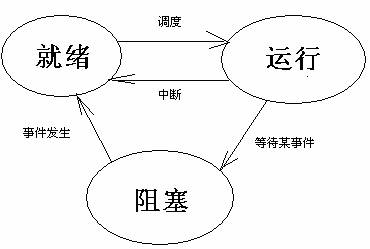
**1972，Edsger W.Dijkstra**

**PV操作：**

进程通常分为就绪、运行和阻塞三个工作状态。三种状态在某些条件下可以转换，三者之间的转换关系如图。进程三个状态之间的转换就是靠PV操作来控制的。



procedure p(var s:semaphore) {

s.value=s.value-1;

if (s.value<0) asleep(s.queue);

}

procedure v(var s:samephore){

s.value=s.value+1;

if (s.value<=0) wakeup(s.queue);

}

例如P1和P2是分别将数据送入缓冲B和从缓冲B读出数据的两个进程，定义两个信号量S1和S2，初值分别为1和0。

1. 进程P1在向缓冲B送入数据前执行P操作P（S1），在送入数据后执行V操作V（S2）
2. 进程P2在从缓冲B读取数据前先执行P操作P（S2），在读出数据后执行V操作V（S1）

**哲学家聚餐问题：**

设有5个哲学家，共享一张放有5把椅子的桌子，每人分得一把椅子，但是，桌子上共有5只筷子，在每人两边各放一只，哲学家们在肚子饥饿时才试图分两次从两边拿起筷子就餐。

**结构化程序设计：**

[自顶向下](https://www.baidu.com/s?wd=%E8%87%AA%E9%A1%B6%E5%90%91%E4%B8%8B&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y4mWfzuyD1rj9WuAubrju90ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHTkPj0vrHmk)、逐步求精；

其程序结构是按功能划分为若干个基本模块，各模块之间的关系简单，在功能上相对独立；

每一模块内部均是由顺序、选择和循环三种基本结构组成；

其模块化实现的具体方法是使用子程序。

结构化程序设计由于采用了模块分解与功能抽象，[自顶向下](https://www.baidu.com/s?wd=%E8%87%AA%E9%A1%B6%E5%90%91%E4%B8%8B&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1Y4mWfzuyD1rj9WuAubrju90ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPHTkPj0vrHmk" \t "_blank)、分而治之的方法，从而有效地将一个较复杂的程序系统设计任务分解成许多易于控制和处理的子任务，便于开发和维护。

**1974，Donald E.Knuth (stanford)**

**计算机程序设计艺术:**

作者计划，整套书共有7卷，现出了4卷：

第1卷 基本算法：第1章 基本概念；第2章 信息结构；

第2卷 半数值算法：第3章[随机数](https://baike.baidu.com/item/%E9%9A%8F%E6%9C%BA%E6%95%B0" \t "_blank)；第4章 算术；

第3卷 排序与查找：第5章 排序；第6章 查找；

第4卷 组合算法：第7章 组合检索；第8章 递归；

未完成：

第5卷 语法算法：第9章 词法扫描；第10章 语法分析

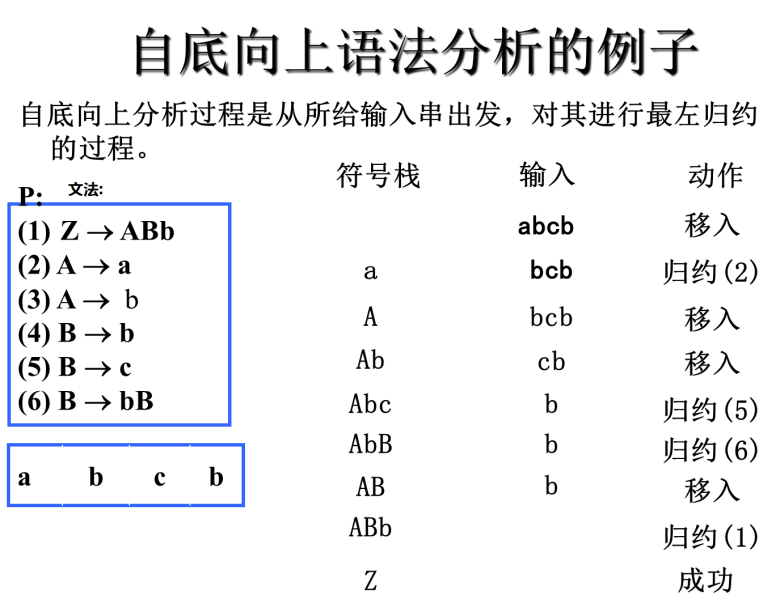
第6卷 上下文无关语言理论

第7卷 编译程序

**KMP算法：**

由D.E.Knuth，J.H.Morris和V.R.Pratt同时发现，解决两字符串的匹配问题，如搜索文档

**LR(K)语法分析方法：L自左向右，R最右推导**

****

LR(k)是从左到右分析，每次向前预测k步的一种编译方法。指从左至右扫描和自底向上的语法分析，且在分析的每一步，只须根据分析栈当前已移进和归约出的全部文法符号，并至多再向前查看K个输入符号

**1976，Michael O.Rabin (以色列)** & **Dana S.Scott (英) 师兄弟 (导师A.Church)**

判定问题：

非确定有穷自动机：

**Alan Turing**是Church的学生

**单向函数：**满足 对于所有属于 f 定义域的任一 x ，可以很容易计算 f( x ) = y； 对于几乎所有属于 f 值域的任一 y ，则在计算上不可能求出 x 使得 y = f( x )。

单向函数一般用于产生消息摘要，[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)加密等，常见的有：

**MD5**（Message Digest Algorithm 5）：是RSA数据安全公司开发的一种单向[散列算法](https://baike.baidu.com/item/%E6%95%A3%E5%88%97%E7%AE%97%E6%B3%95" \t "_blank)，MD5被广泛使用，可以用来把不同长度的数据块进行暗码运算成一个128位的数值；

**SHA**（Secure Hash Algorithm）这是一种较新的[散列算法](https://baike.baidu.com/item/%E6%95%A3%E5%88%97%E7%AE%97%E6%B3%95" \t "_blank)，可以对任意长度的[数据运算](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E8%BF%90%E7%AE%97)生成一个160位的数值；

**MAC**（Message Authentication Code）：消息认证代码，是一种使用[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)的单向函数，可以用它们在系统上或用户之间认证文件或消息。HMAC（用于消息认证的[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)散列法）就是这种函数的一个例子。

**CRC**（Cyclic Redundancy Check）：循环冗余校验码，[CRC校验](https://baike.baidu.com/item/CRC%E6%A0%A1%E9%AA%8C" \t "_blank)由于实现简单，检错能力强，被广泛使用在各种数据校验应用中。占用系统资源少，用软硬件均能实现，是进行数据传输差错检测地一种很好的手段（CRC 并不是严格意义上的[散列算法](https://baike.baidu.com/item/%E6%95%A3%E5%88%97%E7%AE%97%E6%B3%95" \t "_blank)，但它的作用与散列算法大致相同，所以归于此类）。

**S.Scott：语义学：**研究自然语言中词语意义的学科

**1980，C. Anthony R.Hoare (英)**

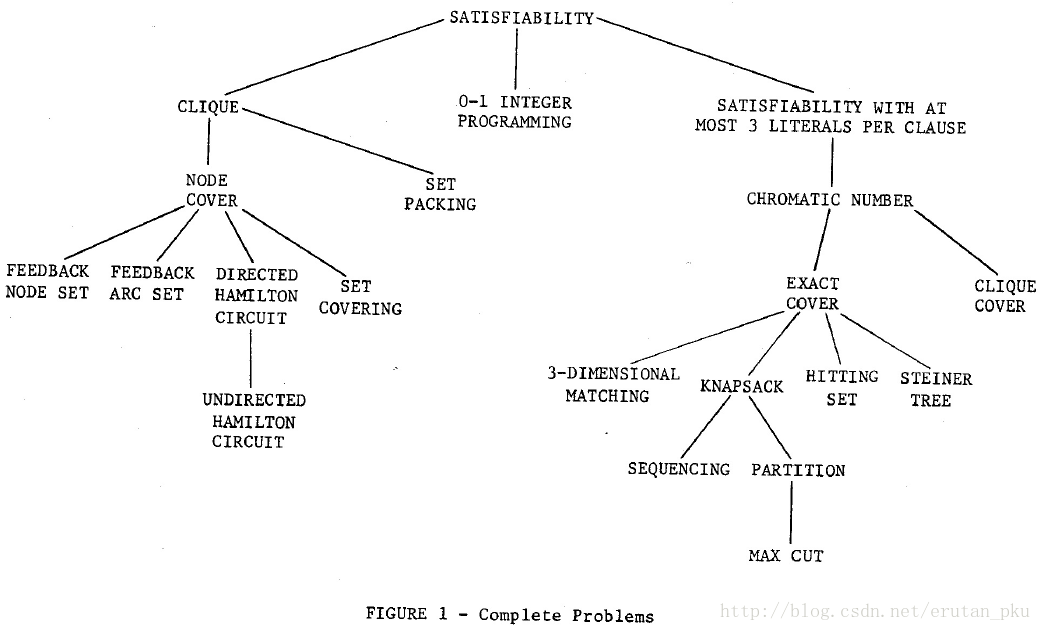
2000年因为其在计算机科学与教育上做出的贡献被封为爵士

**1982，Steven A.Cook (加Toronto大学)**

**Cook定理：**可满足性问题SAT是NP完全的

**1985，Richard M.Karp (UC-Berkeley)**

[Karp的21个NPC问题及其规约](https://www.baidu.com/link?url=A69H-MWQCj8GJEWv454unRwQC0gh2F7NlSkWkZvY9sg2CXwwoUrH4gWJNontdcCLrB1QCq5I_kOnGnhyqQ00qew1cpDQZZA-gozX3uRYmW3&wd=&eqid=a4d6b92300052c830000000359b51bf8)



**2002，Ronald L. Rivest，Adi Shamir，Leonard M. Adelman**

