Proseminar *Algorithms and Data Structures*

Einführungsveranstaltung
Wintersemester 2015/16; 23. Oktober 2015

Thomas Noll
Software Modeling and Verification Group
RWTH Aachen University

`http://moves.rwth-aachen.de/teaching/ws-1516/algds/`

Software Modeling and Verification Chair

RWTH AACHEN UNIVERSITY

Übersicht

Einführung

Termine

Algorithmen

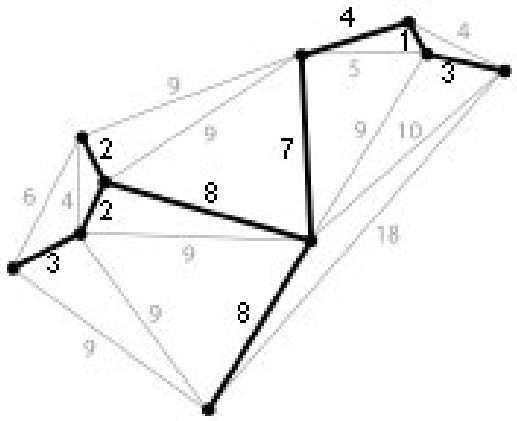
Datenstrukturen

Seminarthema

Thema des Proseminars

Weiterführung und Vertiefung diverser Themen der Vorlesung *Datenstrukturen und Algorithmen*

- Algorithmen:
  - neue Algorithmen
  - andere Komplexitätsmaße, ...
- Datenstrukturen:
  - Bäume
  - Hashing, ...
- Inhalt:
  - Problemstellung
  - Arbeitsweise des Algorithmus bzw. der Datenstruktur
  - Effizienzeigenschaften
  - Anwendungen



Zielsetzung

Ziele des Proseminars

- Selbstständiges Einarbeiten in ein neues Thema
- Literaturrecherche
- Darstellen des Inhalts in einer wissenschaftlichen Ausarbeitung
- Verständliches Präsentieren

Anforderungen Ausarbeitung

Ausarbeitung

- Selbstständiges Verfassen einer ca. 15-seitigen Ausarbeitung
- Vollständiges Literaturverzeichnis
- Korrektes Zitieren
- Plagiarismus:
  Die nicht gekennzeichnete übernahme fremder Inhalte führt zum sofortigen Ausschluss.
- Schriftgröße 12pt, übliche Seitenränder
- Titelseite mit Thema, Titel Proseminar, Semester, Name, Datum
- Sprache Deutsch oder Englisch (Achtung: „[E]“)
- Korrekte Sprache wird vorausgesetzt:
  $\geq$ 10 Fehler pro Seite $\Longrightarrow$ Abbruch der Korrektur

Anforderungen Vortrag

Vortrag

- 30-minütiger Vortrag
- Zielgruppengerechte Präsentation der Inhalte
- übersichtliche Folien:
  - $\leq$ 15 Textzeilen
  - sinnvoller Einsatz von Farben
- Vortrag in Deutsch oder Englisch

Übersicht

Einführung

Termine

Algorithmen

Datenstrukturen

Themenauswahl

Verfahren

- Themenliste wurde/wird ausgehändigt
- Priorisierte Auswahl
- Wir bemühen uns (ohne Garantie) um ein “optimales” Matching
- Zuordnung der Themen bis 27. Oktober online
- Betreuer wurden bereits zugeordnet
- Einige Themen auf Englisch (zumindest Betreuung und Ausarbeitung), Rest nach Wahl

Rücktritt vom Proseminar

- Bis zu drei Wochen nach Einführung: ohne Folgen
- Danach: Fehlversuch

Termine

Bibliothekseinführung

Einführung in die Literaturrecherche

- Einweisung in themenspezifische Literaturrecherche
- Dauer: ca. zwei Stunden
- Teilnahme für BSc-Studierende verpflichtend
- Bedarf bitte auf Themenblatt vermerken
- Termine zur Auswahl:
  - Montag, 09.11., 14 Uhr
  - Mittwoch, 11.11., 16 Uhr
  - Freitag, 13.11., 14 Uhr

Deadlines

Deadlines

Folgende Termine sind einzuhalten:

- 16.11.2015: letzte Rücktrittsmöglichkeit
- 23.11.2015: Vorlage der Inhaltsübersicht
- 21.12.2015: endgültige Fassung der Ausarbeitung
- 01.02.2016: endgültige Fassung der Folien
- 15./16.02.2016: Blockseminar

Übersicht

Einführung

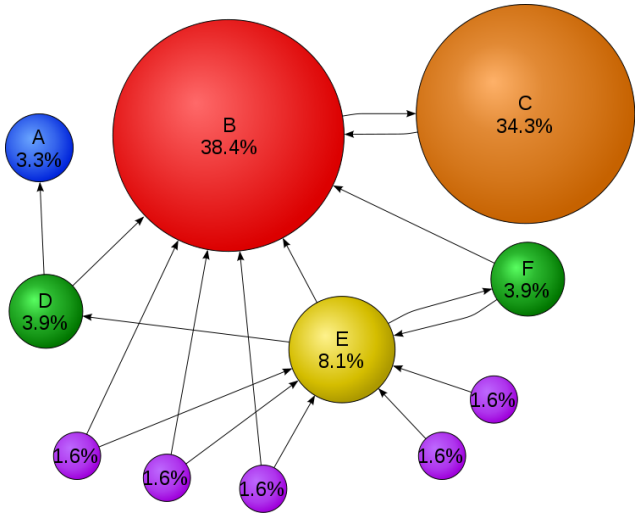
Termine

Algorithmen

Datenstrukturen

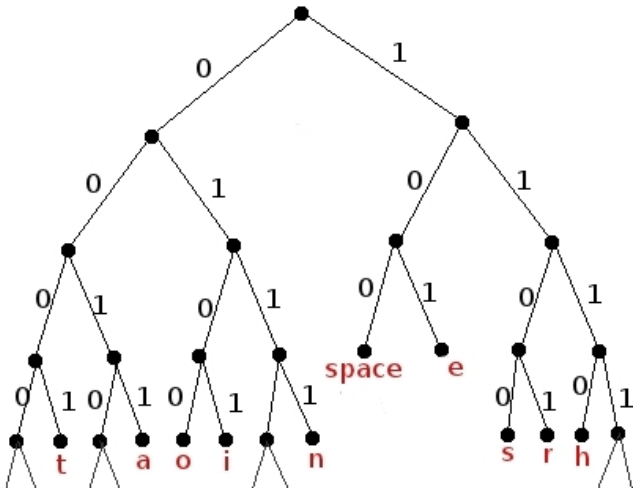
Algorithmen

1. Pagerank-Algorithmus



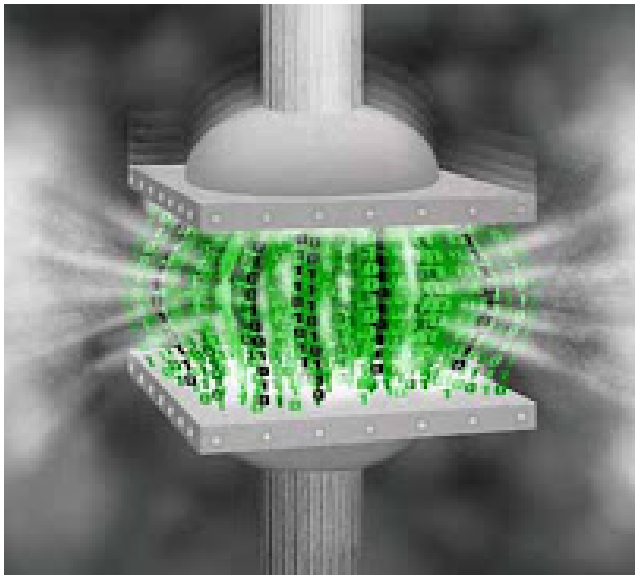
- Algorithmus zur Analyse verlinkter Webseiten
- Allgemeiner: beliebige Graphen
- Basis: Bestimmung der Wahrscheinlichkeit, dass zufälliges Benutzerverhalten auf die Seite führt
- Benutzung in Google-Suchmaschine

2. Huffman-Kodierung



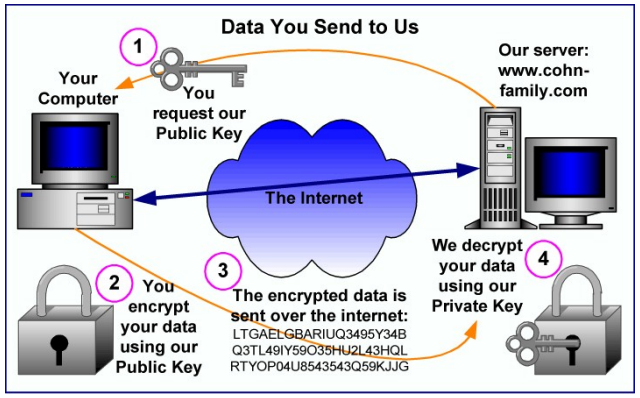
- Ziel: Erstellung einer Binärkodierung mit minimaler mittlerer Wortlänge benötigt
- Basis: Wahrscheinlichkeitsverteilung der einzelnen Zeichen
- Ansatz: Verwendung eines vollen Binärbaums zur Darstellung des Codes

3. LZW-Datenkompression [Englisch]



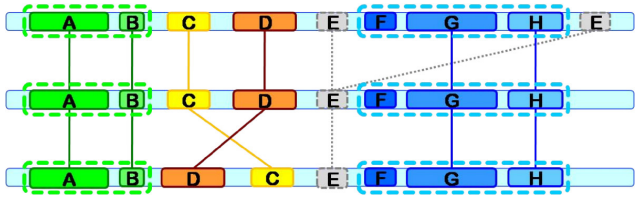
- Lempel-Ziv-Welch-Algorithmus
- Häufig bei Grafikformaten eingesetzt
- Verlustfrei
- Kompression mittels Wörterbüchern mit am häufigsten vorkommenden Zeichenketten
- Wörterbuch nicht zusätzlich gespeichert, sondern aus Datenstrom rekonstruiert

4. RSA-Kryptosystem



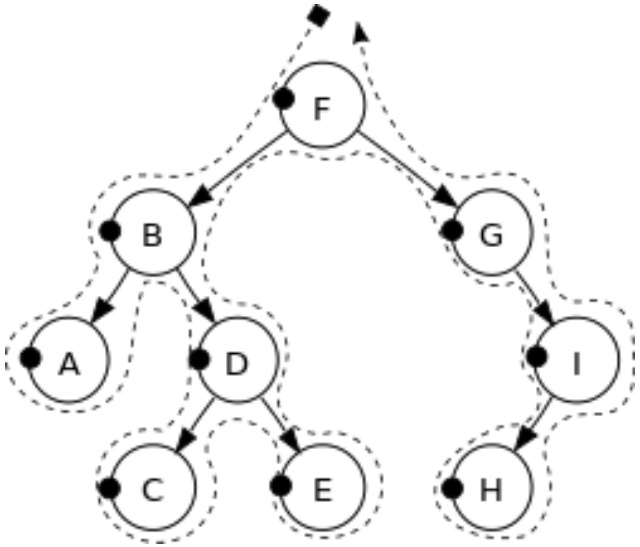
- Verfahren von Rivest, Shamir und Adleman (“public-key encryption”)
- Verwendung zur Verschlüsselung und digitalen Signatur
- Asymmetrisch: Schlüsselpaar privat/öffentlich
  - privat: zum Entschlüsseln oder Signieren
  - öffentlich: zur Verschlüsselung/ Signaturprüfung
- Privater Schlüssel wird geheim gehalten und kann nur mit extrem hohem Aufwand aus dem öffentlichen Schlüssel berechnet werden

5. Longest Common Subsequence



- Definition: längste gemeinsame (nicht notwendigerweise zusammenhängende) Teilsequenz mehrerer Zeichenketten
- Anwendungen: `diff`, Bioinformatik
- NP-hart für beliebige Anzahl von Zeichenketten
- Quadratisch für zwei Zeichenketten (dynamische Programmierung)

6. Deutsch-Schorr-Waite Baumtraversierung

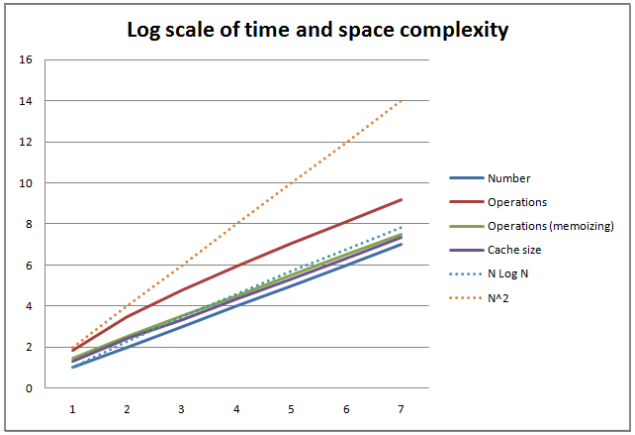


- Klassische Lösung: Rekursion/Stack
- Ziel: Vermeidung des zusätzlichen Speicheraufwands
- Ansatz: Pointerrotation in der Heapdarstellung

7. Problem der K kürzesten Pfade

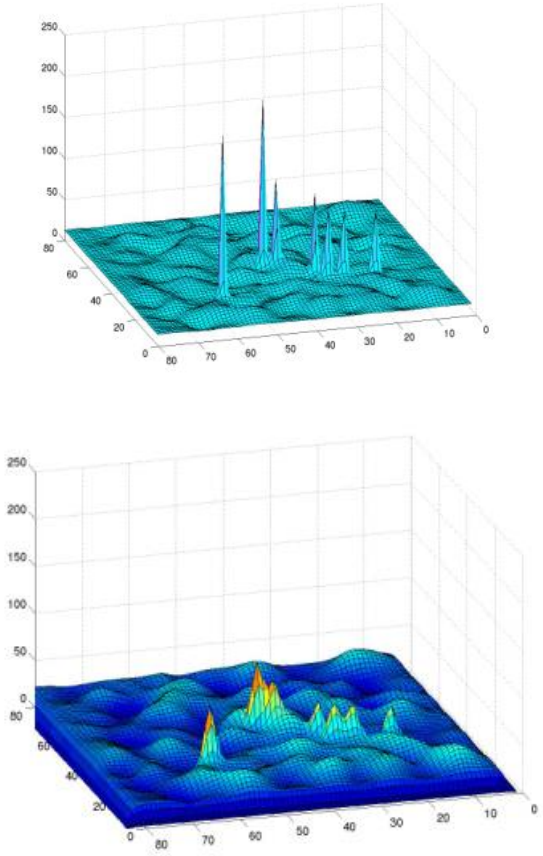
- Problem: Finden der K kürzesten Pfade zwischen Knotenpaar in einem gerichteten Graphen
- Ansatz: Verallgemeinerung der Bellman-Gleichung

8. Amortisierte Laufzeitanalyse [Englisch]



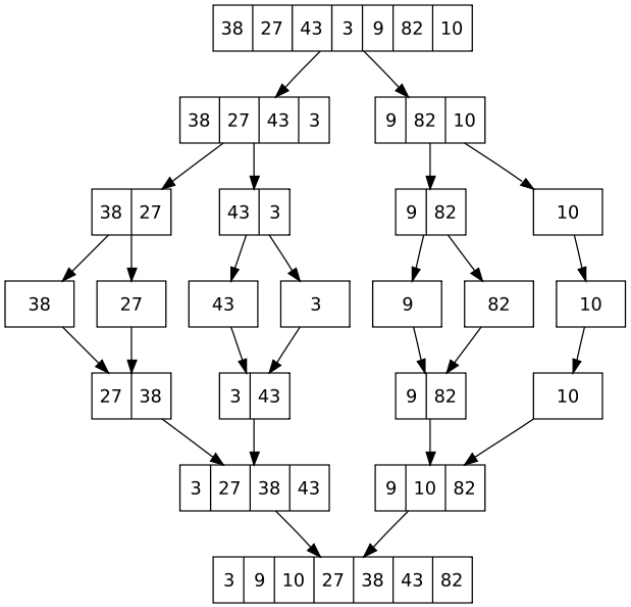
- Allgemeine Laufzeitanalyse: maximale Kosten der einzelnen Schritte
- Amortisierte Laufzeitanalyse: Worst Case aller Operationen im gesamten Durchlauf des Algorithmus
- Verbesserung der oberen Schranke bei seltenem Auftreten teurer Operationen
- Idee: der Worst Case ändert den Zustand der Datenstruktur so ab, dass er nicht wiederholt auftreten kann (z.B. dynamische Arrays)
- Drei unterschiedliche Berechnungsmethoden (Aggregat-, Account-, Potentialfunktion-Methode)

9. Smoothed Analysis



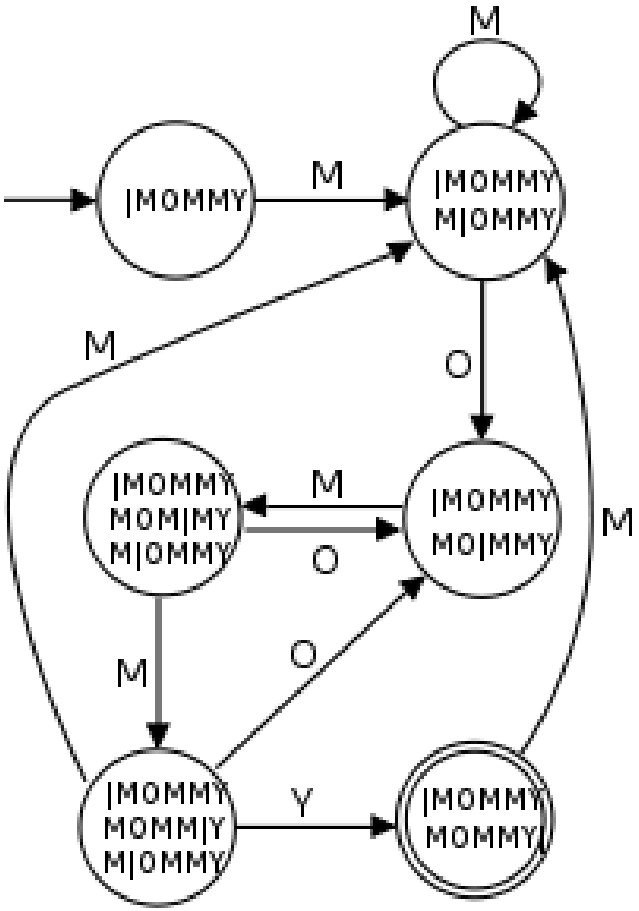
- Ziel: realistische Abschätzung des Laufzeitverhaltens von Algorithmen
- Kombination von Worst-Case- und Average-Case-Analyse
- Annahme: “typische” Eingabewerte mit relativen geringen Abweichungen (“perturbation”)
- Liefert „geglättete“ Effizienzergebnisse

10. Externes Sortieren [Englisch]



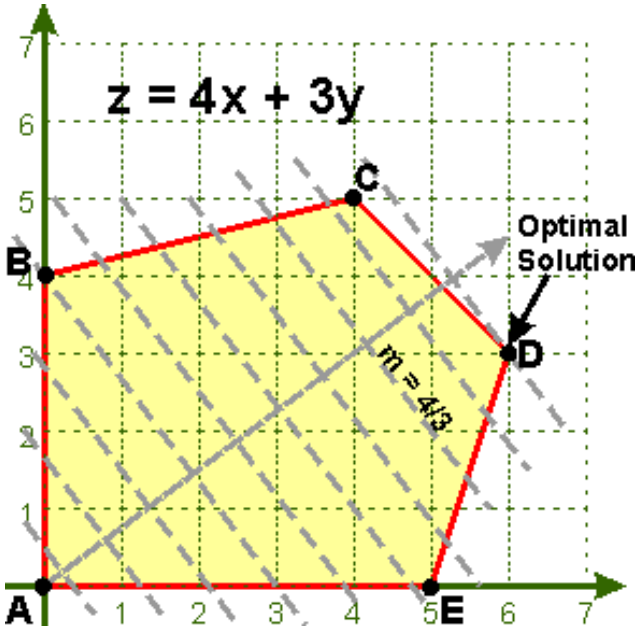
- Ziel: Sortierung großer Datenmengen unter Verwendung externer Speicher
- Typischer Ansatz: divide-and-conquer (sort-merge-Strategien)

11. String Matching [Englisch]



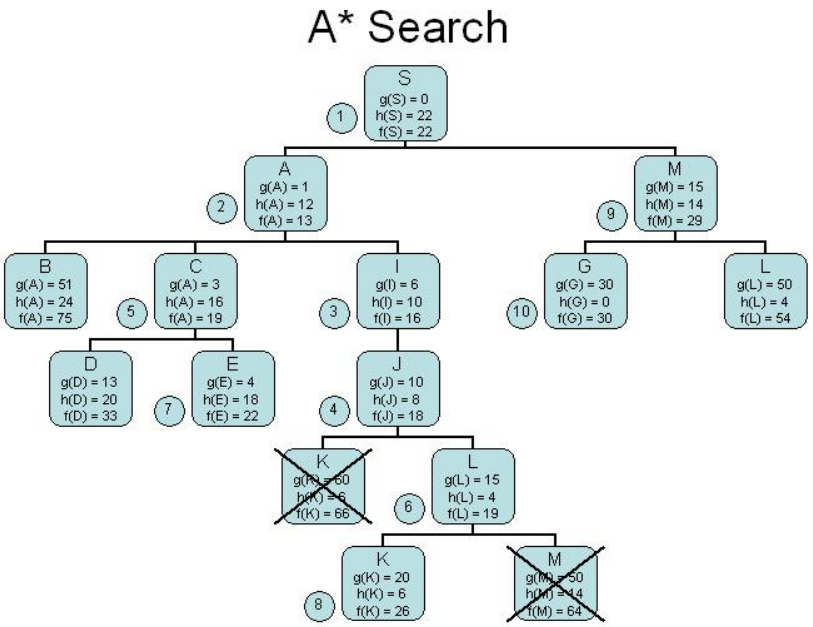
- Ziel: Suche von Vorkommen eines Strings in einem (großen) Text
- Komplexität des naiven Algorithmus: $|\text{Text}| \cdot |\text{String}|$
- Verbesserung durch Vorverarbeitung des Suchstrings

12. Lineare Optimierung (aka Lineare Programmierung)



- Ziel: Optimierung linearer Zielfunktionen über einer Menge, die durch lineare (Un-)Gleichungen eingeschränkt ist
- Simplexverfahren
- Ellipsoid-Methode

13. A*-Suchalgorithmus



- Berechnung eines kürzesten Pfades zwischen zwei Knoten in einem Graphen mit positiven Kantengewichten
- Verallgemeinerung und Erweiterung des Dijkstra-Algorithmus
- Schätzfunktion (Heuristik) zur zielgerichteten Suche

Übersicht

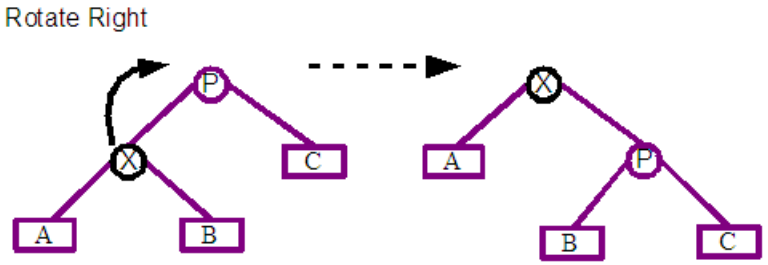
Einführung

Termine

Algorithmen

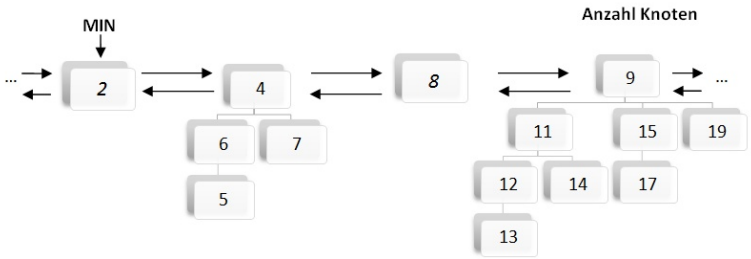
Datenstrukturen

14. Splay Trees



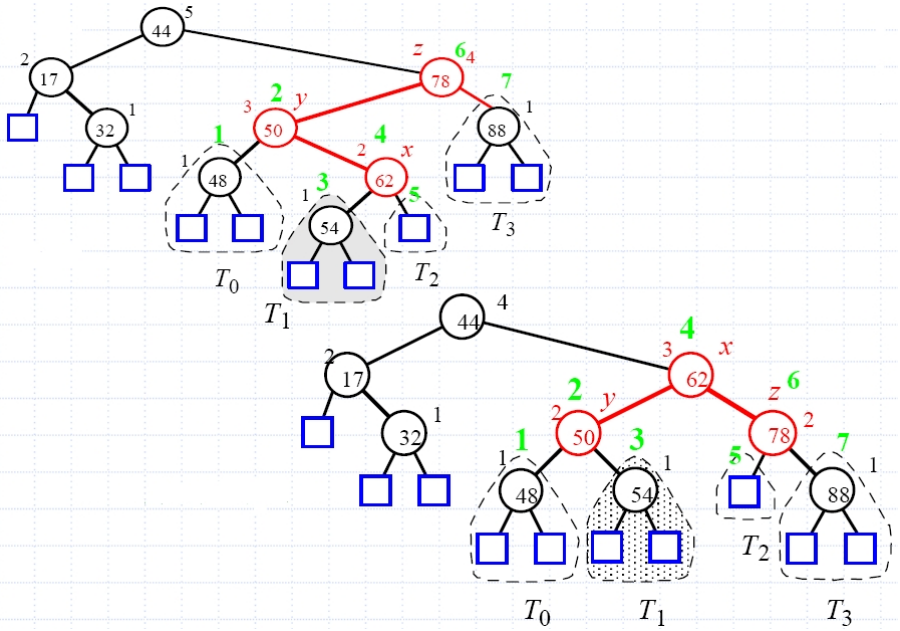
- Baumorganisierte Datenstruktur
- Effizientes Einfügen, Suchen und Löschen von Elementen
- Grundlegende Operation: *splay* (Rotieren eines Elements bis zur Wurzel)

15. Fibonacci Heaps



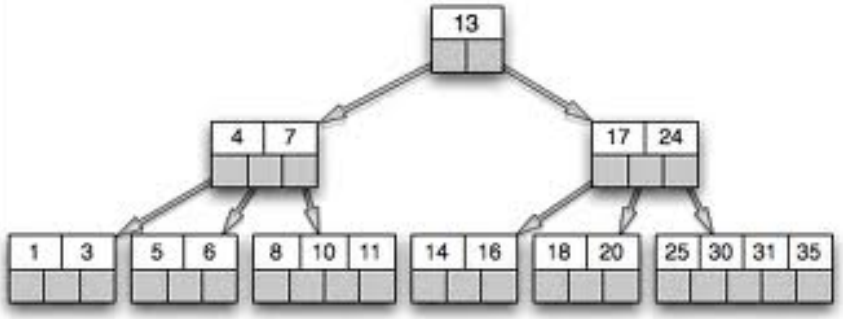
- Datenstruktur zur Implementierung einer Vorrangwarteschlange (*priority queue*)
- Liste von geordneten Bäumen
- Heap-Bedingung: Priorität jedes Knotens mindestens so groß wie Priorität seiner Kinder
- (Fast) alle Operationen haben amortisiert konstante Laufzeit

16. AVL-Bäume [Englisch]



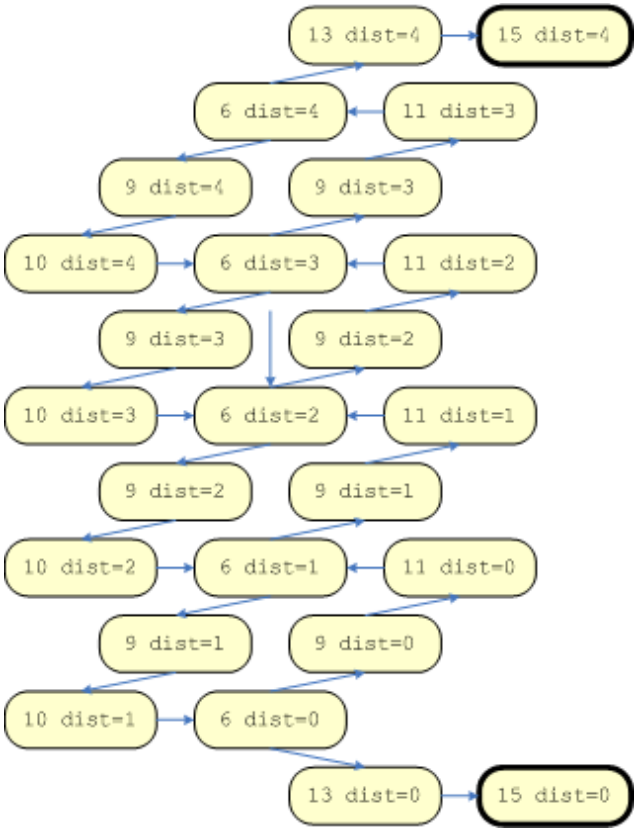
- Balancierter binärer Suchbaum
- Invariante: maximaler Höhenunterschied der Teilbäume jedes Knotens ist 1
- Worst-Case-Komplexität der üblichen Operationen: $O(\log n)$

17. B-Bäume [Englisch]



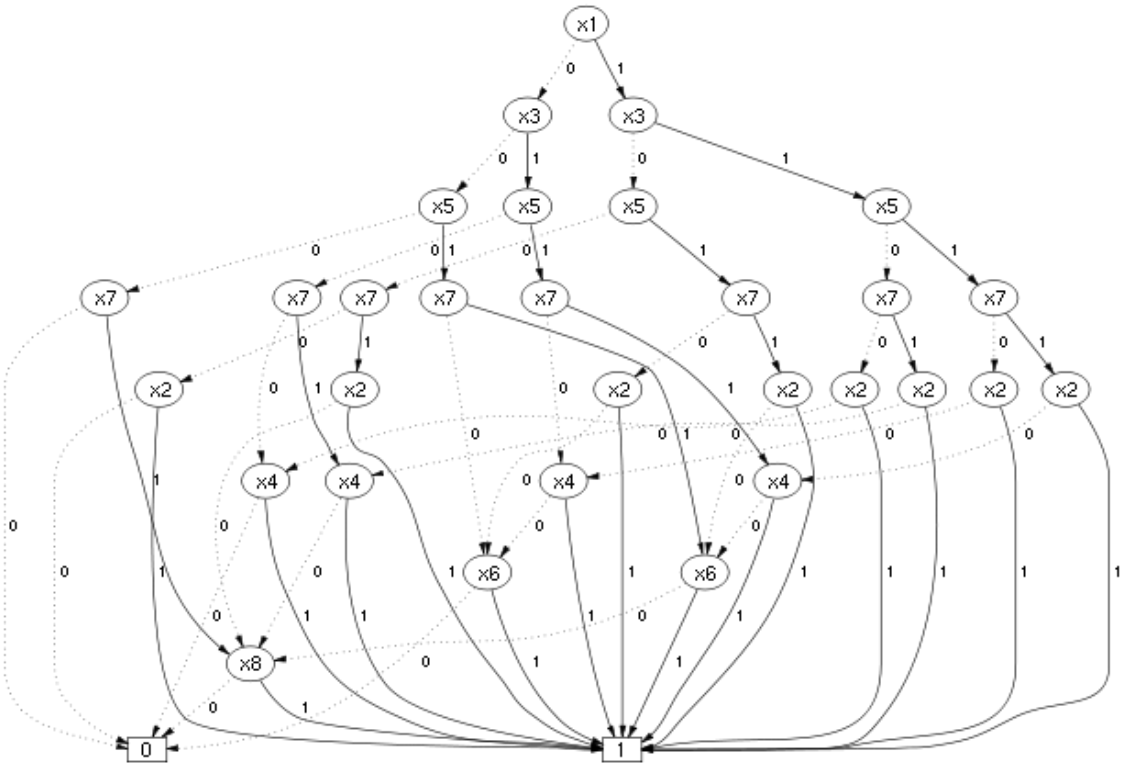
- Immer vollständig balanciert
- Besonderheit: Anzahl der Knotennachfolger variabel (mit Obergrenze)
- Einsatz vor allem in Datenbanken und Dateisystemen

18. Bitstate Hashing



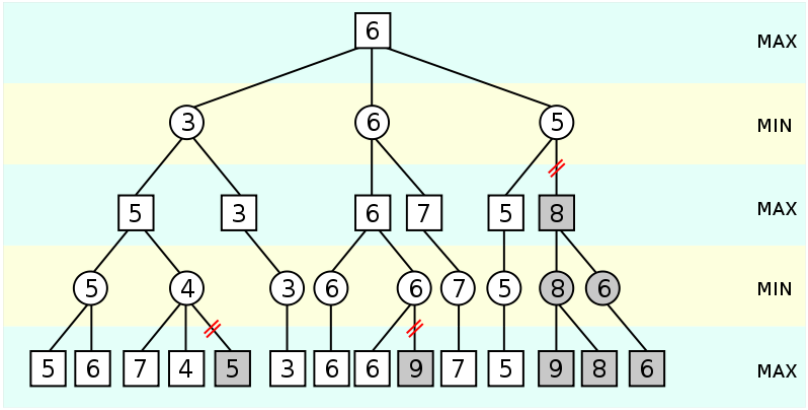
- Effiziente Darstellung von Zustandsräumen
- Problem: Erkennung von Zykeln
- Idee: Benutzung einer Hashfunktion, Speicherung von 0/1 im Hasharray
- Bis zu 98% Speicherersparnis im SPIN-Tool

19. Binäre Entscheidungsdiagramme



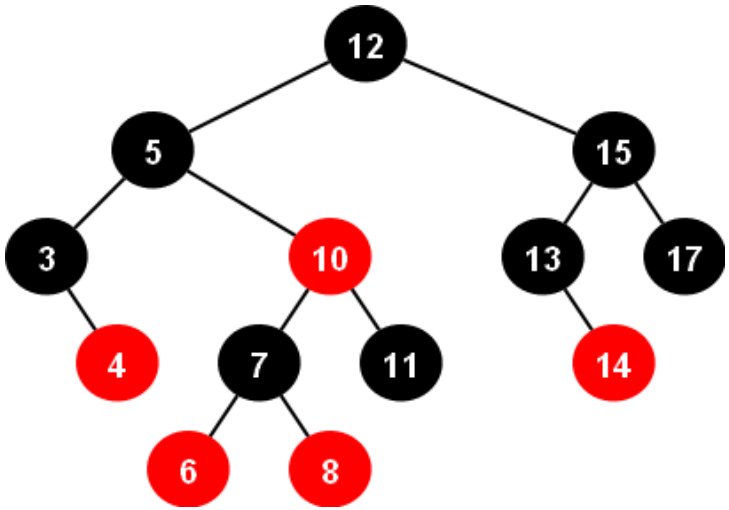
- Datenstruktur zur Darstellung Boolescher Funktionen
- Einsatz: Verifikation von (Hardware-)Systemen
- Problem: Finden geeigneter Variablenordnungen

20. Spielbäume [Englisch]



- Darstellung von 2-Personen-Spielen mit abwechselnden Zügen (z.B. Schach, Go, Reversi, Dame, Mühle oder Vier gewinnt)
- Ziel: Bestimmung optimaler Strategien
- Minimax-Verfahren
- α-β-Pruning

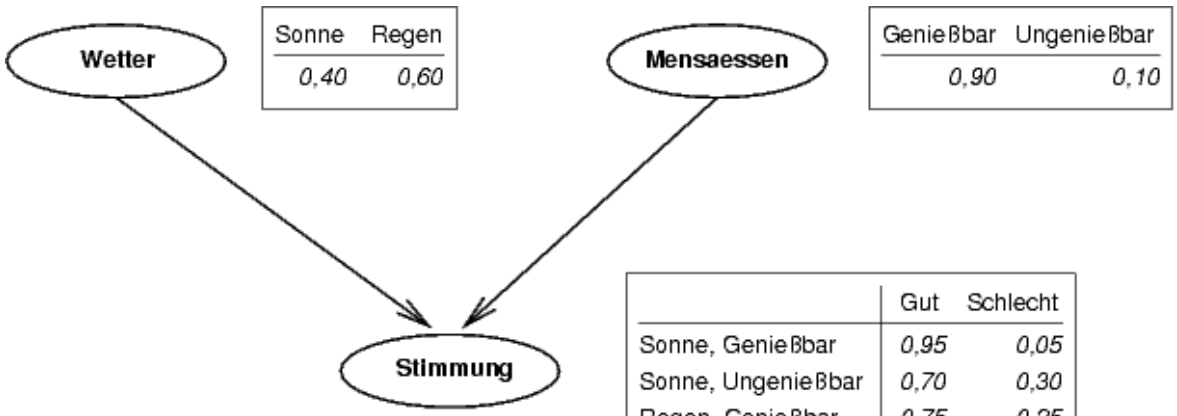
21. Rot-Schwarz-Bäume



- Von binären Suchbäumen abgeleitet
- 5 charakterisierende Eigenschaften:
  1. jeder Knoten ist rot oder schwarz
  2. Wurzel ist schwarz.
  3. alle Blätter sind schwarz.
  4. beide Kinder eines roten Knotens sind schwarz
  5. jeder Pfad von einem Knoten zu seinen Blattknoten enthält die gleiche Anzahl schwarzer Knoten

$\Longrightarrow$ annähernde Balancierung, effiziente Laufzeitschranken für Suchen/Einfügen/Löschen

22. Bayessche Netze



- Gerichteter azyklischer Graph (DAG)
  - Knoten = Zufallsvariablen
  - Kanten = bedingte Abhängigkeiten
- Kompakte Darstellung der gemeinsamen Wahrscheinlichkeitsverteilung aller Variablen
- Algorithmen:
  - Schließen (Inferenz)
  - Lernen