

Funktionen in Pascal

1.1 Vordefinierte Funktionen in Pascal

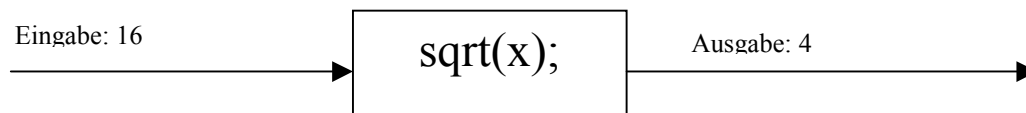
In der Programmiersprache Pascal sind einige Funktionen, z.B. für mathematische Anwendungen, vordefiniert und können beim Programmieren verwendet werden.

Ein Beispiel ist die Quadratwurzelfunktion:

```

program VordefinierteFunktionen;
uses wincrt;
var a      :integer;
    wurzel :real;
begin
write('Bitte geben Sie eine Zahl ein: ');
readln(a);
wurzel := sqrt(a);
write('Die Quadratwurzel von ',a,' beträgt',wurzel);
end;
    
```

Eine solche Funktion kann man sich als ein Unterprogramm vorstellen, das eine bestimmte Eingabe erhält und ein Ergebnis als Ausgabe zurückliefert

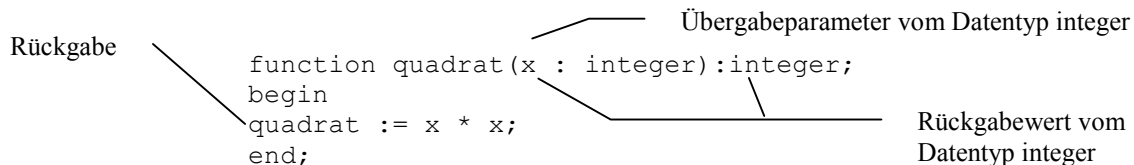


1.2 Selbstdefinierte Funktionen

Es gibt in der Programmiersprache Pascal jedoch auch die Möglichkeit, selbst Funktionen zu definieren.

Eine Funktion ist – wie bereits erwähnt wurde - ein Unterprogramm, das einen Rückgabewert liefert.

Dementsprechend muß man bei der Deklaration von Funktionen die Eingabewerte (Übergabeparameter) und die Ausgabewerte (Rückgabewert) genau deklarieren, d.h. mit einem Bezeichnernamen und einem Datentyp versehen. Das folgende Beispiel demonstriert dies:



Der Aufruf der Funktion im Hauptprogramm erfolgt dann mit Hilfe ihres Funktionsnamens und der Übergabe der Parameter.

```

ErsteZahl := 2;
ergebnis := quadrat(ErsteZahl);
writeln(ergebnis);
ZweiteZahl := 4;
writeln(quadrat(ZeiteZahl));
    
```

1.3 Rekursive Funktionen

Eine Funktion, die sich selbst aufruft wird rekursive Funktion genannt.

```

function pot(a,n:integer):integer;
begin
if n=0 then pot := 1
else pot := pot(a,n-1) * a;
end;
    
```

Programme gewinnen durch die Verwendung von Funktionen und Prozeduren deutlich an Übersichtlichkeit. Zudem verringert sich die Fehleranfälligkeit. Außerdem ist die Wiederverwendbarkeit der Prozeduren und Funktionen von großer Bedeutung. Dadurch werden Programme um vieles kürzer.

1.4 Aufgaben zum Thema

Aufgabe:

Schreiben Sie ein Pascal-Programm, in dem die Funktionen

```
add(a,b)
sub(a,b)
mul(a,b)
quad(a)
wurz(a)
divv(a,b)
modd(a,b)
```

implementiert werden. Nutzen Sie den folgenden Programmaufbau.

```
program MathematikFunktionen;
uses wincrt;
var      a,b           : integer;
         IntErgebnis  : integer;
         RealErgebnis : real;

{Deklaration der Funktionen}
function add(a:integer,b:integer):integer;
begin

end; {of function add}

function sub(a:integer,b:integer):integer;
begin

end; {of function sub}

usw....

begin {Hauptprogramm}

{Eingabe der Daten}

write('Bitte geben Sie die erste Zahl a ein: ');
readln(a);
write('Bitte geben Sie die zweite Zahl b ein: ');
readln(b);

{Berechnung und Ausgabe der Ergebnisse}

write('Die Summe von ',a,' und ',b,' beträgt:');
IntErgebnis:=add(a,b);
writeln(IntErgebnis);

write('Die Differenz von ',a,' und ',b,' beträgt:');
IntErgebnis:=sub(a,b);
writeln(IntErgebnis);

end. {Hauptprogramm}
```