

Projekt/Komplexe Softwaresysteme „Wir bauen einen Termprozessor“

Karsten Wolf

Was ist ein Term?

- Ein heterogener Baum
 - Knoten mit verschiedenen Typen
 - Nachfolgerzahl abhängig von Knotentyp
 - Verfügbare Attribute abhängig vom Knotentyp
 - Attributwerte für jeden Knoten individuell
- Beispiele
 - Syntaxbaum eines Programms
 - Knoten = Statements, Operatoren von Expressions
 - Logische Formeln
 - Knoten = Operatoren, Variablen

Was macht man häufig mit Termen?

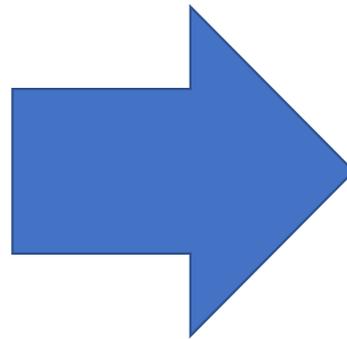
- „Rewriting“: Anwendung von Ersetzungsregeln
 - $\neg(x \wedge y) \rightarrow \neg x \vee \neg y$
 - $2 * x \rightarrow x + x$
 - If false then S1 else S2 \rightarrow S2
 - Repeat_until(body,condition) \rightarrow Sequence(body,While(Not(condition),body))
- „Unparsing“: Systematisches (rekursives) Ablaufen des Baumes
 - Ausgabe einer logischen Formel:
 - Schreibe (
 - *Bearbeite linke Teilformel*
 - Schreibe Operator conjunction(x,y) \rightarrow [“(“ x “ AND “ y “)“]
 - *Bearbeite rechte Teilformel*
 - Schreibe)

Was macht ein Termprozessor

Definition
Knotentypen

Ersetzungsregeln

Unparse-Regeln



Knotentypen.H

Rewrite.CC
Void rewrite()

Unparse.CC
Void unparse()

Was hilft mir ein Termprozessor?

- Kompakte Beschreibung
 - Entlastung von Routineprogrammierung
 - Konzentration auf das Wesentliche
 - Einfaches Ändern/Hinzufügen von Regeln
 - Generierter Code effizient, fehlerarm

Wer nutzt einen Termprozessor?

- Wir. Im Petrinetzwerkzeug „LoLA“ und anderen Werkzeugen
 - Als Syntaxbaum für die Eingabedateien
 - Optimierung und Verarbeitung temporallogischer Anfragen
 - Mehrere hundert Regeln, vor allem logische Tautologien zur Vereinfachung der Anfragen
- Andere. Heterogene Bäume gehören zu den zentralen Datenstrukturen der Informatik → Bereitstellung der Ergebnisse als Open Source Community Service (wenn wir es hinbekommen...)

Warum brauchen wir einen *neuen* Termprozessor?

- Kimwitu++ existiert, wird aber nicht mehr gewartet
 - Veraltete C++-Syntax
 - Nicht thread-safe
 - Hätten gern erweiterte Funktionalität
 - Thread-safe
 - Setzen von Attributen in Rewrite-Regeln
 - Mitgeben von zusätzlichen Parametern in Unparse-Regeln
- die neuen Anforderungen kommen direkt aus unserer eigenen Nutzererfahrung

Schaffen wir das?

- Wir werden vielleicht nicht fertig, können aber, falls Sie sowohl Projekt als auch KSWS absolvieren, im nächsten Semester fortsetzen

Was brauchen wir?

- Lust
- C++