

KV Software Engineering

Übungsaufgaben SS 2005

Harald Gall, Silvio Meier,
Nancy Merlo-Schett, Katja Gräfenhain

Übung 5 – Statische Programmanalyse

Aufgabe 1 (6 Punkte)

Gegeben sei folgendes Programmstück in SPL (Stupid Programming Language).

```
1   int num;
2   boolean odd;
3
4   num = 10;
5
6   read num;
7
8   if (num % 2 != 0)
9       odd = true;
10  else
11      odd = false;
12
13  write odd;
```

a) Zeichnen Sie den abstrakten Syntaxteilbaum (AST) für das if-else-Statement in dem gegebenen Code. Verwenden Sie dafür beiliegende SPL-Grammatik. (3 Punkte)

b) Definieren Sie einen Flussgraphen. Repräsentieren Sie im Graphen die Kontrollabhängigkeiten und die Datenabhängigkeiten. (3 Punkte)

Aufgabe 2 (14 Punkte)

Gegeben sei folgendes C-Programm:

```
1  #define YES 1
2  #define NO 0
3  main()
4  {
5      int c, nl, nw, nc, inword;
6      inword = NO;
7      nl = 0;
8      nw = 0;
9      nc = 0;
10     c = getchar();
11     while ( c != EOF ) {
12         nc = nc + 1;
13         if ( c == '\n' )
14             nl = nl + 1;
15         if ( c == ' ' || c == '\n' || c == '\t' )
16             inword = NO;
17         else if ( inword == NO ) {
18             inword = YES;
19             nw = nw + 1;
20         }
21         c = getchar();
22     }
23     printf("%d \n", nl);
24     printf("%d \n", nw);
25     printf("%d \n", nc);
26 }
```

a) Welche Set-Use, Use-Set, Set-Set Beziehungen gibt es im Programmstück? (3 Punkte)

b) Definieren Sie einen Program Dependency Graphen (PDG) dafür. (5 Punkte)

c) Definieren Sie den Forward Slice $S_1(6, \{inword\})$ für $inword=NO$. Suchen Sie dazu in Ihrem Program Dependency Graph den entsprechenden Pfad und zeichnen Sie diesen ein. Streichen Sie alle Elemente im Quellcode, die nicht erreicht werden. (2 Punkte)

d) Definieren Sie den Backward Slice $S_2(24, \{nw\})$ für `printf("%d \n", nw);` Suchen Sie dazu in Ihrem Program Dependency Graph den entsprechenden rückgerichteten Pfad und zeichnen Sie diesen ein. Streichen Sie alle Elemente im Quellcode, die nicht erreicht werden. (4 Punkte)