程序设计语言的设计

**1、概念**

汇编器(Assembler)，编译器(Compiler)

变量，作用域、分配存储空间、类型

**解释器和编译器**：

在计算机科学中，解释器是一种计算机程序，它**直接执行由编程语言或脚本语言编写的代码，并不会把源代码预编译成机器码。**一个解释器，通常会用以下的姿势来执行程序代码：

（1）分析源代码，并且直接执行。   
（2）把源代码翻译成相对更加高效率的中间码，然后立即执行它。   
（3）执行由**解释器内部的编译器**预编译后保存的代码

解释器通常是工作在「运行时」，并且对于我们输入的源码，是一行一行的解释然后执行，然后返回结果。发生在运行时，产物是「运行结果」。

编译器产物是字节码或机器码等。

**2、C语言类型**

整型：4字节 (-2^31~2^31 – 1) int

长整型：long 8字节

字符型 char 2字节

单精度 float 4字节

双精度 double 8字节

类型转换

**3、程序语言的设计**

语法、语义、文法

**文法分类**：0型文法（全集）

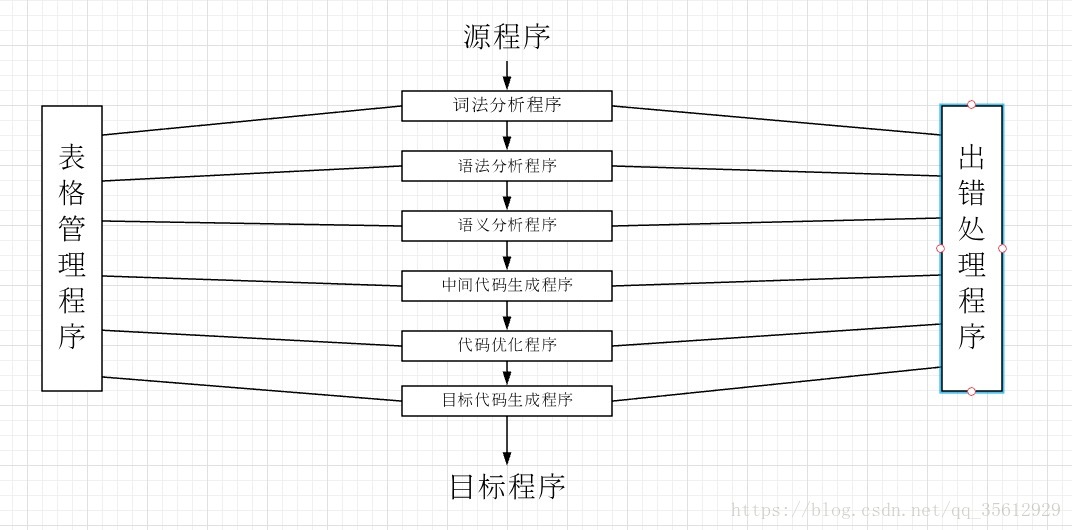
1型文法（上下文有关）

2型文法（上下文无关）

3型文法（正则文法）

语法树

程序设计语言的实现（编译）



**词法分析**：识别出符合词法规则的单词

**语法分析**：对词法分析识别出的符号流、按语法规则进行分析、识别各类语法单位（如表达式，短语）

**语义分析**：采用中间语言描述源程序的语义，**实现汇编命令。**

**优化、代码生成。**

词法分析：类似于分词。词嵌入。

语法分析：