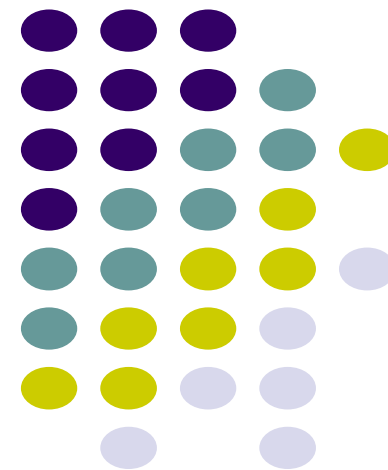


L1:数据结构与算法

谷方明

fmgu2002@sina.com





学习目标

- 了解数据结构的研究内容
- 理解数据结构的定义
- 了解算法的定义
- 掌握算法的**ADL**描述
- 了解算法的评价及正确性证明

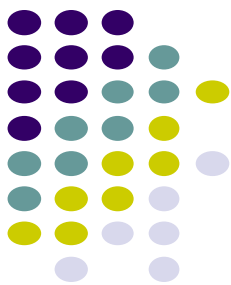


例1：查找问题（引入）

□ 实现功能：在学生信息表中**快速**查询**GPA**。

- ✓ 应用场景：教务系统查自己的GPA
- ✓ 为了直观假定输入姓名查询

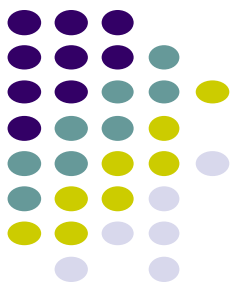
学号	姓名	学院	GPA
21000904	杨戬	计算机学院	3.9
55210906	天蓬	软件学院	3.3
21000912	孙悟空	计算机学院	4.0
55210914	嫦娥	软件学院	3.8
21001613	哪吒	计算机学院	3.0



□ 方案一：将信息表组织成顺序表或链表

序号	姓名	GPA
1	杨戬	3.9
2	天蓬	3.3
3	孙悟空	4.0
4	嫦娥	3.8
5	哪吒	3.0



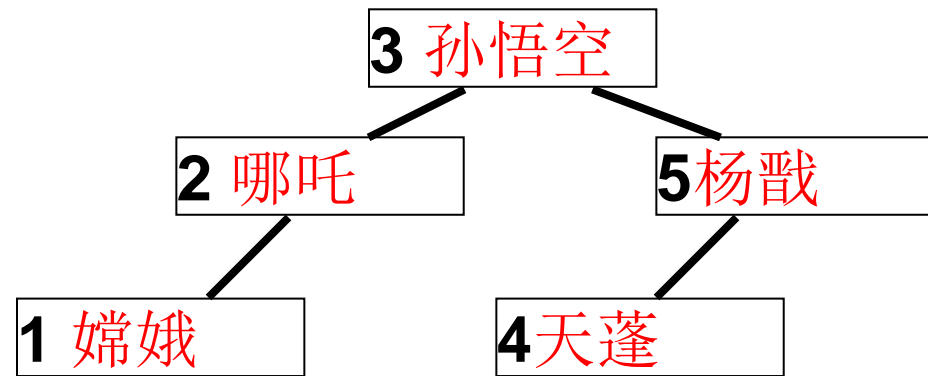


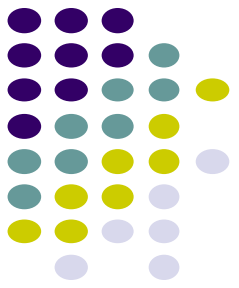
□ 方案二：将信息表组织成有序表（拼音序）

序号	姓名	GPA
1	嫦娥	3.8
2	哪吒	3.0
3	孙悟空	4.0
4	天蓬	3.3
5	杨戬	3.9



□ 方案三：将信息表组织成平衡树





□ 更多组织方案

- ✓ 哈希
- ✓ 字典树
- ✓ 索引（数据库）
- ✓



数据结构的研究内容

- 数据结构研究：**组织数据**（结构化信息）的方法，以支持**有效**的处理。
- 说明
 - ✓ **有效(efficient)**：满足资源限制。本课程关注**时间和空间限制**。
 - ✓ **提高有效性的方法很多**，除了数据结构，还有算法设计、程序优化等。



课程地位（课程说明）

□ 计算机科学的**核心课**

✓ 算法 + 数据结构 = 程序

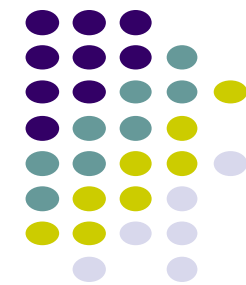
□ 重要的**专业基础课**

✓ 编译原理：栈、语法树

✓ 操作系统：队列、目录树、死锁（环）

✓

□ 计算机、软件等相关专业的**考研课**



卓班课程设置

□ 学时：理论课（**56**） + 实验课（**12**）

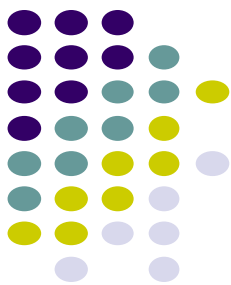
□ 考核方式：

- ✓ 平时（学习通：在线课+作业+课堂活动）；
- ✓ 实验；
- ✓ 期末考试；



教材和参考书

- ❑ 教材：《数据结构》第3版，刘大有等编著，高等教育出版社，2017年3月；
 - ✓ 勘误表（在线课程中的资源里）
- ❑ 参考书
 - ✓ 《算法导论》，CLRS，潘金贵，机械工业出版社
 - ✓ 《计算机编程艺术》，Donald E. Knuth
 - ✓ 《数据结构与算法分析》，Mark Allen Weiss
 - ✓ 《数据结构与算法分析》，Clifford A. Shaffer
 - ✓ STL网上教程（搜索引擎：STL 教程）
 - ✓ 《STL源码剖析》，侯捷，华中科技大学出版社



在线课程

□ 数据结构（学习通）

□ 参考在线课程



✓ MIT 6.006: (Introduction to Algorithms)

<https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/>

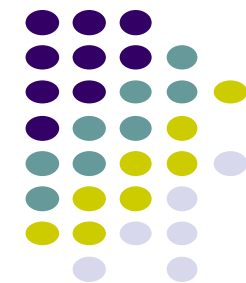
✓ Stanford cs166: <http://web.stanford.edu/class/cs166/>

✓ 清华大学 数据结构 上/下

https://www.xuetangx.com/course/THU08091000384/5883586?channel=search_result

✓ 浙江大学 数据结构

<https://www.icourse163.org/course/zju0901-93001>



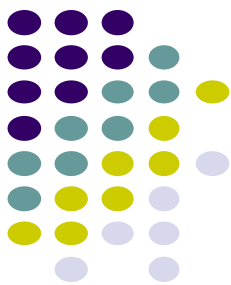
学习建议

□ 课程特点

- ✓ 理论：难（抽象层级）
- ✓ 实践：多（代码行数）

□ 学习方法

- ✓ 多编程；
- ✓ 多参考（参考书、在线课程、博客等）；



其它说明

□ 课程免修

- ✓ 条件：竞赛/认证；论文（数据结构算法相关C）
- ✓ 达到规定条件、任课教师同意、学院讨论

□ 出勤

- ✓ 签到；学院、安全责任；大牛可安静自学或助教

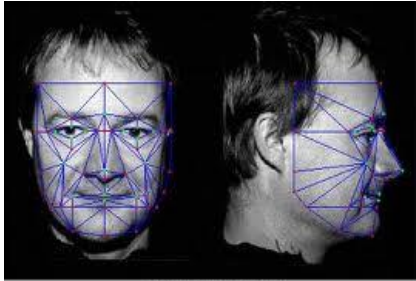


1. 数据

- 数据是对象的表示。

即按照适合于通信、解释或处理（借助人或自动装置）的方式所形成的关于事实、概念或指令的表示（百科）。

- 数据是计算机程序要处理的“原料”，是所有被计算机识别、存储和加工处理的符号的总称。



人脸特征点分布示意图



上证指数

2421.117

▲0.49% +11.69

实时行情

加入自选股

2010-07-07 15:03

查看其他证券“历史行情”

代码/拼音缩写/中文

确定

实时行情

成交明细

历史行情

历史行情

从 2010-05-21 至 2010-07-06

☒按日

☐按周

☐按月

查询

日期	开盘	收盘	涨跌幅	涨跌幅	最低	最高	成交量(手)	成交金额(万)	换手率
累计：	2010-05-21至2010-07-06		-174.10	-6.74%	2319.74	2666.54	20652674	217033504.00	-
2010-07-06	2358.26	2409.42	45.48	1.92%	2356.55	2409.78	613643	5561825.50	-
2010-07-05	2358.76	2363.95	-18.95	-0.80%	2335.57	2378.09	479965	4299967.00	-
2010-07-02	2371.32	2382.90	9.11	0.38%	2319.74	2386.40	683296	6154527.00	-
2010-07-01	2393.95	2373.79	-24.58	-1.02%	2371.78	2410.77	499677	4641516.00	-
2010-06-30	2409.54	2398.37	-28.66	-1.16%	2382.36	2411.44	552106	5115435.00	-
2010-06-29	2534.50	2427.05	-108.22	-4.27%	2425.93	2541.14	737449	7136187.00	-
2010-06-28									
2010-06-25									
2010-06-24									
2010-06-23									
2010-06-22									
2010-06-21									
2010-06-18									
2010-06-17									
2010-06-16									

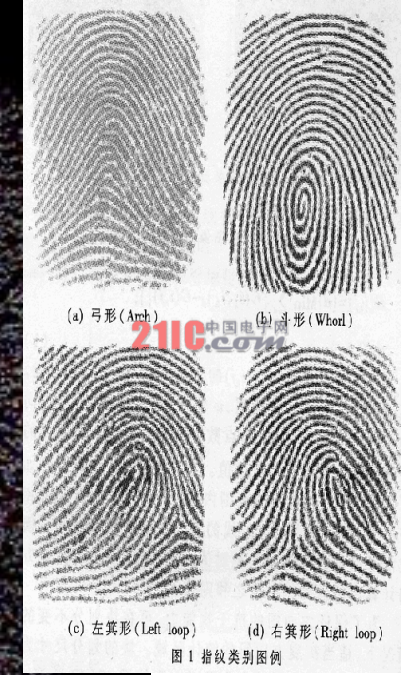
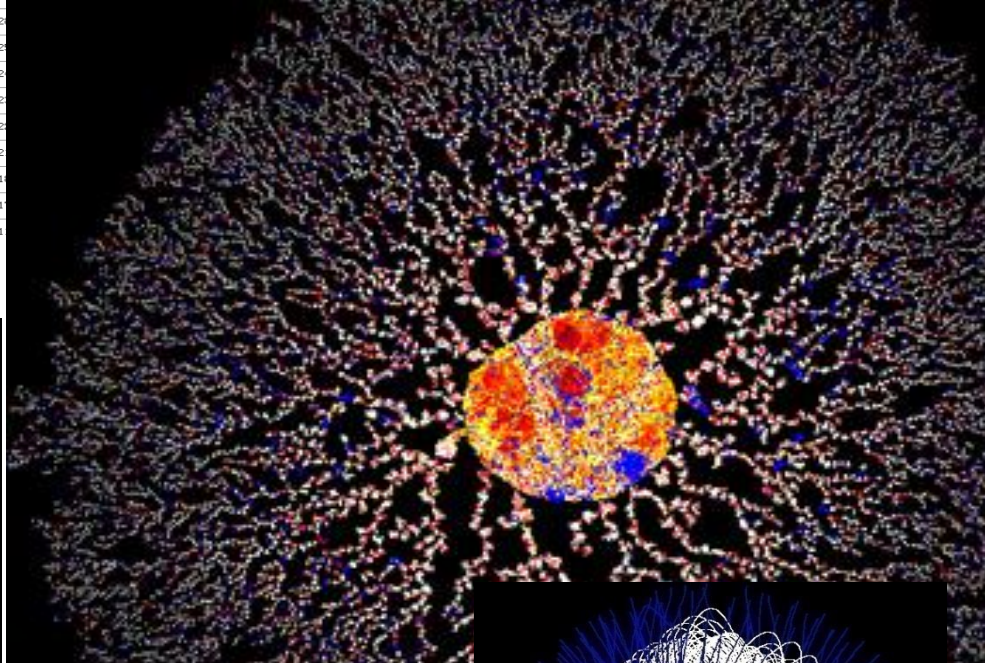
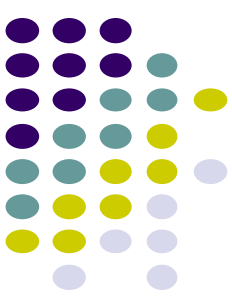
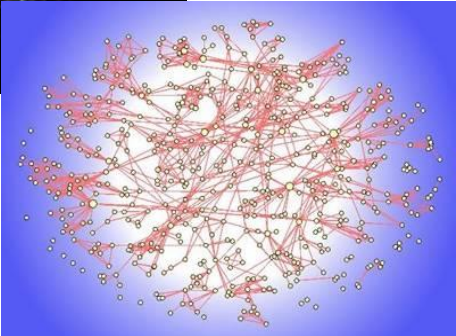
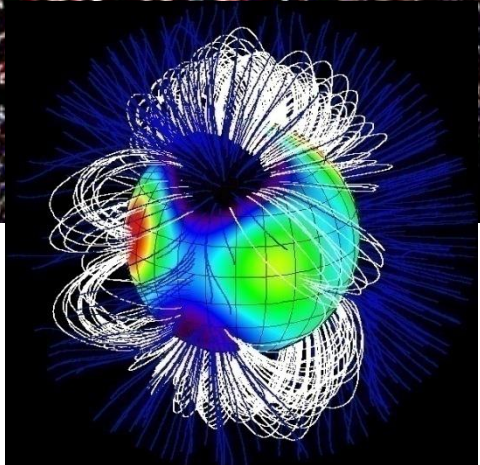
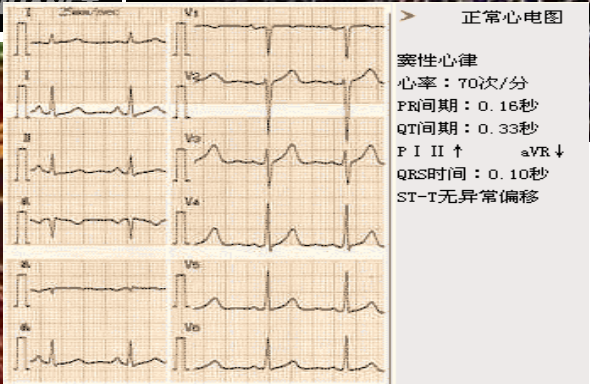


图1 指纹类别图例





数据元素和数据项

- 数据元素是组成数据的基本单位，也称元素、结点、数据成分等。
 - ✓ 数据可理解成数据元素的集合
 - ✓ 数据元素可大可小，在程序中通常把一个数据元素作为一个整体来考虑和处理
- 数据元素可由若干数据项组成。数据项也称为域、字段等；

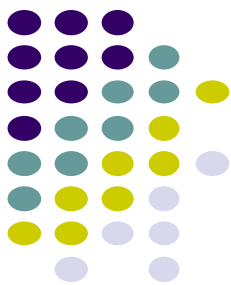


例2: 数据元素和数据项

学号	姓名	学院	GPA
21000904	杨戬	计算机学院	3.9
55210906	天蓬	软件学院	3.3
21000912	孙悟空	计算机学院	4.0
55210914	嫦娥	软件学院	3.8
21001613	哪吒	计算机学院	3.0

数据
元素

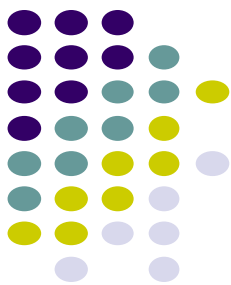
数据
项



2. 数据结构

□ 数据结构由三部分组成

- ✓ 数据的逻辑结构
- ✓ 数据的存储结构
- ✓ 运算（操作）



2.1 数据的逻辑结构

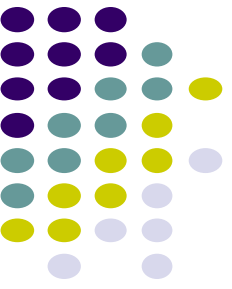
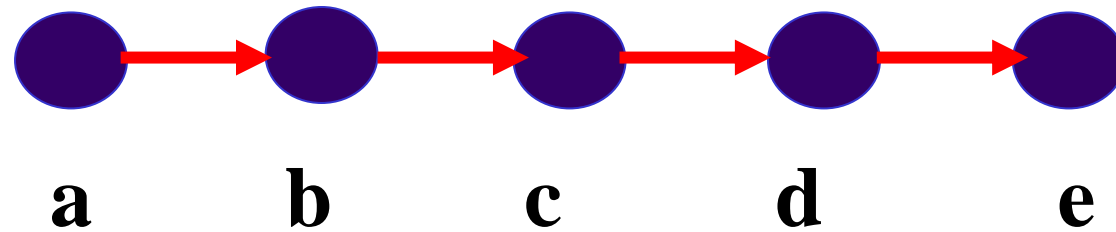
- 数据的逻辑结构是指数据元素及其间的逻辑关系。
- 可形式定义为一个二元组： $L = (N, R)$
 - ✓ N 是有限结点集合，
 - ✓ R 是 N 上的二元关系 r 的集合
- 图形化表示
 - ✓ 将数据元素抽象为结点
 - ✓ 数据元素间的关系抽象为连接结点的边。

例3 $L=(N,R)$,

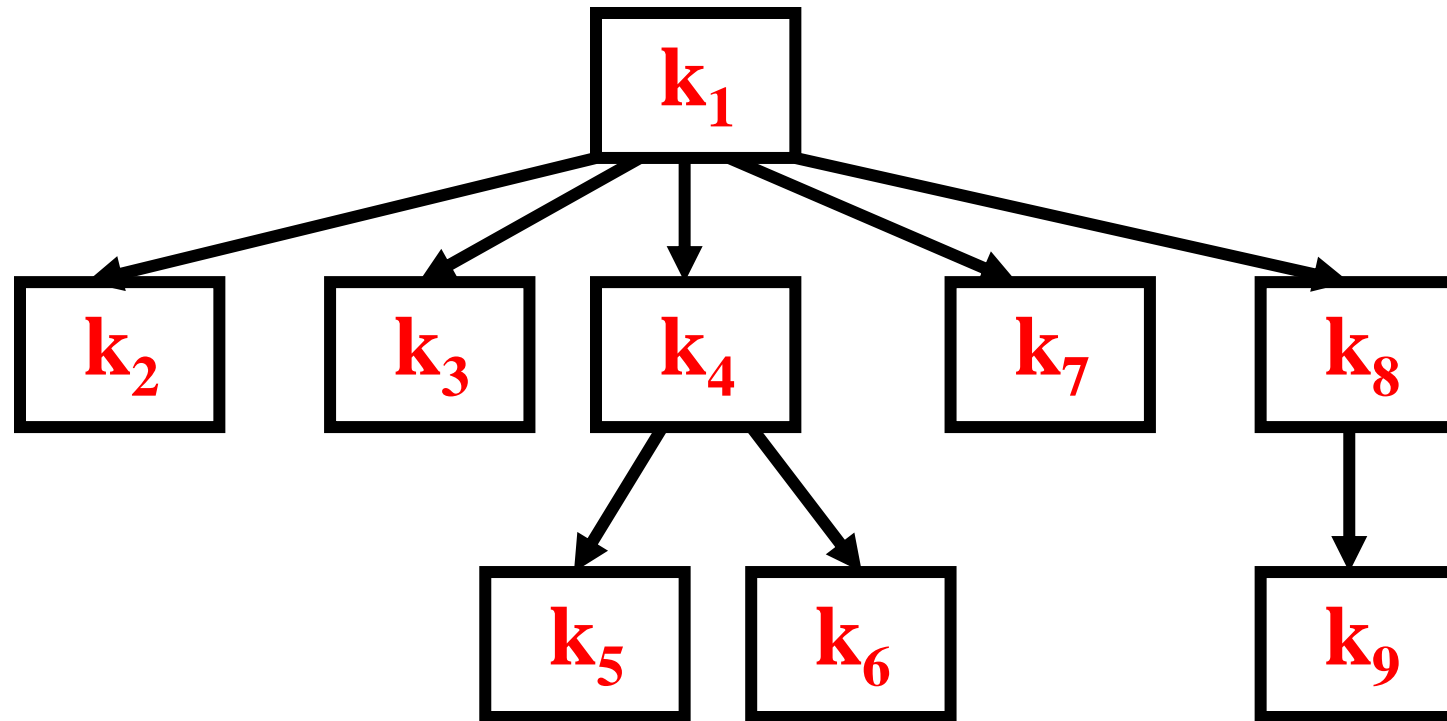
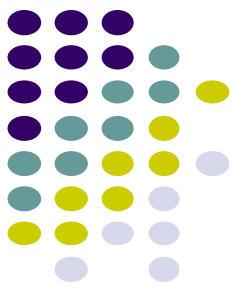
$N=\{a, b, c, d, e\}$,

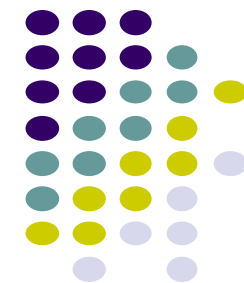
$R=\{r\}$,

$r=\{<a,b>, <b,c>, <c,d>, <d,e>\}$



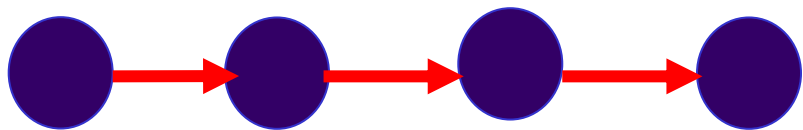
例4 $L=(N,R)$, $N=\{k_1, k_2, \dots, k_9\}$, $R=\{r\}$,
 $r=\{<k_1, k_2>, <k_1, k_3>, <k_1, k_4>, <k_1, k_7>,<k_1, k_8>, <k_4, k_5>, <k_4, k_6>, <k_8, k_9>\}$





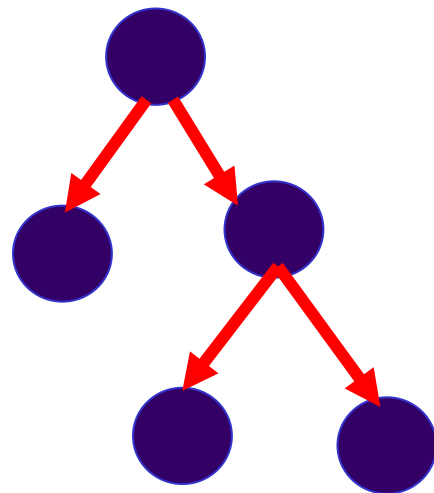
逻辑结构

线性结构



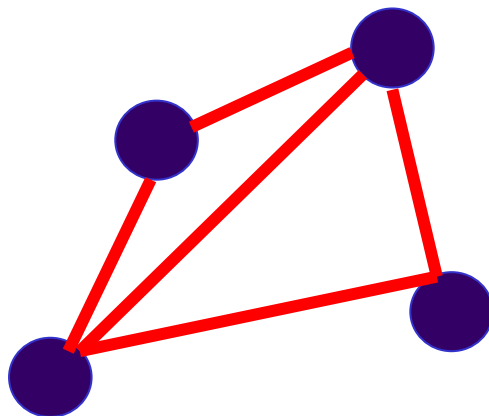
一对一

树



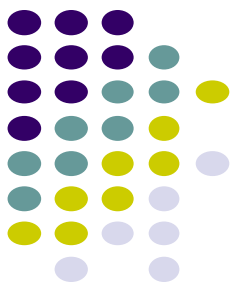
一对多

图

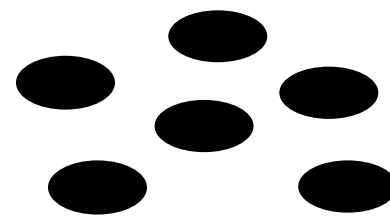


多对多

分类准则：图形表示中的前驱后继个数



逻辑结构说明



集合

□ 这种分类不是绝对的。

- ✓ 有些书将数据的逻辑结构分为四种：集合、线性结构、树和图（集合：数据元素之间除了同属于一个集合的关系之外，别无其他关系）。
- ✓ 未涉及复杂关系（如广义表）。

□ 数据的逻辑结构有时被直接称为数据结构。要根据语境区分。



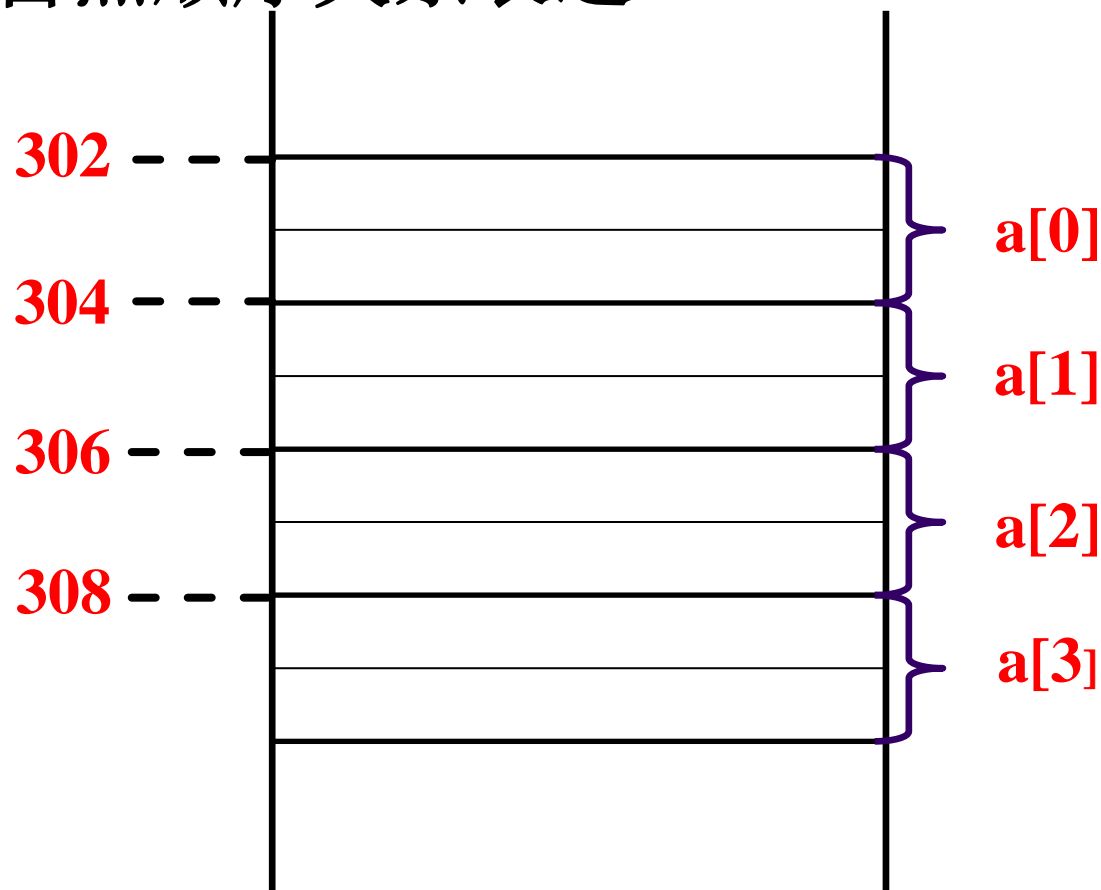
2.2 数据的存储结构（物理结构）

- 数据的存储结构是指：数据的逻辑结构在计算机中所需的**存储空间、空间的构成结构**及对该存储结构的**访问方式**等的总称。
- **数据的存储结构是建立一种由逻辑结构到存储空间的映射。**
 - ✓ 例：学生信息表



1) 顺序存储结构

- 把一组结点存放在地址相邻的存储单元里, 结点间的逻辑关系用存储单元的自然顺序关系表达



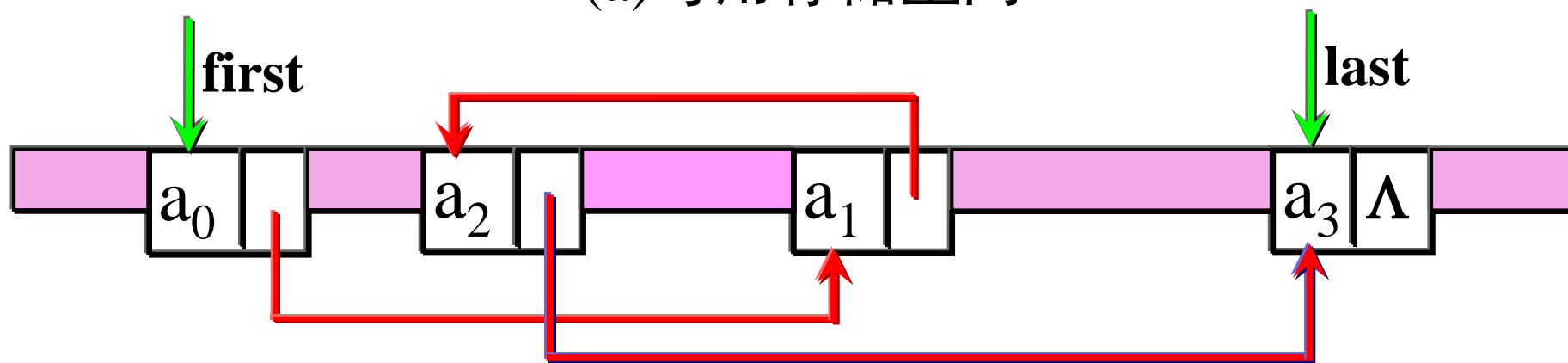


2) 链接存储结构

- 在结点的存储结构中附加指针字段，两个结点的逻辑后继关系用指针的指向来表达。



(a)可用存储空间

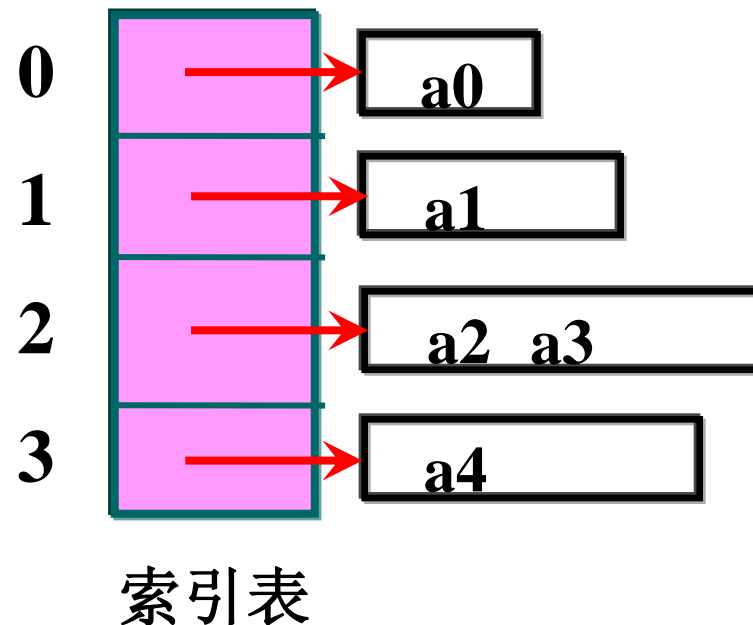


(b)创建若干个结点后



3) 索引存储结构

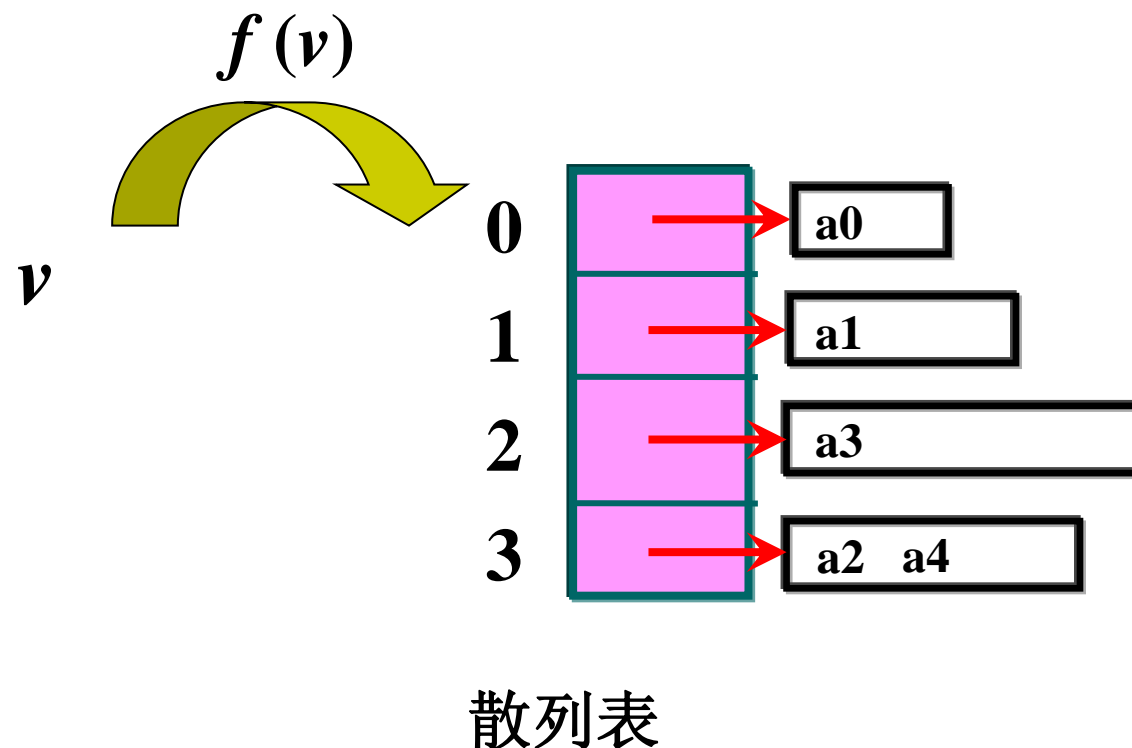
- 索引表：存储一串指针，每个指针指向存储区域的一个数据结点。索引表一般具有序关系。
- 查找时，先查索引表，再找数据结点。

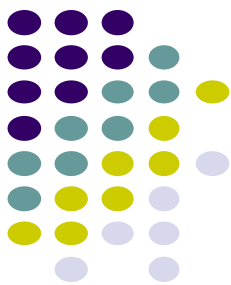




4) 散列存储结构

- 利用散列函数直接计算散列地址，然后再查找结点的地址。可看作索引存储的一种延伸。





小结

□ 四种存储结构

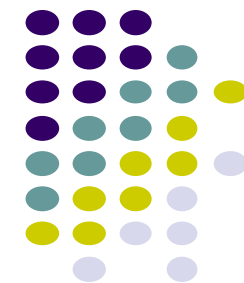
- ✓ 顺序
- ✓ 链接
- ✓ 索引
- ✓ 散列

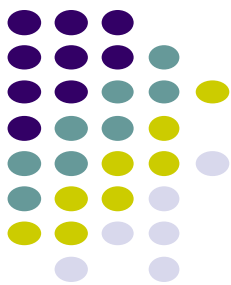
□ 存储结构要正确反映逻辑结构，并便于操作。

□ 存储结构可单独使用，也可组合使用。

2.3 对数据的操作（运算）

- 查找
- 排序
- 插入
- 删除
- 修改
-





例5 学生信息表的数据结构设计

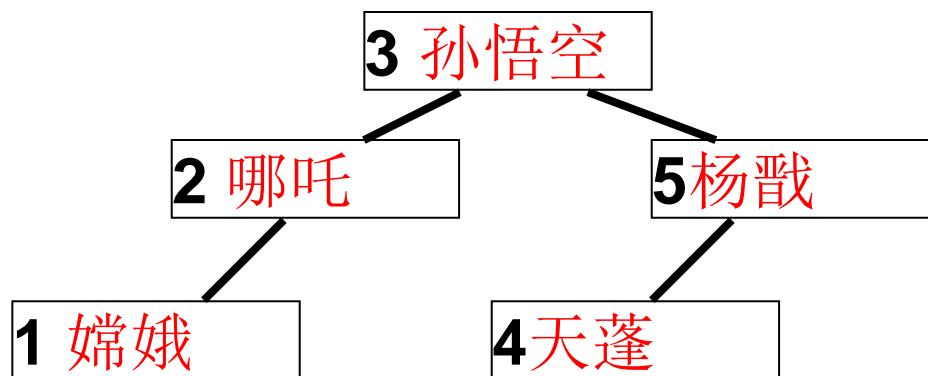
□ 设计方案1

- ✓ 线性结构
- ✓ 顺序存储
- ✓ 查找

序号	姓名	GPA
1	杨戬	3.9
2	天蓬	3.3
3	孙悟空	4.0
4	嫦娥	3.8
5	哪吒	3.0

□ 设计方案2

- ✓ 树
- ✓ 链接存储
- ✓ 查找





2.4 数据结构的定义

- 数据结构包括数据的逻辑结构、数据的存储结构和对数据的操作三方面内容
- 三者紧密相关，构成有机整体
 - ✓ 逻辑结构的不同会产生不同的数据结构。
 - ✓ 存储结构不同，即使逻辑结构相同，也会产生不同的数据结构。线性表是一种逻辑结构，顺序存储则为顺序表；链接存储则为链表；散列存储则为散列表。
 - ✓ 对数据的操作不同，即使其逻辑结构和存储结构相同，也可能对应着不同的数据结构。栈和队列。



相近概念

□ 数据类型

- ✓ 性质相同的**数据元素的集合**及其上定义的一组**操作**
- ✓ 例：int
- ✓ 实现了的数据结构；实现细节对用户透明(不可见)

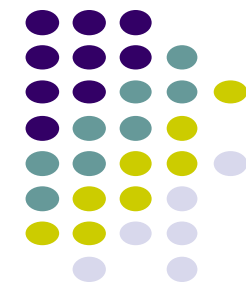
□ 抽象数据类型（ADT）

- ✓ **一组操作的集合；**
- ✓ 不涉及如何实现，用于设计决策阶段
- ✓ 例：学生信息表



2.5 数据结构简史

- 20世纪40年代
 - ✓ 电子计算机、数值计算，不存在数据结构
- 20世纪50年代
 - ✓ 数据类型增多（数组、记录、串和层次表）
- 20世纪60年代
 - ✓ 信息结构，**Knuth** 《计算机编程艺术》
- 20世纪70年代
 - ✓ 数据库（文件组织、存储 等）
 - ✓ **N. Wirth** 《算法+数据结构=程序》
- 20世纪80年代：抽象数据类型（面向对象）
- 20世纪90年代：网络（图模型）
- **21世纪：？ 数据科学**



3 算法(Algorithm)

□ 算法的著名例子：辗转相除法。

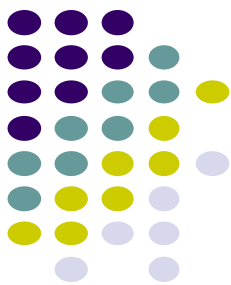
- ✓ 用于求两个正整数最大公约数。
- ✓ Euclid's Algorithm (Euclid's Elements , 第VII卷, 命题i和ii) 。

例：求 15 和 9 的最大公约数 $\gcd(15,9)$



算法的定义

- 算法是有穷规则的集合，规定了解决某一特定类型问题的运算序列（教材）。
- 通常，一个算法有**5**个重要特性（**Knuth**）：
 - 1.有限性：** 算法必须在执行有限步后结束；
 - 2.确定性：** 算法描述必须无歧义，通常结果确定；
 - 3.可行性：** 算法中的运算必须是可行的；
 - 4.输入：** 一个算法有零个或多个输入；
 - 5.输出：** 一个算法有一个或多个输出；



算法的描述

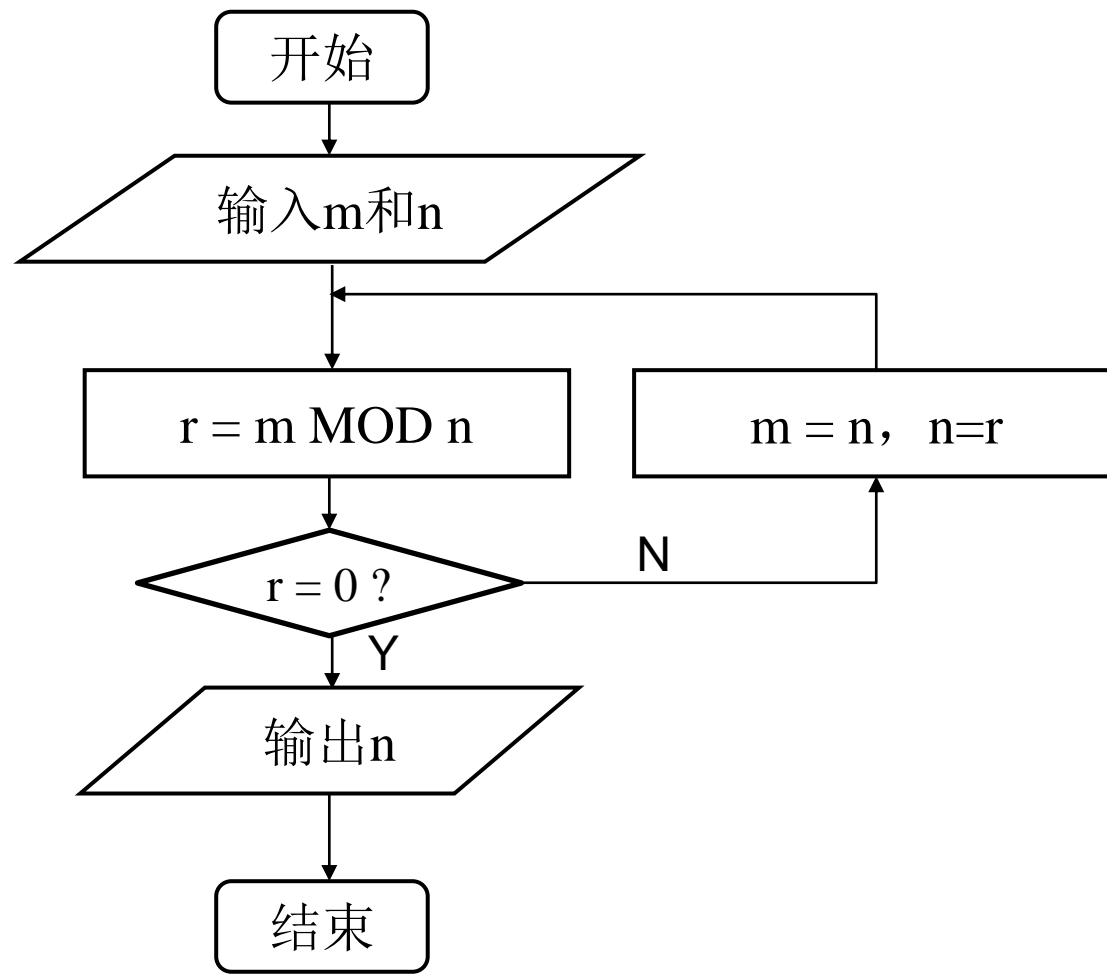
- 自然语言
- 流程图（盒图、**PAD**图）
- 程序（**C\C++、Java**等）
- 伪代码（结合程序设计语言自定义）
-



欧几里得算法（自然语言描述）

1. 输入两个正整数 m 和 n .
2. 循环执行如下两步:
 - 2.1. 计算余数 $r = m \text{ MOD } n$;
 - 2.2. 若 $r = 0$ ，则输出 n 、算法结束；
否则 m 变为 n 、 n 变为 r .

欧几里得算法（流程图描述）



欧几里得算法（C语言程序）



```
int main() {  
    int m,n,r;  
    scanf("%d %d",&m,&n);  
    while(true){  
        r = m % n;  
        if( r == 0){  
            printf("%d\n",n); break;  
        }else m=n, n=r;  
    }  
  
    return 0;  
}
```



欧几里得算法（ADL描述-类Pascal）

算法 E ($m, n . n$)

/* 输入两个正整数 m 和 n ，算法E求其最大公因子，输出结果在 n 中 */

E1. [求余数] $r \leftarrow m \text{ MOD } n$.

E2. [循环处理]

WHILE $r \neq 0$ **DO** (

$m \leftarrow n . n \leftarrow r$.

$r \leftarrow m \text{ MOD } n$.

)**■**



欧几里得算法（ADL描述-类C++）

算法 $E(m, n . n)$

/* 输入两个正整数 m 和 n ，算法E求其最大公因子，输出结果在 n 中 */

E1. [求余数] $r = m \% n$;

E2. [循环处理]

while (r) {

$m = n, n = r$;

$r = m \% n$;

} ■



ADL描述规范

算法<算法名>(变量 i_1 , ..., 变量 i_m . 变量 j_1 , ..., 变量 j_n)

//<算法的概括说明> 或者 /*<算法的概括说明>*/

<步骤名><步骤号>. [<本步骤的概括说明>]

<操作1>

...

<操作J>

<步骤名><步骤号>. [<本步骤的概括说明>]

...





ADL规范说明

- **<>**: 明确规范中的可替换部分，描述算法时去掉
- **算法名**: 由字母和数字组成的有限字符串，且第一个符号必须是字母；通常要短一些
- **算法的概括说明**：算法的功能、参数说明等，通常必须有
- **操作**
 - ✓ **表达式**: 可使用数学表示
 - ✓ **语句**: 类PASCAL；语句结束符号用“.”；可换成C++
- 用符号“**■**”作为算法描述完毕的结束符。



算法描述小结

- **自然语言描述**便于理解，适合**顶层描述**；根据需要可引入数学运算等；
- 图形描述法要画图，适合理解过程或特殊用途；
- **伪代码描述**能专注计算和逻辑（忽略数据类型、直接使用数学运算和符号），适合**函数级算法**。
- 代码描述能直接运行，但关注存储细节，导致描述较长、不利于设计层面的思考；
 - ✓ 教材用PASCAL风格的算法描述语言ADL；代码使用的是C++语言（扫描二维码）。



算法描述建议

□ 函数级算法建议使用伪代码描述

- ✓ 能读懂教材上的**ADL**描述；
- ✓ 描述算法时，可用**C/C++**风格的指令替换**PASCAL**指令，**ADL**只是一种规范/模板。

□ 问题复杂或抽象程度较高时，建议使用自然语言。

□ 程序设计语言描述不建议。

- ✓ 算法复杂时，程序设计语言描述很长；不能突出算法核心和算法设计



例：伪代码结合自然语言

INSERTION-SORT(A)

1 **for** $j \leftarrow 2$ **to** $\text{length}[A]$

2 **do** $\text{key} \leftarrow A[j]$

3 \triangleright Insert $A[j]$ into the sorted sequence $A[1..j-1]$.

4 $i \leftarrow j-1$

5 **while** $i > 0$ and $A[i] > \text{key}$

6 **do** $A[i+1] \leftarrow A[i]$

7 $i \leftarrow i-1$

8 $A[i+1] \leftarrow \text{key}$



例：自然语言结合伪代码

DAG-SHORTEST-PATHS(G, w, s)

1 topologically sort the vertices of G

2 INITIALIZE-SINGLE-SOURCE(G, s)

3 **for** each vertex u , taken in topologically sorted order

4 **do for** each vertex $v \in Adj[u]$

5 **do** RELAX(u, v, w)



算法定义的说明

□ 算法和程序

- ✓ 算法可看作是程序的思想；程序可看作是算法的实例（不是所有的程序都是算法的实例）
- ✓ 有时不加区分的使用两者

□ 算法的一个经典定义：算法就是**定义良好的计算过程**，它取一个或一组值作为输入，并产生一个或一组值作为输出。——《算法导论》

- ✓ 定义良好指满足前述算法的5个特性。



算法与数据结构的关系

□ 算法与数据结构相互依存

- ✓ 算法依附于具体的数据结构，数据结构直接关系到算法的选择和效率。
- ✓ 对数据结构的操作由计算机来完成，这就要设计相应的插入、删除和修改等算法。

□ 算法+数据结构=程序



算法的评价准则

□ 一个好的算法通常考虑以下5个方面

- ✓ 正确性
- ✓ 时间效率：运行时间
- ✓ 空间效率：运行空间（内存等）
- ✓ 可读性
- ✓ 鲁棒性（ROBUSTNESS）

□ 算法设计的基本要求（数据结构课程）是正确性和时空效率



4 算法的正确性证明

- 一个算法是正确的，是指对于一切合法的输入数据，该算法都会在有限时间产生正确的结果。
- 对于一个具体算法，证明步骤如下
 - ✓ 算法有限步终止；
 - ✓ 算法终止时，得到的结果是正确的；
- 算法的正确性一般通过数学方法证明，如直接证明法、数学归纳法、循环不变式、反证法等

数学归纳法



- 数学归纳法，通常被用于证明某个给定命题（定理）在整个（或者局部）自然数范围内成立。
- 数学归纳法依靠假设的事实来证明定理，是用“有限”步骤解决“无限”步骤问题的一种严格证明方法。
- 例 证明由如下递归关系式

$$T(n) = 0 \quad \text{当 } n = 1$$

$$T(n) = T(n-1) + 1 \quad \text{当 } n > 1$$

可推出 $T(n) = n - 1$ ，其中 $n \geq 1$ 。



证明（数学归纳法）

1. 基础情形： $n = 1$ 时， $T(1) = 1 - 1 = 0$ ，结论成立。

2. 归纳步骤：

假设 $T(n - 1) = n - 2$ 成立，往证 $T(n) = n - 1$ 成立。

由递归定义知：

$$n > 1 \text{ 时，有 } T(n) = T(n - 1) + 1$$

再由归纳假设：

$$T(n) = T(n - 1) + 1 = n - 2 + 1 = n - 1$$

由数学归纳法推出 $T(n) = n - 1$ 成立。 ■



第一数学归纳法

设 T 是一个定理， n 是 T 中的正整数参数。数学归纳法表明，如果下面两个条件为真，则对于正整数参数 n 的任何值， T 都是正确的：

①基础情形： $n = c$ 时， T 成立。

②归纳步骤： 若 $n = k-1$ 时 T 成立, 则 $n=k$ 时 T 也成立。

其中， c 是一个较小的正整数常量， $n \geq c$.



第二数学归纳法 (强归纳法)

- ①基础情形: $n = c$ 时, T 成立。
- ②归纳步骤: 若对于所有的 $n < k$ 时 T 都成立, 则 $n = k$ 时 T 也成立。

第一数学归纳法的归纳步骤:

若 $n = k-1$ 时 T 成立, 则 $n=k$ 时 T 也成立。

- 强归纳法的假设条件更强;
- 两者等价; 根据问题的情况选用;

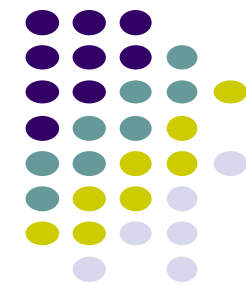


证明算法E的正确性

1. 算法E有限步终止

r 最大是 $\min(m,n)$ ，每次辗转相除， r 至少减1

2. 终止时得到正确结果。即证明：经过辗转相除，当 $r = 0$ 时，得到的变量 n 的值 n' 是 m 和 n 的最大公因子。



引理1.1

- 若 m, n 为正整数, $r = m \bmod n$, 则 m 和 n 最大公因子 等于 n 和 r 的最大公因子。



引理1.1证明

- 设 r 为 m 除以 n 所得的余数，则 m 可以表示为： $m = kn + r$ 。其中， k 是正整数。
- 设 m 和 n 的公因子集为 F ， n 和 r 的公因子集为 G 。
 - ✓ 任取 $c \in F$ ，则 c 整除 m ， c 整除 n 。而 $r = m - kn$ ，因此， c 整除 r ，即 $c \in G$ ， $F \subseteq G$ 。
 - ✓ 任取 $d \in G$ ，则 d 整除 n ， d 整除 r 。而 $m = kn + r$ ，因此， d 整除 m ，即 $d \in F$ ， $G \subseteq F$ 。
- 因此， $F = G$ 。即 m 和 n 的公因子等于 n 和 r 的公因子。特别地， m 和 n 的最大公因子等于 n 和 r 的最大公因子。



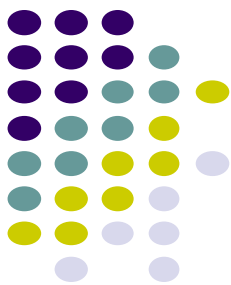
算法E的正确性证明：对 n 归纳

- 1.基础情形： $n=1$ 时， m 和 n 的最大公因数为1，此时， $n' = 1$. 结论成立。
- 2.归纳步骤：假设 $n < k$ 时， n' 是 m 和 n 的最大公因子，结论成立。往证 $n = k$ 时， n' 是 m 和 n 的最大公因子；
 - ✓ 设第一次运行算法时， $m = dn + r$ ， $m \leftarrow n$ ， $n \leftarrow r$ 。由于 $r < k$ ，由假设知： n' 即为 n 和 r 的最大公因子。
 - ✓ 由引理1.1， m 和 n 的最大公因子等于 n 和 r 的最大公因子。因此， n' 是 m 和 n 的最大公因子。证毕。



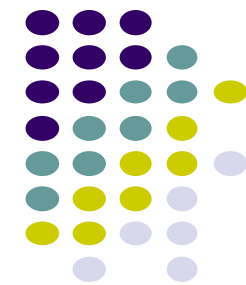
通用算法的正确性证明

- 通用算法的停机问题是不可计算问题
 - ✓ 例：冰雹猜想/角谷猜想/ $3x+1$
- 通用算法的正确性证明是不可计算问题



总结

- 数据结构的研究内容
- 数据结构的定义
- 算法的定义（**5**个特性）
- 算法的描述（**ADL**规范和自然语言，能读会写）
- 算法的评价及正确性证明（两步、证明方法）



课后任务

□ 慕课

- ✓ 在线学习/预习 第 2 章 视频

□ 作业

- ✓ P48: 2-1, 2-2, 2-5,
2-6, 2-9, 2-10, 2-11
- ✓ 在线提交