Prof. Dr. Hans-Jörg Kreowski, Dr. Sabine Kuske
Studiengang Informatik
Linzer Str. 9a
OAS 3001, 3005
Tel.: 2956,2335, 3697 (Sekr.), Fax: 4322
E-Mail: {kreo,kuske}@informatik.uni-bremen.de
www.informatik.uni-bremen.de/theorie

21. Januar 2006
Arbeitsbogen 6

Theoretische Informatik I

Arbeitsbogen zu while-Programmen

Um den Umgang mit und das Erstellen von *while*-Programmen zu erleichtern, ist es sinnvoll, oft benutzten Konstrukten Namen zu geben, sich also sozusagen Abkürzungen für Programmstücke zu definieren. Diese Abkürzungen heißen *Makro-Anweisungen*. Ein Beispiel ist die Makro-Anweisung $X := Y$, die definiert ist als

$$\mathit{begin}\ X := \mathit{succ}(Y);\ X := \mathit{pred}(X)\ \mathit{end}.$$

Nachdem wir dies jetzt definiert haben, können wir Anweisungen der Form $X := Y$ wie jede andere Anweisung in *while*-Programmen benutzen. Insbesondere kann man sie zur Definition weiterer Makro-Anweisungen einsetzen, zum Beispiel für $Z := X + Y$, was definiert ist als

begin
  $U := X;\ V := 0;$
  while $V \neq Y$ *do*
  begin
    $U := \mathit{succ}(U);$
    $V := \mathit{succ}(V)$
  end;
  $Z := U$
end

Hierbei sind U und V Hilfsvariablen, von denen wir annehmen, dass sie immer so gewählt werden, dass sie unter den ansonsten benutzten Variablen eines Programms nicht vorkommen. Dies kann ohne Beschränkung der Allgemeinheit angenommen werden, da in jedem Programm nur endlich viele der potentiell zur Verfügung stehenden Variablen $X1, X2, \ldots$ vorkommen und daher $U = Xi$ und $V = Xj$ immer geeignet gewählt werden können.

1. Definiere die folgenden Operatoren als Makro-Anweisungen:

   (a) $X := n$ für eine beliebige natürliche Zahl n,

   (b) die Multiplikation $Z := X \cdot Y$,

   (c) die ganzzahlige Division $Z := X\ \mathit{div}\ Y$ (wobei $X\ \mathit{div}\ 0$ undefiniert sein soll),

   (d) *if* $X \neq Y$ *then* S_1 *else* S_2, wobei S_1 und S_2 Anweisungen sind.

   Dabei kann jeweils von den bereits definierten und den oben erwähnten Makro-Anweisungen sowie der Makro-Anweisung $Z := X \dot{-} Y$ aus Aufgabe 3 Gebrauch gemacht werden (letztere könnte sich insbesondere für *div* als nützlich erweisen).

2. Die Menge aller *Tests* sei rekursiv wie folgt definiert:

   - $X < Y$ ist ein Test.
   - Für Tests C und C' sind $\neg C$, $C \vee C'$ und $C \wedge C'$ ebenfalls Tests.

   Definiere für jeden Test C die Wiederholung *while* C *do* S als Makro-Anweisung. (Hinweis: Definiere u.a. für jedes C eine Makro-Anweisung $U := E(C)$, wobei $E(C)$ den Wert 1 liefert, falls C wahr ist und ansonsten den Wert 0 hat.)

3. Zeige, dass *pred* in PASCALchen nicht benötigt wird, dass also jede berechenbare Funktion schon durch ein *while*-Programm berechnet wird, in dem *pred* nicht benutzt wird.

4. Die folgende Makro-Anweisung $Z := X \dot{-} Y$ weist Z die Differenz der Werte von X und Y zu, falls diese größer als Null ist, und 0 sonst.

   begin
     $Z := X;\ V := Y;\ W := 0;$
     while $V \neq W$ *do* *begin*
       $Z := \mathit{pred}(Z);$
       $V := \mathit{pred}(V)$
     end
   end

   Beachte, dass dieses *while*-Programm bereits eine Makro-Anweisung $X := Y$ benutzt, welche der Variablen X den Wert von Y zuweist.

   (a) Gib für dieses *while*-Programm eine Berechnung $a_0 A_1 a_1 \cdots A_n a_n$ an, wobei $(Z, X, Y, V, W) = (X1, \ldots, X5)$, $a_i = (z_i, x_i, y_i, v_i, w_i)$ für $i = 0, \ldots, n$ und $y_0 > 0$.

   (b) Für jede Berechnung $a_0 A_1 a_1 \cdots A_n a_n$ dieses *while*-Programms weise folgende Behauptungen nach:

      i. Sei A_i der Test $V \neq W$. Dann gilt $z_i \dot{-} v_i = x_0 \dot{-} y_0$.

      ii. $z_n = x_0 \dot{-} y_0$.