

Rekursion in Pascal

Was ist Rekursion

Als Rekursion bezeichnet man das Verfahren, dass eine Funktion ihren Rückgabewert durch Aufruf von sich selbst berechnet. Dabei muss die Funktion mindestens einen Startwert entgegennehmen und diesen in jedem Durchgang wieder neu berechnen. Damit dies kein endloser Vorgang wird, muss weiterhin sichergestellt sein, dass die Rekursion bei Erreichen eines festgelegten Startwertes oder Ergebnisses innerhalb der Berechnung endet.

Was bedeutet das aber?

Beispiel anhand Fakultät

$$n! = n * (n - 1) * (n - 2) * (n - 3) * \dots$$

$$5! = 5 * 4 * 3 * 2 * 1 = 120$$

Fakultät in Pascal berechnen ohne Rekursion (Iterativ)

```
function FakultaetIterativ(n: Integer): Int64; //Int64 da hohe Zahlen
var
    i: Integer;
    erg: Int64;
begin
    erg := 1; //Anfangswert 1, weil 1 das neutrale Element der Multiplikation ist
    for i := 2 to n do //Schleife von 2 bis n (1 brauch man nicht weil erg=1)
        erg := erg * i; //Multipliziere erg immer mit der nächsten Zahl.
    Result := erg; //Ergebnis zurückgeben
end;
```

Fakultät in Pascal berechnen mit Rekursion

```
function FakultaetRekursiv(n: Integer): Int64; //Auch wieder Int64 wegen hohen Zahlen
begin
    if n <= 1 then //Wenn n kleiner oder gleich 1 ist, ist die Fakultät 1
        Result := 1
    else
        Result := n * FakultaetRekursiv(n - 1); // Sonst n mal Fakultät von n - 1 bis n = 1
    end;
```

Visuelles Beispiel anhand Fakultät.exe

<https://tjh.li/files/gfs.zip>

<https://tjh.li/files/code.txt>

Vor und Nachteile

Positiv

Macht manche Probleme einfacher wie
Türme von Hanoi

Code kürzer und übersichtlicher

Negativ

Braucht mehr Speicher durch
wiederholtes aufrufen der Funktion

In vielen Fällen langsamer

DANKE FÜR ZUHÖREN.



NOCH FRAGEN?