1. 编译器是一个程序，核心功能是**把源代码翻译成目标代码**。输出是可执行程序（离线）
2. 解释器也是处理程序的一种程序，输出是程序运行的结果（在线 online）
3. 编译器具有非常模块化结构的高层系统，由多个阶段组成，每个阶段都要处理不同的问题：使用不同的理论、数据结构和算法；编译器设计中的重要问题是如何合理划分组织各个阶段。
4. 编译器的阶段：词法分析->语法树构建->代码生成（后序遍历）
5. 词法分析的任务：字符流转换到记号流；记号流是编译器内部定义的数据结构，编码所识别出的词法单元
6. 描述一个系统最重要的就是它的输入输出。
7. 词法分析器的实现方法：
   1. 手功编码实现法：相对复杂、且容易出错；主流的实现方法
   2. 词法分析器的生成器：可快速原型、代码量较少；难控制细节

8. 正则表达式：

e -> ε

-> c

-> e1 e2 连接

-> e1 | e2

-> e1 \* 闭包

有限状态自动机

1. 确定状态有限自动机DFA: 对任意的字符，最多有一个状态可以转移。（有向图）

2. 非确定的有限状态自动机NFA：对于任意的字符，有多于一个状态可以转移

RE->NFA:Thompson算法

1. 基于对RE的结构做归纳

* 对基本的RE直接构造
* 对复合的RE递归构造

# 《编译原理（龙书）》

## 第一章 引论

1. 环境是从一个名字到存储位置的映射。即从名字到变量的映射

2. 状态是一个从内存位置到它们值的映射