



中國人民大學
RENMIN UNIVERSITY OF CHINA

第4讲 控制语句

余力

buaayuli@ruc.edu.cn

内容提要

- 4.1 逻辑思维与计算机解题
- 4.2 C语言的分支控制语句
- 4.3 C语言的循环控制语句



中國人民大學
RENMIN UNIVERSITY OF CHINA



01. 逻辑计算思维解题

先从一个例子说起.....

- 人大附中有四位同学中的一位做了好事，不留名，表扬信来了之后，校长问这四位是谁做的好事。

A说：不是我。

B说：是C。

C说：是D。

D说：他胡说。

已知三个人说的是真话，一个人说的是假话。现在要根据这些信息，找出做了好事的人。

- 题目如何求解？
- 题目用计算机如何求解？

问题1：如何对同学说的话建模

■ 回顾：关系运算符和关系表达式

运算符	说明	
>	大于	} 优先级相同（高）
>=	大于等于	
<	小于	
<=	小于等于	
=	等于	} 优先级相同（低）
!=	不等于	

- 关系表达式返回的值是一个逻辑值，即“真”（1）或“假”（0）

利用关系表达式建模四个人所说的话 (1)

- 结合任务，可以将四个人说的四句话写成关系表达式
 - 在声明变量时，我们让 **thisman** 表示要寻找的做了好事的人，定义它是字符变量

char thisman=""; // 定义字符变量并初始化为空

- 接着让 “==” 的含义为 “是”
- 让 “!=” 的含义为 “不是”

利用关系表达式建模四个人所说的话 (2)

说话人	说的话	写成关系表达式
A	“不是我”	thisman!= 'A'
B	“是C”	thisman== 'C'
C	“是D”	thisman== 'D'
D	“他胡说”	thisman!= 'D'

问题2：如何确定可能的答案

■ 枚举法

- 结合任务分析，A、B、C、D四个人，只有一位是做好事者。令做好事者为1，未做好事者为0，可以有如下4种状态

状态	A	B	C	D
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

四种状态的形式化表示

状态	赋值表达式
1	thisman='A'
2	thisman='B'
3	thisman='C'
4	thisman='D'

- 显然第一种状态是假定A是做好事者，第二种状态是假定B是做好事者， ...
- 所谓枚举是按照者四种假定逐一地去测试四个人的话有几句是真话，如果不满足三句为真，就否定掉这一假定，换下一个状态再试。
- 具体做法如下：

(1) 假定让thisman= 'A' 代入四句话中

说话人	说的话	关系表达式	值
A	thisman!='A';	'A'!='A'	0
B	thisman=='C';	'A'=='C'	0
C	thisman=='D';	'A'=='D'	0
D	thisman!='D';	'A'!='D'	1

四个关系表达式的值的和为1，不满足3句话为真，假设不成立，因此显然不是 'A' 做的好事。

(2) 假定让thisman= 'B' 代入四句话中

说话人	说的话	关系表达式	值
A	thisman!='A';	'B'!='A'	1
B	thisman=='C';	'B'=='C'	0
C	thisman=='D';	'B'=='D'	0
D	thisman!='D';	'B'!='D'	1

四个关系表达式的值的和为2，显然不是 'B' 做的好事。

(3) 假定让thisman= 'C' 代入四句话中

说话人	说的话	关系表达式	值
A	thisman!='A';	'C'!='A'	1
B	thisman=='C';	'C'=='C'	1
C	thisman=='D';	'C'=='D'	0
D	thisman!='D';	'C'!='D'	1

四个关系表达式的值的和为3，就是 'C' 做的好事。

枚举

- 按照上面的思路，一个人一个人去试，就是枚举。
- 从编写程序看，实现枚举最好使用

循环程序结构

- 对于每个人，实现逻辑判断最好使用

分支程序结构

学习目标

- 将实际问题抽象为逻辑关系
- 使用枚举法解题
- 通过实例串联起之前内容
 - 关系表达式与逻辑表达式
 - 程序的循环结构
 - 程序的分支结构

回顾示例

- 人大附中有四位同学中的一位做了好事，不留名，表扬信来了之后，校长问这四位是谁做的好事。

A说：不是我。

B说：是C。

C说：是D。

D说：他胡说。

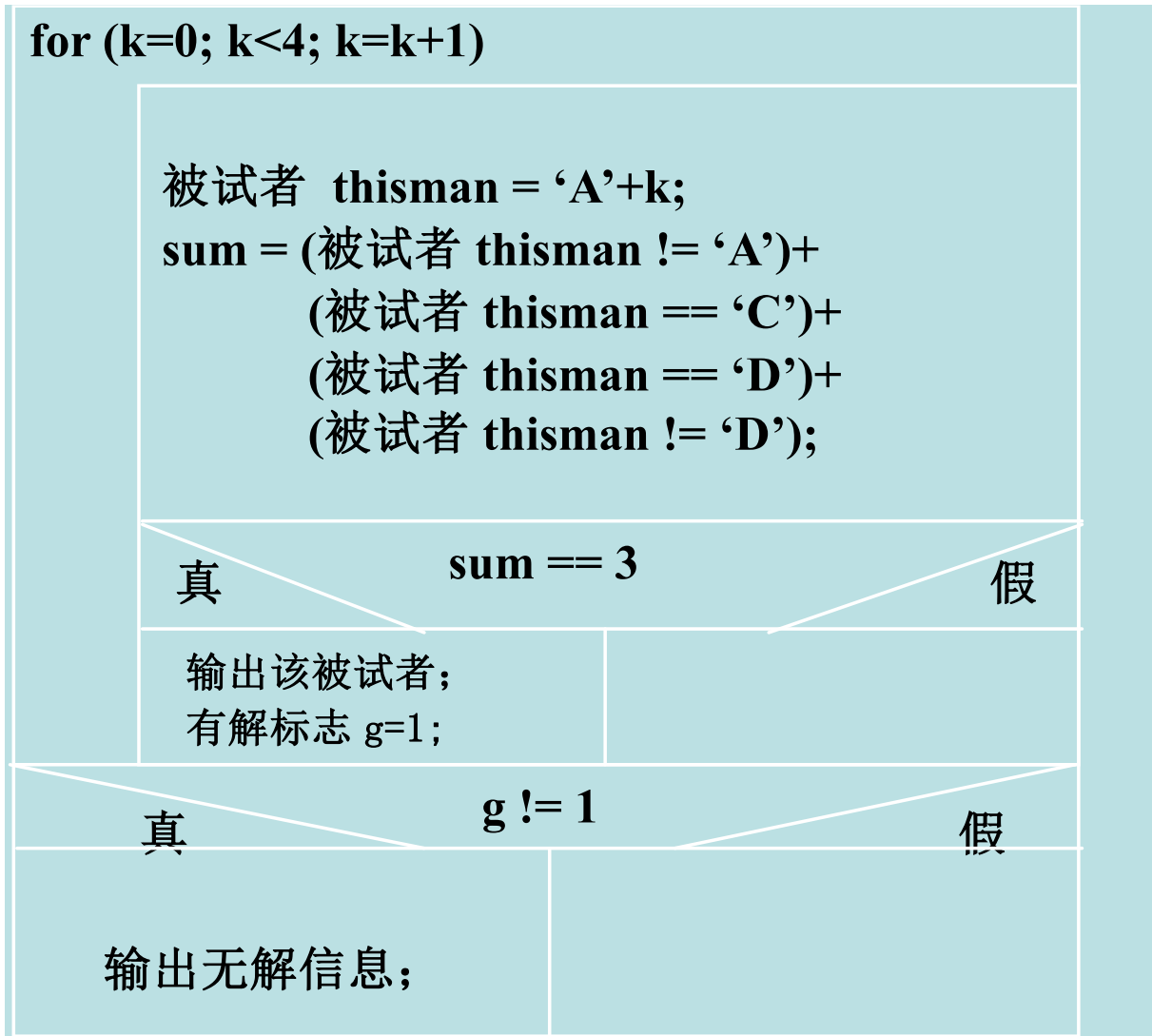
已知三个人说的是真话，一个人说的是假话。现在要根据这些信息，找出做了好事的人。

- 题目如何求解？
- 题目用计算机如何求解？

逻辑思维与计算机解题

- 一些逻辑问题必须转换成计算机能够看得懂的数学表达式和一定的程序指令
- 解题思路
 - 枚举法
- 数学表达式
 - 关系表达式与逻辑表达式
- 程序指令
 - 程序的循环结构
 - 程序的分支结构

程序NS图

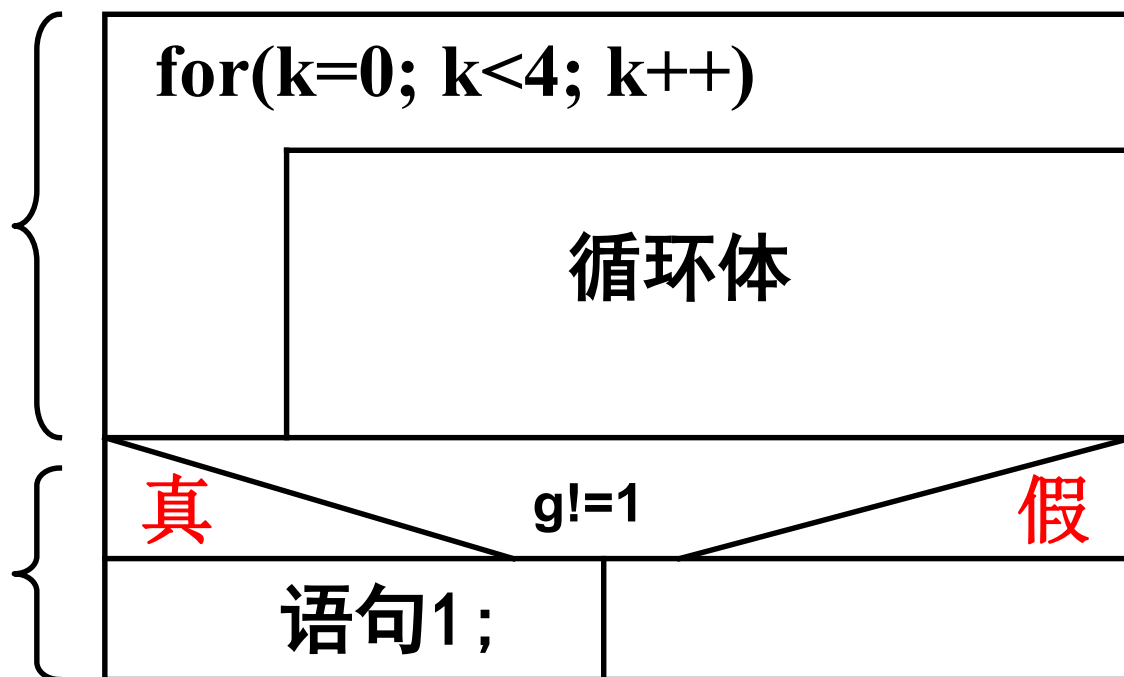


A说: 不是我。
B说: 是C。
C说: 是D。
D说: 他胡说。

程序NS图核心

第一块
循环结构

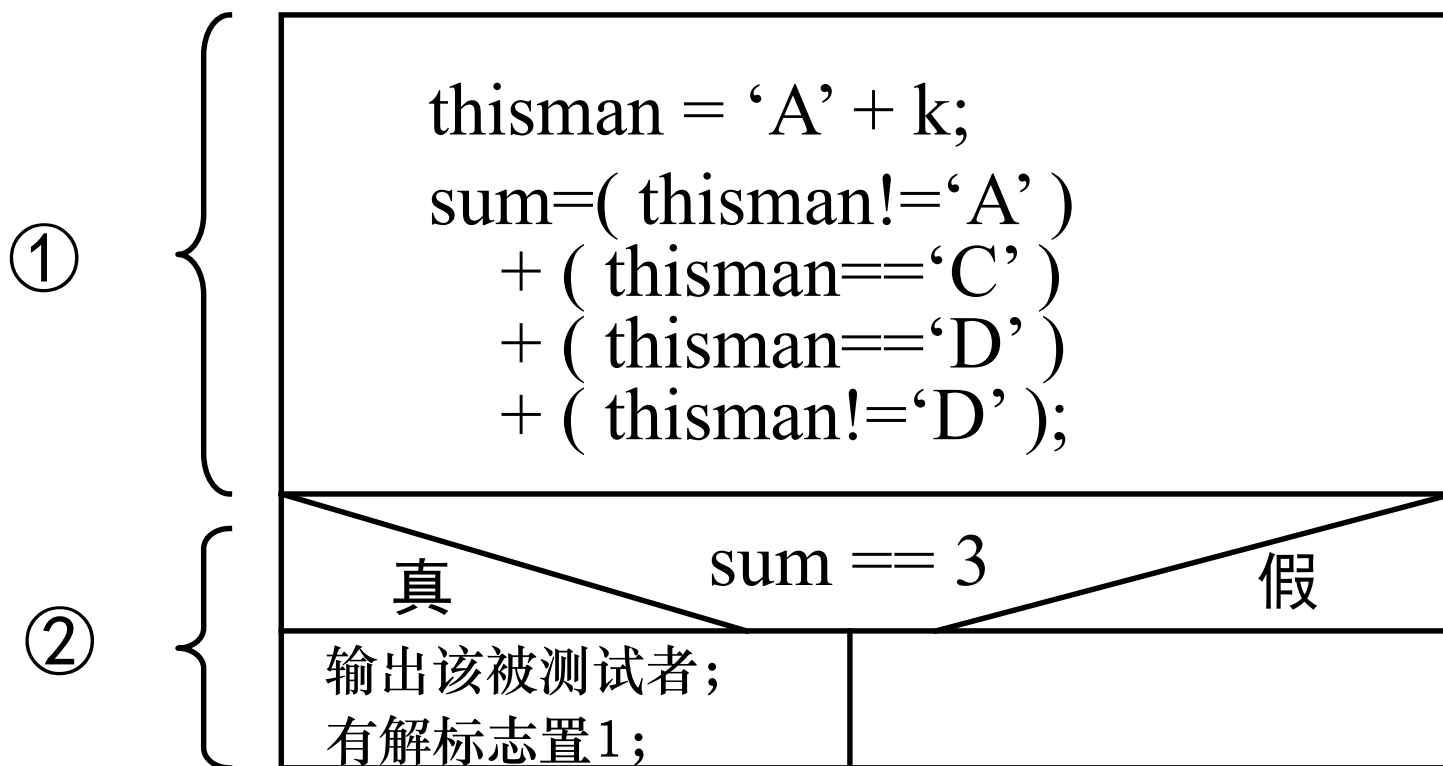
第二块
分支结构



- 再细看，第一块的循环体又由两块组成，如图4.9，

①中含两条赋值语句

②中含一条分支语句



```

1.  #include <stdio.h>    // 预编译命令
2.  int main() {          // 主函数
3.      int k=0,sum=0,g=0; //声明整数变量, 且均初始化为0
4.      char thisman=' ';

5.      for(k=0; k<4; k++) { //k是计数器, 表示第k+1个人
6.          thisman = 'A'+k;
7.          sum =( thisman!='A' )
8.              + ( thisman=='C' )
9.              + ( thisman=='D' )
10.             + ( thisman !='D' )
11.         +
12.         +;
13.         if (sum==3) { // 如果4句话有3句话为真, 则输出该人
14.             printf( "做好事者为%c\n ", thisman );
15.             g=1;      // 有解标志置1
16.         }
17.     }
18.     if (g!=1)          // 则输出无解信息
19.         printf( "Can' t found!\n ");

```

程序讨论

■ 变量设计

- 变量k: 为什么k初始化为0, 而不是1?
- 变量g: 为什么设计变量g?

■ 程序结构

- `for(k=0; k<4; k++)`
- 语句分号分隔的部分作用分别是什么?
- 根据题意, 能否提前终止循环?

示例2：找出作案人

- 某地刑侦队对涉及六个嫌疑人的一桩疑案进行案情分析：
 - A、B至少有一人作案；
 - A、D 不可能是同案犯；
 - A、E、F三人中至少有两人参与作案；
 - B、C或同时作案,或与本案无关；
 - C、D 中有且仅有一人作案；
 - 若 D 没参与，则 E 也不可能参与。
- 试编写程序将作案人找出来。

一脸懵逼



解题思路 - 「道」

■ 核心思路

枚举法

■ 比上例复杂的地方

- “一位做了好事” VS 人人皆可犯案
- “是”、“不是” VS “同案犯”、“有且仅有”、“若...则...”

回顾高中「排列组合」知识

- 有六个嫌疑人，人人皆可能犯案，总共有多少种可能的犯案情况？
- 乘法原理
 - 嫌疑人A有两种可能性：犯案、不犯案
 - 嫌疑人B有两种可能性：犯案、不犯案
 -
 - 总共有 $2^6 = 64$ 种可能性
- 如何设计变量？
 - 设计6个01变量：1表示参与；0表示没参与

回顾「逻辑表达式」

- 如何对复杂的“案情”进行建模？
- 现有两队进行篮球比赛，设变量A和B表示A和B两队是否到场，指出下面表达式的含义，判断比赛是否能顺利进行
 - 表达式1： $A \ \&\& \ B$
 - 表达式2： $A \ || \ B$
 - 表达式3： $A \ \&\& \ !B$
 - 表达式4： $!(!B \ || \ !A)$

变量设计

- 六个人每个人都有作案或不作案两种可能,因此有64 (2^6) 种组合,从这些组合中挑出符合条件的作案者。
- 定义6个整型变量,分别表示六个人A - F
- 如何枚举每个人的可能性?
 - 取值 0 表示没有参与作案;
 - 取值 1 表示参与作案

案情建模

- 基于定义的变量，如何做案情分析？
 - A、B至少有一人作案；
 - A、D 不可能是同案犯；
 - A、E、F三人中至少有两人参与作案；
 - B、C或同时作案,或与本案无关；
 - C、D 中有且仅有一人作案；
 - 如果 D 没有参与作案,则 E 也不可能参与。

用逻辑表达式建模

对案情逐一建模 (1)

- 已知：变量A和B分别表示A和B作案
 - 怎么表示 “A、B至少有一人作案” ？
 - 表达式：CC1 = A || B
- 已知：变量A和D分别表示A和D作案
 - 怎么表示 “A、D不可能是同案犯” ？
 - 考虑相反事件：A和D是同案犯 → A && D
 - 表达式：CC2 = !(A && D)

对案情逐一建模 (2)

■ 考虑三个变量A、E、F

- 怎么表示 “A、E、F三人中至少有两人参与作案” ？
- 可能有几种情况发生
 - 两个人A和E同时参与作案 $\rightarrow A \ \&\& \ E$
 - 两个人A和F同时参与作案 $\rightarrow A \ \&\& \ F$
 - 两个人E和F同时参与作案 $\rightarrow E \ \&\& \ F$
- 表达式： $CC3 = (A\&\&E) \parallel (A\&\&F) \parallel (E\&\&F)$

对案情逐一建模 (3)

■ 考虑两个变量B、C

- 怎么表示 “B、C或同时作案或同时与本案无关” ?
- 可能有几种情况发生
 - 两个人B和C同时参与作案 $\rightarrow B \ \&\& \ C$
 - 两个人A和F与本案无关 $\rightarrow !B \ \&\& \ !C$
- 表达式: $CC4 = (B\&\&C) \ || \ (!B\&\&!C)$

对案情逐一建模 (4)

■ 考虑两个变量C、D

- 怎么表示 “C、D中有且仅有一人作案” ？
- 可能有几种情况发生
 - 只有C作案，D没作案 $\rightarrow C \ \&\& \ !D$
 - 只有D作案，C没作案 $\rightarrow !C \ \&\& \ D$
- 表达式： $CC5 = (C\&\&!D) \ || \ (!C\&\&D)$

对案情逐一建模 (5)

■ 考虑两个变量D、E

- 怎么表示 “如果D没有参与作案，则E也不可能参与作案” ？
- 可能有几种情况发生
 - 如果D没有参与作案，E也没作案 $\rightarrow !D \ \&\& \ !E$
 - 如果D参与作案，则E既作案也没作案 $\rightarrow D$
- 表达式： $CC6 = (!D \ \&\& \ !E) \ || \ (D)$

案情建模

- 如何表示所有的案情分析都成立

$$CC1+CC2+\dots+CC6==6$$

- 与“谁做了好事”案例的区别
 - 简单的关系表达式：“=”与“!=”
 - 复杂的逻辑表达式
- 相同点：逻辑建模的思路

万事俱备，只欠.....

如何进行枚举呢？

- 方法1：沿用上例的单循环语句
 - `for(k=0; k<4; k++)`
- 方法2：

多重循环！

什么是多重循环？

- 考虑下面的例子：
- 编一个程序输出所有的排列数
 - 000000
 - 000001
 - 000010
 - 000011
 -
 - 111111

for (A=0; A<=1; A=A+1)

for (B=0; B<=1; B=B+1)

for (C=0; C<=1; C=C+1)

for (D=0; D<=1; D=D+1)

for (E=0; E<=1; E=E+1)

for (F=0; F<=1; F=F+1)

CC1=A||B;

CC2=! (A&&D);

CC3=(A&&E)|| (A&&F) || (E&&F);

CC4=(B&&C)|| (!B&&!C);

CC5=(C&&!D)|| (D&&!C);

CC6=D|| (!E);

真

CC1+CC2+CC3+CC4+CC5+CC6==6

假

输出

思路小节

- 枚举思想
 - 完备考虑所有可能性(体现在代码中)
 - 逐一处理 (使用循环语句)
- 自然语言描述-->数学语言描述
 - 以实现其 “可计算性”
 - 关系表达式与逻辑表达式
- 数学语言表达-->程序语言表达
 - 满足其 “可行性” ,变成可执行的程序

编程思考题

- 假设有8间教室，容量分别为C1 – C8。现考虑四个班级，人数分别为N1 – N4。如果班级人数小于等于教室容量，就可以把教室分配给该班级，共有多少种可行的分配方案
- 编写一个程序，输出下面的内容

A
ABA
ABCBA
ABCDCA
ABCDEDCBA

10-08上机

➤ 10-08上机

➤ #7. 求三角形面积

➤ 加入if语句判断是否构成三角形

➤ #8. 计算学分绩点

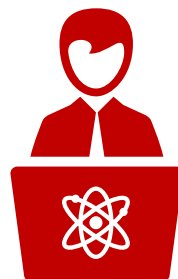
➤ 需要用到if语句

➤ #9. 按要求读入数据并输出

➤ 练习课堂上讲的例子：“谁做好事” + “鉴别6个案犯”



中國人民大學
RENMIN UNIVERSITY OF CHINA



谢谢大家!

