

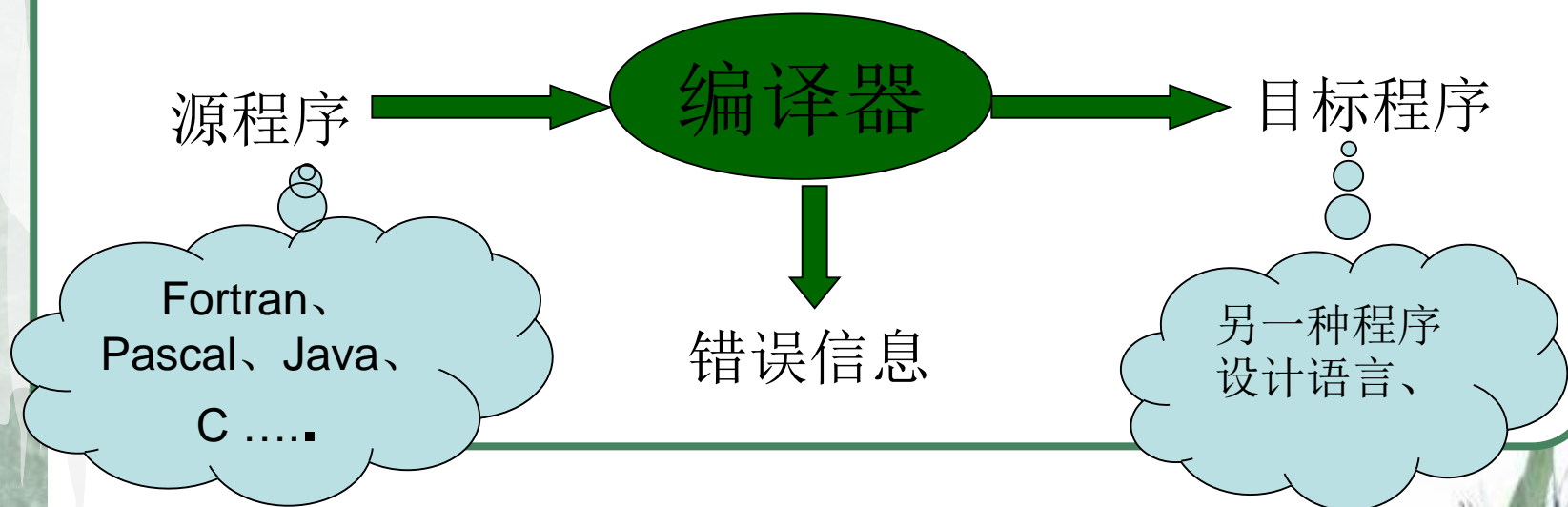


# 编译原理

## 第一章 引论

## 1.1 什么叫编译程序

- 编译器就是一个**程序**，它**读入**用某种语言编写的源程序，并**翻译**成一个**与之等价**的另一种语言编写的源程序。



## 1.2 编译过程概述

掌握编译过程的五个基本阶段，是我们学习编译原理课程的基本内容，把编译的五个基本阶段与英译中的五个步骤相比较，有利于对编译过程的理解：

### 翻译

识别出句子中的单词 ←  
分析句子的语法结构 ←  
根据句子的含义进行初步分析 ←  
对译文进行修饰 ←  
写出最后的译文 ←

### 编译

构成编译程序各个阶段

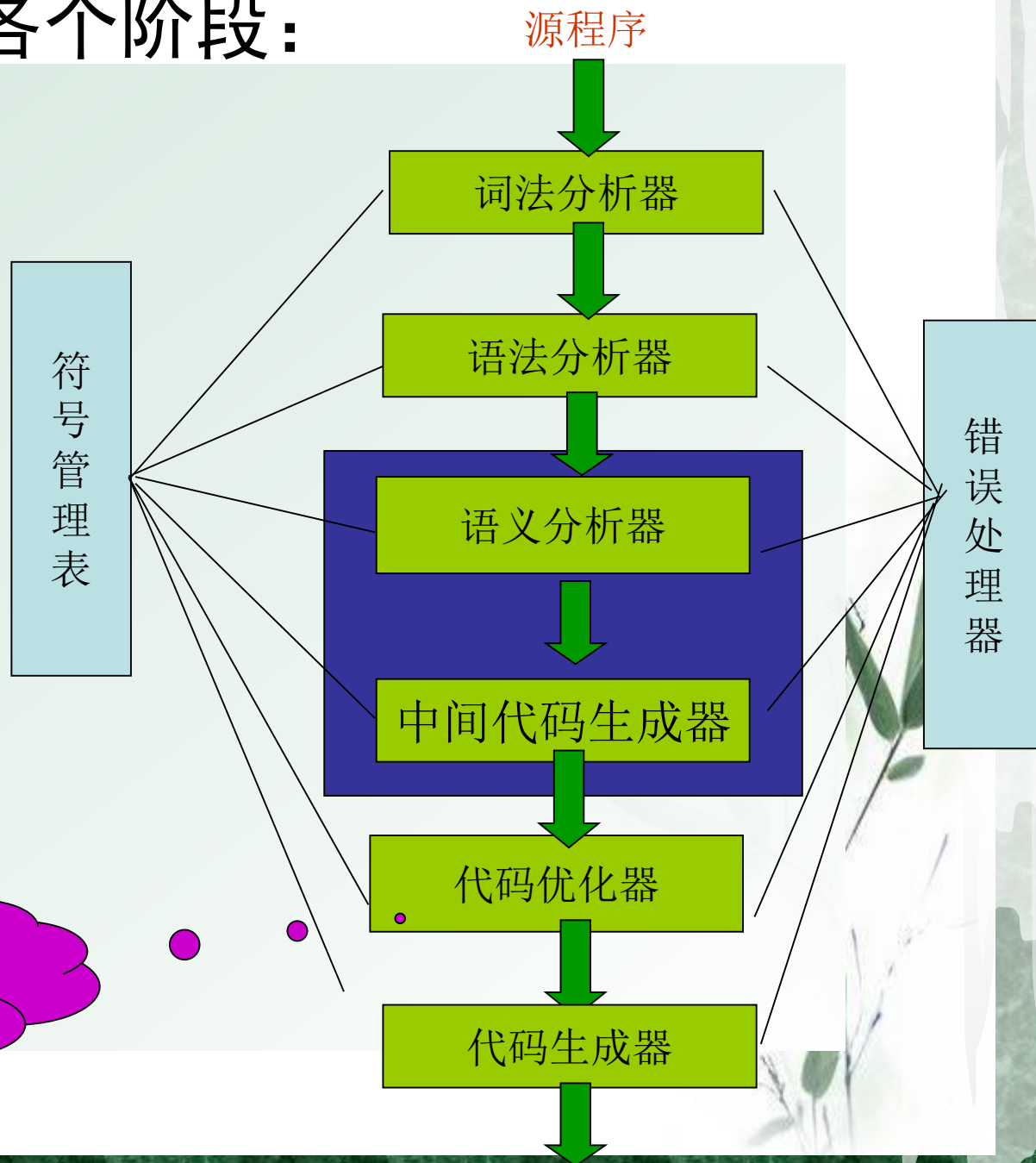
词法分析  
语法分析  
语义分析及中间代码生成  
代码优化  
目标代码生成

# 编译器的各个阶段：

编译器是分  
阶段执行的。

每个阶段将源程序从一种表示转换成另一种表示

编译的各个阶段



# 各分析阶段

随着编译器各个阶段的进展，源程序的内部表示不断地发生变化。

以  $a=b+c*d$  为例

## 1. 词法分析

完成任务： 读入源程序

识别出单词：  $a$ 、 $=$ 、 $b$ 、 $+$ 、 $c$ 、 $*$ 、 $d$

并用记号方式表示识别出的单词

输入源程序，对构成源程序的字符串进行扫描和分解，识别出一个个单词（也称单词符号，或简称符号）

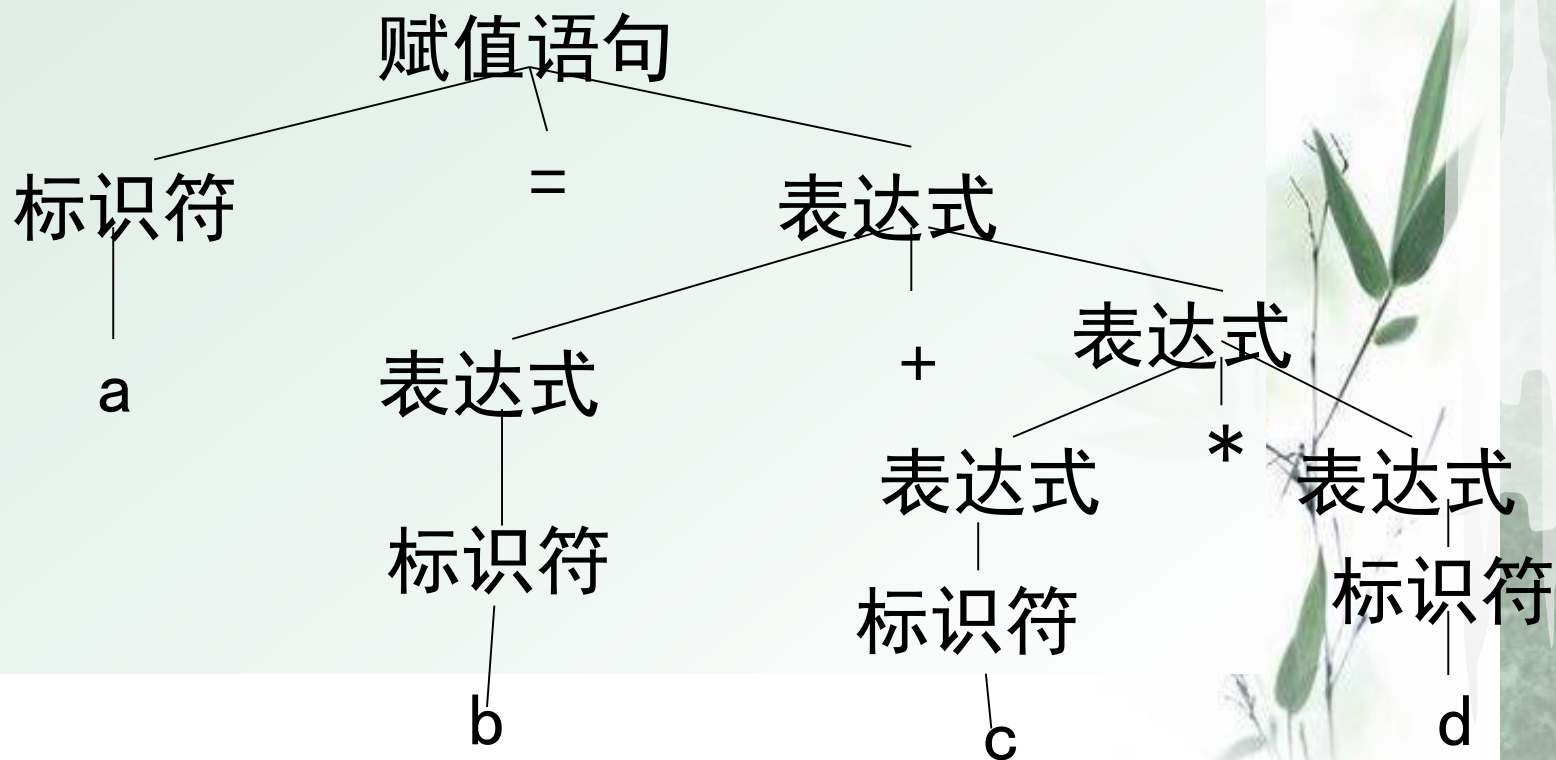
在词法分析阶段工作所依循的是语言的词法规则。描述词法规则的有效工具是正规式和有限自动机。



# 语法分析

在词法分析的基础上，根据语言的语法规则，把单词符号串组成各类语法单位。

具体的说，语法分析是在单词流的基础上建立一个层次结构——建立语法树



# 语义分析阶段

语义分析利用语法分析阶段确定的层次结构来识别表达式和语句中的操作信息及类型信息



# 中间代码生成阶段

本阶段将产生源程序的一个显式中间表示  
这种中间表示可以看成是某种抽象的程序，通常是与平台无关的

其重要性质：1. 易于产生

2. 易于翻译成目标程序

下面是用**三地址码**和四元式表示的例子：

**temp1=c\*d**            (\* , c , d , tempt1)

**temp2=b+temp1**    (+ , b, tempt1 , tempt2)

**a=temp2**            (= , tempt2 , , a)



# 代码优化阶段

试图改进中间代码，以产生执行速度较快的  
机器代码

对上面中间代码进行优化处理后，产生如下的  
代码：

```
temp1=c*d
```

```
temp2=b+temp1
```

```
a=temp2
```



```
temp1=c*d
```

```
a=b+temp1
```

# 代码生成阶段

生成可重定位的机器代码或汇编代码

Movf R2, c

Mulf R2, d

Movf R1, b

Addf R2, R1

Movf a, R2

# 1. 3. 3错误检测与报告

在编译的各个阶段都会发现源程序中的错误

为了使编译器能继续运行，以检测出源程序中更多的错误，在检测到错误后，必须以合适的方式进行错误处理。



# 1. 3. 5编译各阶段的分组

## 一、前端和后端

**前端**主要由与源语言有关但与目标机无关的那些部分组成。通常包括词法分析、语法分析、语义分析与中间代码产生，有的代码优化工作，也可以包括在前端。

**后端**包括编译程序中与目标代码有关的部分，如与目标机有关的有关的优化，和目标代码的生成等。

## 1.4 编译程序与程序设计环境

编译程序无疑是实现高级语言的一个最重要的工具。但支持程序设计人员进行程序设计开发通常还需要其它一些工具：如编辑程序、连接程序、调试程序等。编译程序与这些程序设计工具一起构成所谓的**程序设计环境**。

在一个程序设计环境中，编译程序起着中心的作用。连接程序、调试程序、程序分析等工具直接依赖于编译程序所产生的结果，而其它工具的构造也常常要用到编译的原理、方法和技术。





## 1.5 编译程序的生成

以前构造编译程序大多是用机器语言或汇编语言作工具的。为了充分发挥各种不同硬件系统的效率，为了满足各种不同的具体要求，现在许多人仍然使用这种工具来构造编译程序（或编译程序的核心部分）

但是越来越多的人已经使用高级语言作工具来编译程序。因为这样可以大大节省程序设计的时间，热切构造出来的编译程序易于阅读、维护和移植。

为此我们用T形图来表示源语言S、目标语言T和编译语言I之间的关系，如果A机器上已有一个用A机器码实现的某高级语言L1的编译程序，则我们可以用L1语言编写另一种高级语言L2的编译程序，把写好的L2编译程序经过L1编译程序编译后就可得到A机器代码实现的L2编译程序。



我们还可以采用“自编译方式”产生编译程序。方法是，先对语言的核心部分构造一个小小的编译程序（可用低级语言实现），再以他为工具构造一个能够编译更多语言成分的较大编译程序。如此扩展下去，就像滚雪球一样，越滚越大，最后形成人们所期望的整个编译程序。这种通过一系列的自展途径而形成编译程序的过程叫做**自编译过程**。