Compiler

Ein Compiler, von Englisch "to compile – zusammenführen", übersetzt Programmcode in für den Computer lesbare Sprache. Dabei besitzen alle Compiler 3 Grundphasen. Folgender Code soll als Beispiel übersetzt werden:

f(x) = x exponent 2

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.

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

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. Wie die Regeln angewendet werden ist von Compiler zu Compiler unterschiedlich.

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.