



Aufgabenblatt 13

Besprechung am Dienstag, den 14. Februar 2006, 8:00 Uhr V38.01

Aufgabe 1 (Axiomatische Semantik)

```
Get(x);  
y:=0;  
while (y<x) loop  
  y:=y+1;  
end loop;
```

Zeigen Sie unter Verwendung der Schleifeninvariante $\{y \leq x\}$, dass

$\{x < y\} \langle \text{Schleife} \rangle \{x = y\}$

gilt (x und y sind vom Typ *natural*).

Aufgabe 2

Gegeben ist folgendes kleines Programmfragment (x und y sind vom Typ *integer*):

```
x:=x-y;  
y:=x+y;  
x:=y-x;
```

Beschreiben Sie umgangssprachlich die Funktion dieses Fragments. Spezifizieren Sie nun formal die Vor- und Nachbedingung. Weisen Sie die Korrektheit des Fragments nach.

Aufgabe 3 (Axiomatische Semantik)

```
Get(n);  
i:=0;  
Erg:=0  
while (i<n) loop  
(1)  Erg:=Erg+i;  
(2)  i:=i+1;  
end loop;
```

Finden Sie eine Schleifeninvariante, mit deren Hilfe Sie die realisierte Abbildung beschreiben können (n ist vom Typ *natural*).

Vertauschen Sie nun die Zeilen (1) und (2). Was ändert sich? Was passiert, wenn man Erg mit 1 initialisiert?

Aufgabe 4

Gegeben seien folgende Programmstücke (x und y sind vom Typ *natural*).

```
Get(x);  
Erg:=0;  
while ((Erg + 1)* (Erg + 1))<= x) loop  
    Erg := Erg + 1;  
end loop;
```

--

```
Get(x); Get(y); Erg:=0;  
while (x>=y) loop  
    x:=x-y; Erg:=Erg+1;  
end loop;
```

--

```
Get(x);  
y:=x;  
while (y>0) loop  
    y:=y-1;  
    x:=x*y;  
end loop;
```

Was berechnen diese Programme? Zeigen Sie Ihre Vermutung mittels der Hoareschen Regeln.

Aufgabe 5

```
function exp(x,y:natural):natural;  
var Erg:natural;  
begin Erg:=1;  
  while y>0 loop  
    if y mod 2 = 0 then  
      y:=y div 2;  
      x:=x*x;  
    elsif  
      y:=y-1;  
      Erg:=Erg*x;  
    end if;  
  end loop;  
  return(Erg);  
end;
```

Beweisen Sie mit geeigneten Zusicherungen und den Hoareschen Regeln, dass die Funktion $\text{exp } x^y$ berechnet.

Aufgabe 6 (ehemalige Prüfungsaufgabe)

Das folgende Programm Aufgabe6 bestimmt die Summe aller geraden Zahlen von 2 bis X. Sie sollen zeigen, dass das Programm Aufgabe6 bezüglich der Spezifikation

$$\{X \geq 0 \text{ und } X \text{ gerade}\} \langle \text{Aufgabe6} \rangle \{SUM = (X * X + 2 * X) / 4\}$$

partiell korrekt ist. Partiiell korrekt heißt: Korrekt, unter der Voraussetzung, dass das Programm terminiert.

```
program Aufgabe6 is
sum, x, z : integer;
begin
  Get(x);
  sum := 0;
  z := 2;
  while z < (X+2) loop
    sum := sum + z;
    z := z + 2;
  end;
end.
```

Aufgabe 7 (weakest Precondition)

Gegeben seien die folgenden Fallunterscheidungen und deren Nachbedingung. Bestimmen Sie die Vorbedingung für die gegebenen Fallunterscheidungen.

- $z = x;$ if $(z < y)$
 $z = y;$
 $\{z = \max(x, y)\}$
- if $(x < y)$
 $a = x;$
else
 $a = y;$
 $\{a = \min(x, y)\}$

Aufgabe 8 (weakest Precondition)

Bestimmen Sie die *weakest Precondition* für die folgenden Wertzuweisungen bei angegebener Nachbedingung:

- $wp(a = a + 1; a < b)$
- $n = n + m;$
 $\{n > 0\}$
- $wp(n = m + n; a > 0)$

Aufgabe 9

Leiten Sie die *schwächste Vorbedingung* (*weakest precondition*) wp aus der entsprechenden *Nachbedingung* (*postcondition*) für folgendes Programmstück ab:

```
a, b: INTEGER
a := 2 * b - 2;
if a < 16 then
  b := 2 * a - 16;
else
  b := a+8;
end if;
b := b + 4;
{ Q ≡ 8 ≤ b ≤ 20 }
```

Allgemeine Hinweise:

- Bei weiteren Fragen, wenden Sie sich bitte an W. Schmid (sltsoftware@yahoo.de).
- Weitere Hinweise finden Sie auf unserer Veranstaltungswebseite unter:
<http://www.info2.de.vu>
<http://www.zusatzkurs.de.vu>