Programmiertechniken in der Computerlinguistik I

Wintersemester 2003/2004

Inhaltsverzeichnis

1. Organisatorisches
2. Literaturhinweise
3. Einführung
4. Fakten, Regeln, Anfragen
5. Syntax und Datenstrukturen
6. Beweisen
7. Occur Check
8. Debuggen (Fehlersuche)
9. Arithmetik
10. Operatoren
11. Daten- und Kontrollfluss
12. Listen
13. Rekursive Listenverarbeitung
14. Ein- und Ausgabe
15. Rekursive Programmiertechniken
16. Definite Clause Grammar (DCG)
17. Definite Clause Grammar II (DCG II)
18. Term-Prädikate
19. Shift-Reduce-Parsing
20. Mengen-Prädikate

!/0: 11.13f
": 14.14
%: 5.24
*/2: 9.4
,/2: 5.7, 10.5
,: 5.17
./2: 12.16f
.: 5.5
///2: 9.4
//2: 5.20, 9.4
-/2: 9.4
:-/1: 5.8, 10.5
:-/2: 5.6, 10.5
;/2: 5.7, 10.5, 11.6
?-/1: 5.8, 10.5
@</2: 18.6
@=</2: 18.7
@>/2: 18.7
@>=/2: 18.7
[]: 12.5
\+/1: 11.10, 11.19
\==/2: 18.5
^/2: 20.6
_: 5.14
{/1: 17.14
|: 12.11
}/1: 17.14
+/2: 9.4
</2: 9.7
=../2: 18.2
=/2: 6.12f
=:=/2: 9.7
=\=/2: 9.7
=</2: 9.7
==/2: 18.5
-->/2: 16.7
>/2: 9.7
>=/2: 9.7
abs/1: 9.4
append/3: 15.11f
arg/3: 18.4
atom/1: 5.12
atom_chars/2: 14.13
atom_codes/2: 14.13
atomic/1: 5.10
bagof/3: 20.5
'C'/3: 16.15
call/1: 11.18
compound/1: 5.18
debug/0: 8.10
fail/0: 11.8
findall/3: 20.2f
float/1: 5.13
functor/3: 18.3
get/1: 14.8
get0/1: 14.9
integer/1: 5.13
is/2: 9.3
is_list/1: 13.4
length/2: 13.8
listing/0-1: 8.2
member/2: 13.9f
mod/2: 9.4
name/2: 14.12
nl/0: 14.10
nodebug/0: 8.10
nospy/1: 8.10
notrace/0: 8.6
number/1: 5.12
number_chars/2: 14.13
number_codes/2: 14.13
op/3: 10.6
phrase/2: 16.9
pop/3: 19.9
push/3: 19.9
put/1: 14.7
read/1: 14.16
repeat/0: 14.23
reverse/2: 15.16f
round/1: 9.4
see/1: 14.20
seeing/1: 14.21
seen/0: 14.20
setof/3: 20.7
sort/2: 18.8
spy/1: 8.10
tab/1: 14.10
tell/1: 14.18
telling/1: 14.19
told/0: 14.18
trace/0: 8.6
unify_with_occurs_check/2: 7.3
user/0: 14.24
var/1: 5.16
write/1: 14.17
write_canonical/1: 10.7, 14.17