Debugging

- Der Prolog-Interpreter besitzt spezielle Prädikate für das Verfolgen von Beweisabläufen. Sie sind hilfreich bei der Fehlersuche und zum Verständnis der Arbeitsweise des Interpreters.
- Aus logischer Sicht werden die Beweismöglichkeiten durch einen Und-Oder-Baum beschrieben, bei dem anstelle von Variablen alle möglichen Instantiierungen vorhanden sind (entsprechend viele Alternativen an den Oder-Knoten).
- Aus prozeduraler Sicht werden die Instantiierungen erst verzögert vorgenommen (das entspricht der Darstellung bei trace usw.).

Fragen zum Nachdenken: Vergleichen Sie die Beschreibung des Verlaufs von Beweisversuchen mittels Und-Oder-Baum, Ferguson-Diagramm und Box-Modell.

Prolog-Prädikate

- Der Prolog-Interpreter stellt eingebaute Prädikate zur Verfügung. Neben den „rein logischen“ Prädikate gibt z.B. es Prädikate für Eingabe/Ausgabe, Programmsteuerung und für die Analyse/Synthese von Termen.

Fragen zum Nachdenken: Wieweit werden durch die eingebauten Prädikate Unterschiede zwischen deklarativer und prozeduraler Semantik verursacht?

Abstrakte Datentypen

- Abstrakte Datentypen definieren den „mathematischen Kern“ von oft benutzten Datenstrukturen einschließlich der dabei benötigten Verfahren. Dazu gehören auch Angaben über die Komplexität.
- Abstrakte Datentypen sind ein Hilfsmittel bei der Verständigung über Programme und bei ihrer Implementation.

Fragen zum Nachdenken: Wieweit reichen Angaben über das Eingabe-/Ausgabeverhalten (Schnittstellendefinition) von Programmen aus für die Wiederverwendbarkeit?

Listen

- Listen können auf unterschiedliche Weise definiert werden.
- Bei geordnete Listen entspricht die Reihenfolge einer Ordnungsrelation.
- Sortierverfahren ermöglichen das effiziente Sortieren.
- Differenzlisten ermöglichen in Prolog eine effizientere Verarbeitung.

Fragen zum Nachdenken: Welche Eigenschaften ermöglichen die Effizienz unterschiedlicher Sortierverfahren? Verfolgen Sie den Ablauf von append in Prolog mittels trace bzw. Ferguson-Diagramm.

Komplexität

- Die Verarbeitungszeit hängt bis auf einen konstanten Faktor nicht von der konkreten Maschine ab. Mit der O-Notation wird die Größenordnung beschrieben.
- Der Zeitbedarf kann oft verringert werden, wenn mehr Speicher eingesetzt wird.
- Bei vielen Problemen ist der Zeitbedarf so groß, dass auch stark verbesserte Rechentechnik nur noch geringe Vorteile bringt.

Fragen zum Nachdenken: Hinsichtlich der O-Notation spielt die Effizienz einer Maschine keine Rolle. Trotzdem lohnt es sich in der Praxis, ein Programm auf einer schnelleren Maschine laufen zu lassen. Worin liegt der „Widerspruch“?

Keller, Warteschlangen

- Keller und Warteschlangen sind Listen mit speziellen (eingeschränkten) Zugriffsformen. Sie werden z.B. bei Suchverfahren eingesetzt.
- Keller sind wichtig für die Abarbeitung von Programmen (Auswertung von Ausdrücken, Laufzeitkeller für Prozedur-Aufrufe).

Fragen zum Nachdenken: Wieweit eignet sich das „Kellerprinzip“ für die Programmierung selbständig handelnder Agenten?

Graphen

- Graphen dienen der Beschreibung (und oft Veranschaulichung) von Zusammenhängen. Sie besitzen Kanten und Knoten, die jeweils beschriftet sein können.
- Graphen können in unterschiedlicher Weise implementiert werden.

Fragen zum Nachdenken: Wann ist welche Implementierung vorteilhaft?

Graphen

- Graphen können statische oder dynamische Zusammenhänge beschreiben.
- Bei dynamischen Zusammenhängen sind „Zustände“ und „Übergänge“ zu definieren (Beispiele: Transitionssysteme, Zustandsmaschinen, Automaten, Akzeptoren, Petri-Netze, Neuronale Netze, ...).

Fragen zum Nachdenken: Wie lassen sich die dynamischen Zusammenhänge (unterschiedliche Zustände) in einem Graphen veranschaulichen? Kann die Dynamik einer Turing-Maschine als Graph dargestellt werden?

Bäume

- Bäume sind spezielle Graphen (insbesondere ohne Zyklen und Maschen).
- Bäume können rekursiv definiert werden.
- Ausgehend von der rekursiven Struktur können Bäume auch sequentiell (durch eine Zeichenkette) beschrieben werden.
- Die Menge der Wege in einem Graphen definiert den „Erreichbarkeitsbaum“.

Fragen zum Nachdenken: Kann aus dem Erreichbarkeits-baum der ursprüngliche Graph rekonstruiert werden?

Binäre Relationen

- Binäre Relationen können durch Graphen veranschaulicht werden: Die Knoten beschreiben die Objekte, die Kanten die Relation. Bestimmte Arten von Relationen entsprechen bestimmten Arten von Graphen.

Fragen zum Nachdenken: Gibt es Ordnungsrelationen, die durch einen kreisförmigen Graphen veranschaulicht werden?

Suche in Graphen

- Viele Probleme lassen sich als Suchprobleme in Graphen formulieren (Suche nach irgendeinem Weg, Suche nach dem kürzesten Weg, ...).
- Suchprobleme in Graphen werden grob unterschieden bzgl.
  - blinde Suche/heuristische Suche
  - Suche nach irgendeinem/nach einem kürzesten Weg

  Innerhalb dieser Einteilungen gibt es weitere Unterschiede (Tiefe/Breite-Zuerst, Expansionsrichtung, ...)

Fragen zum Nachdenken: Warum kann in praktischen Problemen die Liste CLOSED oft nicht verwendet werden? In welchem Zusammenhang stehen Erreichbarkeitsbaum und Suchverfahren?