

Präprozessorproblematik

```
1  #ifdef CONDITION
2      typedef int t;
3      typedef int u;
4  #else
5      int t(int i) {};
6      int u;
7  #endif
8
9  int x,y;
10
11 int main() {
12     t(x);
13     u * y;
14 }
```

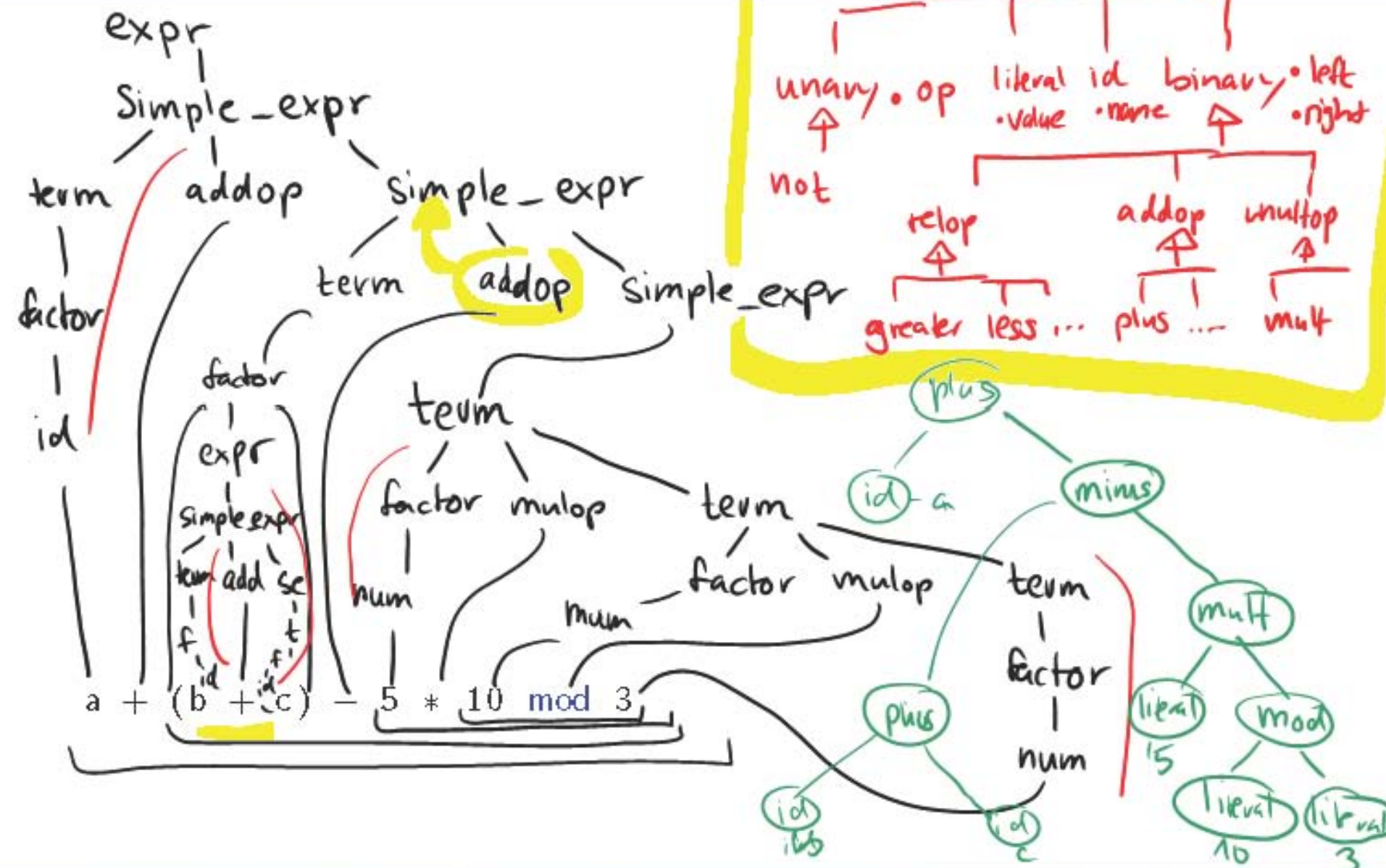
Erstellt mit einer Testversion von PDF Annotator - www.PDFAnnotator.com

Abstrakte Syntax

```
expr      ::= simple_expr [relop expr] .
relop     ::= "=" | "<" | "<=" | ">=" | "<" | ">" .
simple_expr ::= term [addop simple_expr] .
addop     ::= "+" | "-" | "or" .
term      ::= factor [mulop term] .
mulop     ::= "*" | "/" | "mod" | "and" .
factor    ::= num | id
           | "(" expr ")"
           | "not" factor .
num       ::= ["-"] digit {digit} ["." digit {digit}] .
digit     ::= "0" | "1" | "2" | "3" | ... | "9" .
id        ::= letter {letter} .
letter    ::= "a" | "b" | "c" | ... | "z" .
```

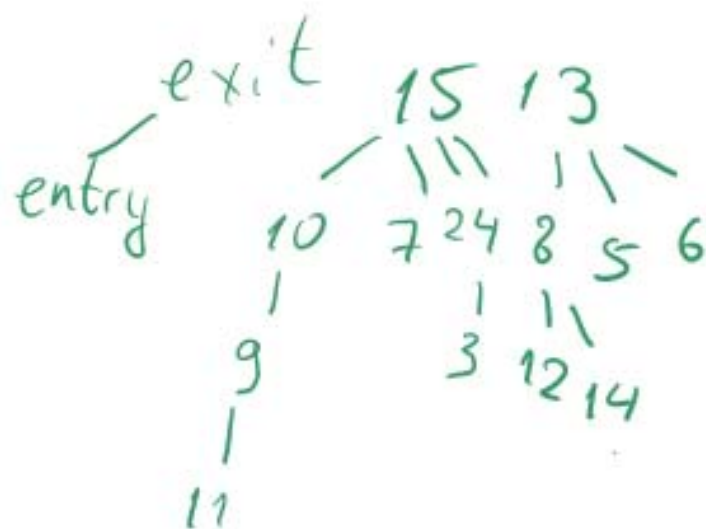
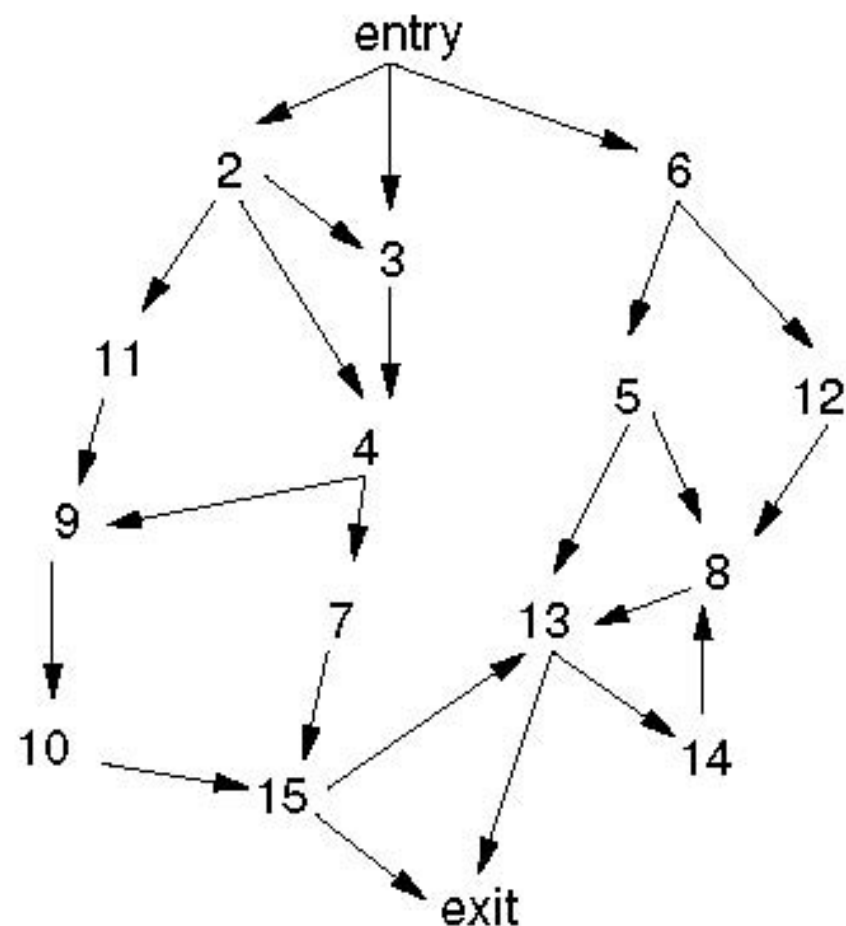
Erstellt mit einer Testversion von PDF Annotator - www.PDFAnnotator.com

Abstrakte Syntax



Erstellt mit einer Testversion von PDF Annotator - www.PDFAnnotator.com

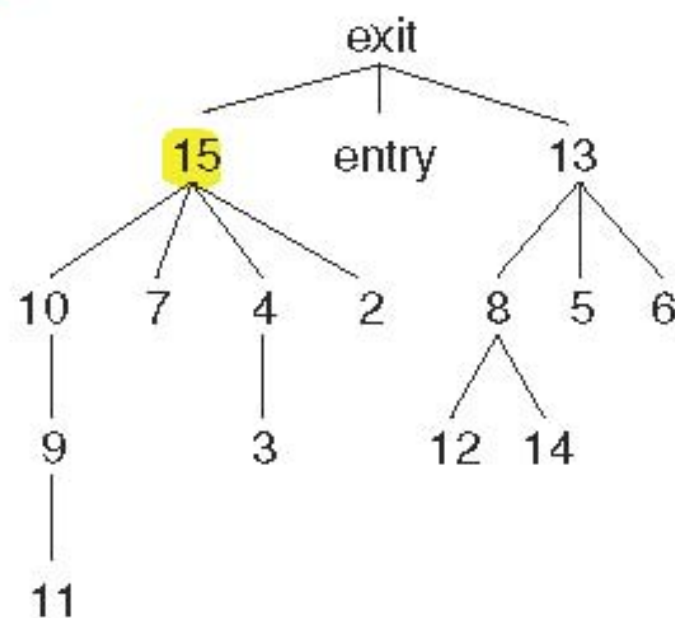
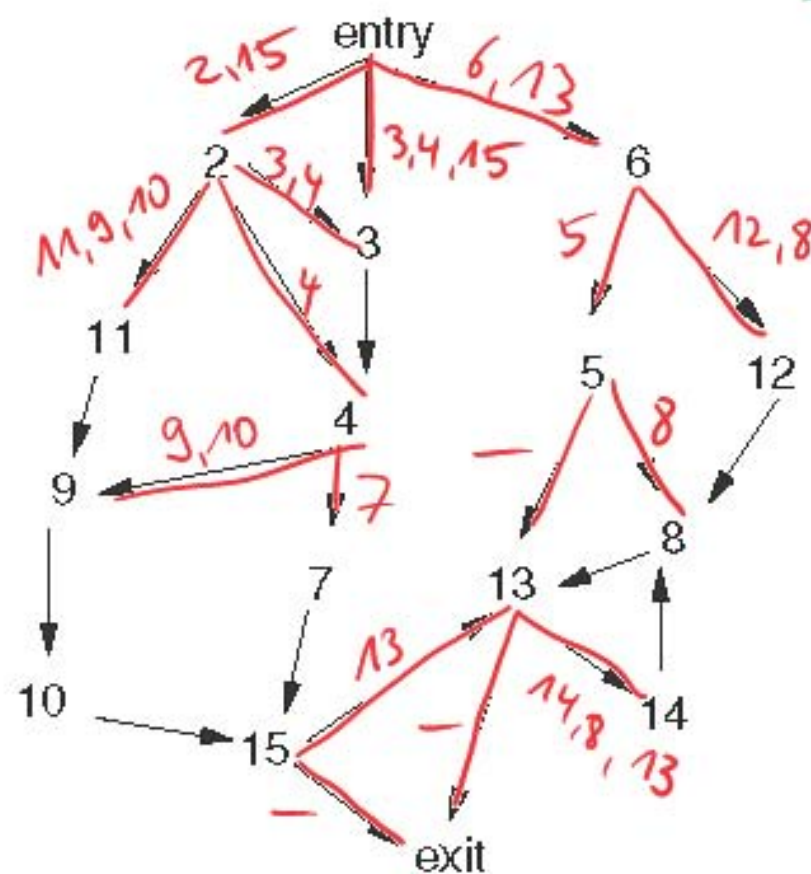
Kontrollflussgraph



Erstellt mit einer Testversion von PDF Annotator - www.PDFAnnotator.com

Postdominanzbaum

$B \rightsquigarrow pdom(B)$
 $B \rightsquigarrow S$



B	S	pdom(B)	cdeps
e	2	exit	2, 15
e	3	exit	4, 15, 3
e	6	exit	6, 13
2	11	15	11, 9, 10
2	4	15	4
2	3	15	3, 4
4	9	15	9, 10
4	7	15	7
15	13	exit	13
15	exit	exit	–
6	5	13	5
6	12	13	12
5	13	13	–
5	8	13	6
13	14	exit	14, 8, 13
13	exit	exit	–

Intra- und interprozeduraler Kontrollfluss + SSA

```
1  int sum, prod;
2
3  void foo (int value) {
4      if (value > 0) {
5          sum = sum + value;
6          prod = prod * value;
7          foo (value - 1);
8      }
9  }
10
11 int main () {
12     sum = 0;
13     prod = 1;
14     foo (10);
15     print (sum);
16     print (prod);
17 }
```

Erstellt mit einer Testversion von PDF Annotator - www.PDFAnnotator.com

Globale Variable zu Parameter

```
1 void foo (int in value, int in out su, int in out pr) {  
2     if (value > 0) {  
3         su = su + value;  
4         pr = pr * value;  
5         foo (value - 1, su, pr);  
6     }  
7 }  
8  
9 int main () {  
10     int sum, prod;  
11  
12     sum = 0;  
13     prod = 1;  
14     foo (10, sum, prod);  
15 }
```


Erstellt mit einer Testversion von PDF Annotator - www.PDFAnn

```

1 void foo () {
2     value = valuein; su0 = suin; pr0 = prin;
3     if (value0 > 0) {
4         su1 = su0 + value0;
5         pr1 = pr0 * value0;
6         valuein = value0 - 1; suin = su1; prin = pr1;
7         foo;
8         su2 = suout; pr2 = prout;
9     }
10    suout = su3; prout = pr3;
11 }
12 int main () {
13     int sum, prod;
14     sum0 = 0;
15     prod0 = 1;
16
17     valuein = 10; suin = sum0; prin = prod0;
18     foo;
19     sum1 = suout; prod1 = prout;
20 }

```

$su_3 = \emptyset (su_0, su_2)$ $pr_3 = \emptyset (pr_0, pr_2)$

Erstellt mit einer Testversion von PDF Annotator - www.PDFAnn

