Compiler

Ein Compiler, von Englisch "to compile – zusammenführen", übersetzt Programmcode in für den Computer lesbare Sprache. Dabei besitzen alle Compiler 3 Grundphasen.

1. Phase: Der Lexer zerlegt den Programmcode in sogentannte Lexeme, die den Code gruppiereren. Ein Lexem könnte zum Beispiel eine Ganzzahl, eine Zeichenkette oder ein Sonderzeichen sein.

2. Phase: Die Lexeme werden, basierend auf gramatikalischen Regeln, in einen Abstrakten-Syntax-Baum konvertiert. Dieser repräsentiert den Quellcode, mit der der Compiler arbeiten kann.

3. Phase: Der Syntaxbaum wird zu Befehlen konvertiert, die der Prozessor oder die Laufzeitumgebung ausführen können.

Heutige Compiler besitzen noch Zwischenschritte, wie zum Beispiel Codeoptimierungen.

Beispiel:

1. Folgender Code wird zu Lexemen gruppiert:

f(x) = x square 2

Identifier: f  
OpenBrace  
Identifier: x  
CloseBrace  
Whitespace  
Equals  
Whitespace  
Identifier: x  
Whitespace  
Identifier: square  
Whitespace  
Integer: 2  
EndOfLine

2. Nun die Erzeugung eines Baumes, basierend auf gramatikalischen Regeln: