

## Inhaltsverzeichnis

1. Einführung	1
1.1. Aufgaben und Struktur von Compilern	3
1.2. Attributierungen im Compilerbau	4
1.3. Attributierte Grammatik	6
1.4. Attributabhängigkeiten	14
1.5. Einsatz bei der Entwicklung von Compiler-Compilern	16
1.6. Historische und bibliographische Anmerkungen	17
2. Attributierte Grammatiken – ein systematischer Überblick	19
2.1. Wohldefinierte Attributierte Grammatiken	19
2.1.1. Wohldefiniertheit	19
2.1.2. Zerlegbarkeit	22
2.1.3. Problematik von Attributierungsalgorithmen	25
2.1.4. Induzierte Attributabhängigkeiten	26
2.2. Zyklenfreiheit	29
2.2.1. Verfahren zum Test auf Zyklenfreiheit nach Knuth	29
2.2.2. Verfahren zum Test auf Zyklenfreiheit nach Jazayeri	32
2.2.3. Zeitkomplexität des Circularity-Problems	34
2.2.4. Zyklenfreiheitstest nach Chebotar	43
2.3. 1-Pass-Attributierte Grammatiken	46
2.3.1. L-Attributierte Grammatiken	46
2.3.2. 1-Visit-Attributierte Grammatiken	50
2.3.3. S-Attributierte Grammatiken	54
2.3.4. LL/LR-Attributierte Grammatiken	56
2.4. Multi-Pass-L/R-Attributierte Grammatiken	58
2.4.1. Die LAG-Eigenschaft	58
2.4.2. Algorithmus zum Test der LAG-Eigenschaft	60
2.4.3. Links-Rechts-Attributierungsalgorithmus	62
2.4.4. Die RAG-Eigenschaft	63
2.4.5. Vergleich von LAG und RAG	65
2.5. Alternierende Attributierte Grammatiken	66
2.5.1. Die AAG-Eigenschaft	66
2.5.2. Algorithmen für Alternierende Attributierte Grammatiken	68
2.5.3. Vergleich von Multi-Pass-Attributierten Grammatiken	72
2.5.4. Bestimmung der optimalen Richtungsfolge	77
2.5.5. Markierter Attributabhängigkeitsgraph	82

---

2.6. Multi-Sweep-Attributierte Grammatiken	91
2.7. Multi-Visit und Geordnete Attributierte Grammatiken	94
2.7.1. Multi-Visit-Attributierte Grammatiken	94
2.7.2. Attributzerlegungen für Geordnete AG	98
2.7.3. Geordnete Attributierte Grammatiken	100
2.7.4. Besuchsfolgen für Geordnete Attributierte Grammatiken	105
2.7.5. Beispiel von Kontextbeziehungen in Programmiersprachen	109
2.7.6. Attributierung von Multi-Visit-AG und Geordneten AG	114
2.7.7. Inkrementelle Attributierung von Geordneten AG	116
2.8. Reine Attributierte Grammatiken	119
2.8.1. Reine Multi-Pass-Attributierte Grammatiken	119
2.8.2. Reine Multi-Sweep-Attributierte Grammatiken	123
2.8.3. Reine Multi-Visit-Attributierte Grammatiken	125
2.9. Absolut zyklenfreie Attributierte Grammatiken	130
2.9.1. Charakterisierung von Absolut zyklenfreien AG	130
2.9.2. Der Attributierungsalgorithmus von Kennedy-Warren	142
2.9.3. Konstruktion eines Kennedy-Warren-Algorithmus	144
2.9.4. Automatische Konstruktion eines KW-Algorithmus	149
2.9.5. Rekursive Attributierung von Absolut zyklenfreien AG	154
2.10. Dynamische Attributierung	159
2.10.1. Einfache Dynamische Attributierung	159
2.10.2. Inkrementelle Dynamische Attributierung	162
2.10.3. Optimale Dynamische Attributierung	165
2.11. Algebraische Definition einer Attributierten Grammatik	171
2.11.1. Attributierte Grammatik als heterogene Algebra	171
2.11.2. Stetige Attributierte Grammatik und Fixpunktattributierung	175
2.11.3. Äquivalenz von Attributierten Grammatiken	178
2.11.4. Elimination von ererbten Attributen	181
2.12. Affix-Grammatiken	184
2.13. Hierarchie der Attributierten Grammatiken	189
3. Einsatz von AG in Compiler-Erzeugenden Systemen	191
3.1. ALADIN – eine applikative Sprache zur Beschreibung einer AG	191
3.2. GAG – ein Compilergenerator für Geordnete AG	196
3.3. HLP78 – ein Compilergenerator für Alternierende AG	208
3.4. FNC/ERN	218
3.5. COCO – ein Compiler-Compiler für Mikrocomputer	226
3.6. LINGUIST-86 – ein kommerziell genutzter Compiler-Compiler	229
3.7. Weitere Compilergeneratoren	231
3.8. Übersicht über Attributierungen von Programmiersprachen	235

4. Ausblick	236
5. Anhang	239
5.1. Übersetzung englischer Fachausdrücke	239
5.2. Bibliographie über Attributierte Grammatiken	242
5.3. Sonstige Literatur	271