungeordnet: Ergebnis der Ziehung als Menge

Ziehen mit Zurücklegen / geordnet

- In jedem Zug hat man n Möglichkeiten, insgesamt also

$$n^k$$

Möglichkeiten.

Beispiel 1.23

Betrachte $\Sigma = \{q, r, s, t\}$ und

$$L = \{w \in \Sigma^* \mid |w| = 3\}$$

Frage: Wie viele Wörter enthält L? Anders gefragt: Wie viele unterschiedliche Wörter der Länge 3 gibt es in Σ^*.
Antwort: Es gibt $|\Sigma|^3 = 4^3 = 64$ solcher Wörter - $|L| = 64$

Ziehen ohne Zurücklegen / geordnet

- Im ersten Zug n Möglichkeiten, im zweiten Zug $(n-1)$ Möglichkeiten usf., bis man im letzten Zug noch $(n-k+1)$ Möglichkeiten hat.
  Insgesamt hat man daher

$$n^{\underline{k}} = n \cdot (n-1) \cdot \; \cdots \; \cdot (n-k+1) = \prod_{i=0}^{k-1} (n-i)$$

  Möglichkeiten.
- $n^{\underline{k}}$ nennt man **fallende Faktorielle**. und es gelten:
  - $n^{\underline{n}} = n!$
  - $n^{\underline{0}} = 1$

Beispiel 1.24

Betrachte $\Sigma = \{q, r, s, t\}$ und

$$L = \left\{ w \in \Sigma^* \;\middle|\; |w| = 3 \;\wedge\; \forall a \in \Sigma : |w|_a \leq 1 \right\}$$

Frage: Wie viele Wörter enthält L? Anders gefragt: Wie viele unterschiedliche Wörter der Länge 3 gibt es in Σ^*, bei denen kein Buchstabe doppelt auftritt?
Antwort: Es gibt $|\Sigma|^{\underline{3}} = 4 \cdot 3 \cdot 2 = 24$ solcher Wörter - $|L| = 24$

Ziehen ohne Zurücklegen / ungeordnet

- Bei den k gezogenen Elementen ist die Reihenfolge uninteressant
  - z.B. werden q, r, s und r, s, q als identisches Ereignis betrachtet
  - Es gibt $k^{\underline{k}} = k!$ viele Möglichkeiten, k Elemente anzuordnen
    Ziehen ohne Zurücklegen, geordnet
- Damit gibt es

$$\frac{n^{\underline{k}}}{k!} = \binom{n}{k} = \frac{n!}{(n-k)! \cdot k!}$$

viele Möglichkeiten k Elemente ungeordnet aus n Elementen zu ziehen.

Beispiel 1.25

Betrachte $\Sigma = \{q, r, s, t\}$ und

$$L = \left\{ w \in \Sigma^* \;\middle|\; |w| = 5 \;\wedge\; |w|_q = 2 \right\}$$

Frage: Wie viele Wörter enthält L? Anders gefragt: Wie viele unterschiedliche Wörter der Länge 5 gibt es in Σ^*, die genau 2 q's enthalten?
Antwort: Es gibt $\binom{5}{2} \cdot 3^3 = \frac{5^{\underline{2}}}{2!} \cdot 27 = 270$ solcher Wörter. Idee: Die beiden q's stehen an zwei der fünf möglichen Positionen; die restlichen drei Positionen werden mit den verbleibenden drei Buchstaben gefüllt.

Hierzu führen wir das Konzept der **Multimenge** ein:

- In einer Multimenge dürfen Elemente wiederholt auftreten, z.B. ist

$$S = \{e, b, e, e, c, b\} = \{b, b, c, e, e, e\}$$

  eine Multimenge über der Menge $M = \{b, c, e\}$
- Die Häufigkeit des Auftretens eines Elements a in einer Multimenge S bezeichnen wir als **Vielfachheit** (i.Z. $|S|_a$); z.B. ist $|\{d, e, e\}|_e = 2$
- Die **Mächtigkeit** einer Multimenge entspricht der Zahl ihrer Elemente gezählt mit ihrer Vielfachheit; z.B. ist $|\{e, e, c, c, b\} = 5|$

Kodierung von Multimengen

Definition 1.30 (Kodierung von Multimengen)

Seien $M = \{a_1, \cdots, a_n\}$ eine Menge, $< \subseteq M \times M$ eine Relation (Ordnung über M) mit $a_1 < a_2 < \cdots < a_n$, $S = \{a_{i_1}, \cdots, a_{i_k}\}$ eine Multimenge (mit $i_j \in \{1, \cdots, n\}$ für $1 \leq j \leq k$) über M und $\Sigma = \{|, *\}$ ein Alphabet. Dann kodieren wir S wie folgt als Wort über Σ^*:

$$*^{|S|_{a_1}} \mid *^{|S|_{a_2}} \mid \cdots \mid *^{|S|_{a_n}}$$

Beispiel 1.26

Seien $M = \{a, b, c, d, e\}$ und $S = \{a, a, a, b, d, d, d\}$. Dann kodieren wir S durch

$$\underbrace{* * *}_{\text{Anzahl } a\text{'s}} \mid \underbrace{*}_{\text{Anzahl } b\text{'s}} \mid\mid \underbrace{* * *}_{\text{Anzahl } d\text{'s}} \mid$$

! Die Kodierung einer Multimenge der Mächtigkeit k über einer Menge der Mächtigkeit n enthält $n + k - 1$ viele Symbole

k-mal »$*$« und $n - 1$-mal »$|$«

Ziehen mit Zurücklegen / ungeordnet

- **Experiment**: k Elemente ungeordnet aus n Ziehen; mit Zurücklegen
- **Gesucht**: Zahl der Multimengen der Mächtigkeit k über einer Menge mit n Elementen
- **Idee**: Zahl der k-elementigen Multimengen entspricht der Zahl der möglichen Kodierungen, diese ist

$$\binom{n+k-1}{k}$$

ziehe für k Symbole »$*$« eine Position aus $n+k-1$ möglichen Positionen - Ziehen ohne Zurücklegen; ungeordnet