



## 4.2 分支程序设计技术

### 顺序程序

是指程序中的每一条指令都是按指令的排列顺序执行的，它是最简单的一种程序结构。这在程序段中是大量存在的，但作为完整的程序则很少见。(参考前例)



## 4.2 分支程序设计技术

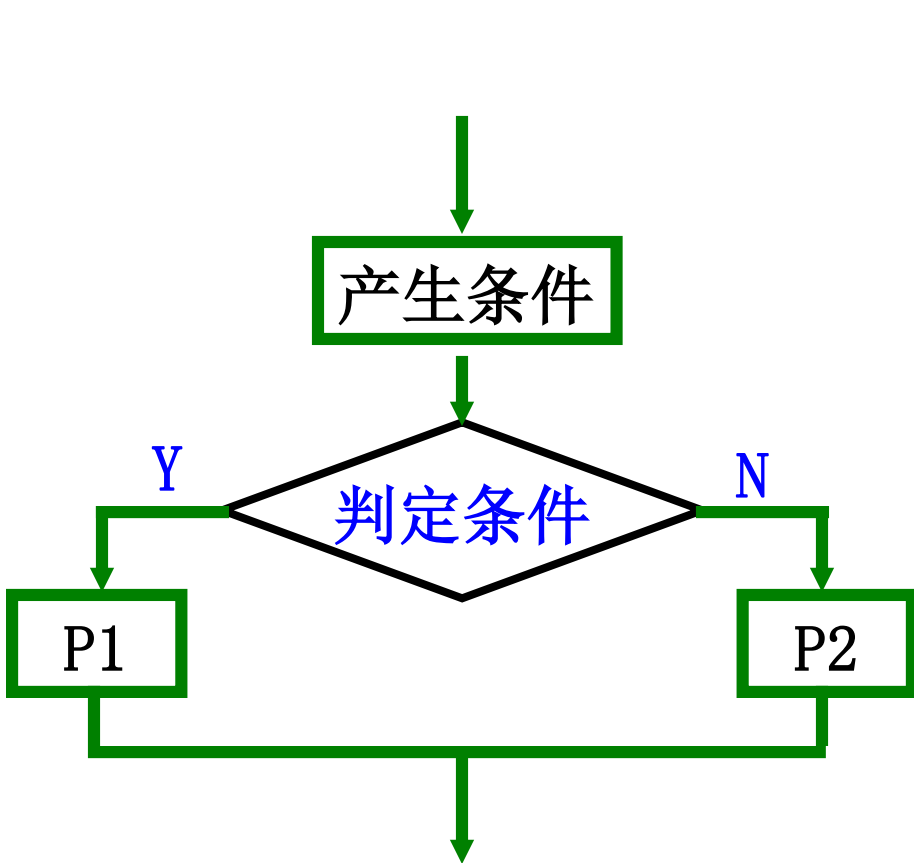
### 分支程序

#### 1. 分支程序的结构形式

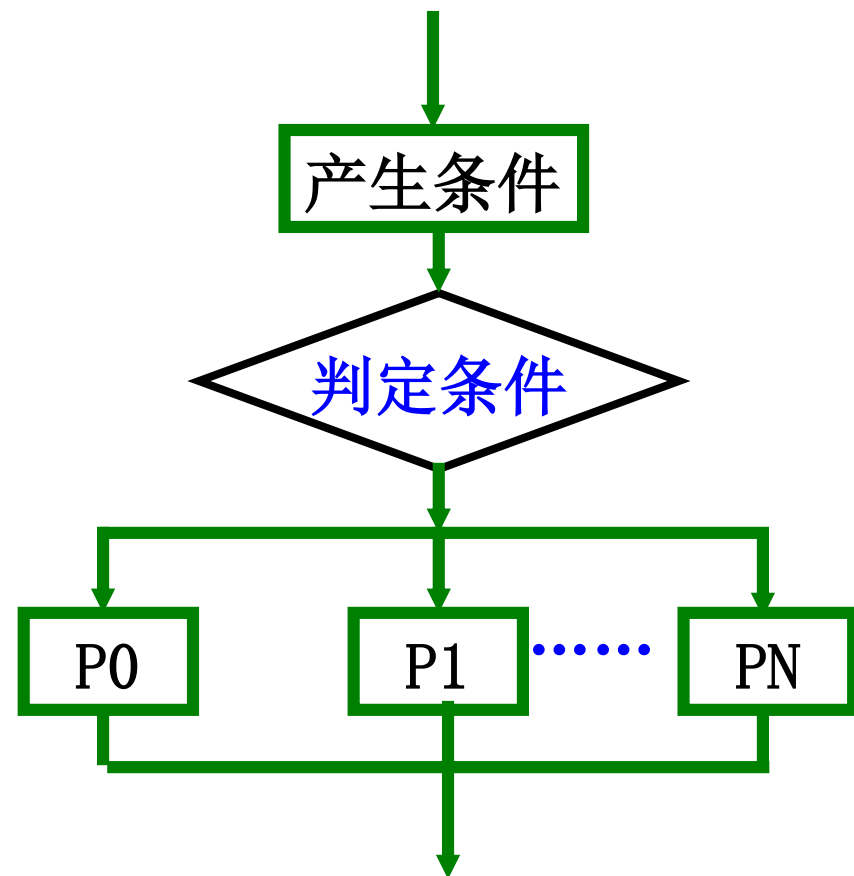
计算机在完成某种运算或某个过程的控制时，经常需要根据不同的情况（条件）实现不同的功能，这就要求在程序的执行过程中能够进行某种条件的判定，并根据判定结果决定程序的流向，这就是分支程序。分支程序的两种形式：



## 4.2 分支程序设计技术



(A) 单分支程序结构  
(IF-THEN-ELSE结构)



(B) 多分支程序结构  
(CASE结构)



## 4.2 分支程序设计技术

### 2. 转移指令

- 无条件转移指令 JMP
- 条件转移指令 (回顾)



## 4.2 分支程序设计技术

### 无条件转移指令

格式:

**JMP LABEL** ; 转移到标号**LABEL**处执行程序

**JMP REG16** ; 转移到由通用寄存器**REG**指定的位置执行程序

**JMP MEM** ; 转移到由存储单元**MEM**指定的位置执行程序



## 4.2 分支程序设计技术

### 条件转移指令

根据单个标志位的状态进行转移

指令格式		测试条件	功能
JC	LABEL	(CF=1)	有进/借位
JNC	LABEL	(CF=0)	无进/借位
JE/JZ	LABEL	(ZF=1)	相等
JNE/JNZ	LABEL	(ZF=0)	不相等
JS	LABEL	(SF=1)	负数
JNS	LABEL	(SF=0)	正数
JO	LABEL	(OF=1)	有溢出
JNO	LABEL	(OF=0)	无溢出
JP/JPE	LABEL	(PF=1)	有偶数个1
JNP/JPO	LABEL	(PF=0)	有奇数个1



## 4.2 分支程序设计技术

### 无符号数比较

<b>JA/JNBE LABEL</b>	<b><math>(CF=0) \wedge (ZF=0)</math></b>	<b>高于/不低于等于</b>
<b>JAE/JNB LABEL</b>	<b><math>(CF=0)</math></b>	<b>高于等于/不低于</b>
<b>JB/JNAE LABEL</b>	<b><math>(CF=1)</math></b>	<b>低于/不高于等于</b>
<b>JBE/JNA LABEL</b>	<b><math>(CF=1) \vee (ZF=1)</math></b>	<b>低于等于/不高于</b>

### 有符号数比较

<b>JG/JNLE LABEL</b>	<b><math>((SF \neq OF) \vee ZF)=0</math></b>	<b>大于/不小于等于</b>
<b>JGE/JNL LABEL</b>	<b><math>(SF \neq OF)=0</math></b>	<b>大于等于/不小于</b>
<b>JL/JNGE LABEL</b>	<b><math>(SF \neq OF)=1</math></b>	<b>小于/不大于等于</b>
<b>JLE/JNG LABEL</b>	<b><math>((SF \neq OF) \vee ZF)=1</math></b>	<b>小于等于/不大于</b>



## 4.2 分支程序设计技术

### 3. 分支程序设计时必须注意下面几个要点

- (1) 正确选择判定条件和相应的条件转移指令;
- (2) 在编程时必须保证每条分支都有完整的结果;
- (3) 在检查和调试时必须逐条分支进行, 因为一条或其中几条分支正确还不足以说明整个程序正确。





## 4.2 分支程序设计技术

例：编写汇编程序实现从键盘输入一个数字，然后判断其奇偶性；如果为奇数，则在屏幕输出1，否则输出0.

思路：

该问题为典型的分支结构程序，主要要解决以下几点：

- 接受用户的输入(调用DOS中断的01H号功能实现)
- 奇偶数的判断(利用RCR循环右移一位得到输入数值的最低位，判断是0还是1)
- 根据奇偶数条件的跳转
- 条件输出（调用DOS中断的02H号功能实现）



## 4.2 分支程序设计技术

CODE SEGMENT

ASSUME CS:CODE

START:

MOV AH,01H ;调用DOS中断的01号功能，输入字符，存储于AL中

INT 21H

CMP AL,'0' ;输入字符存在AL寄存器中，使其与0比较

JB START ;如果其ASC码比'0'小，则继续输入

CMP AL,'9' ;与9比较

JA START ;如果比输入的ASC码比'9'还大，则继续输入

CLC ;CF标志清零

RCR AL,1 ;将AL的低1位右移出到CF中

JNC EVE ;如果CF!=1,则判断此数为偶数，则跳转到偶数的操作

MOV AL, 31H ;如果没有跳转，则为奇数，则输出字符为1，及其ASCII  
;码为31H



## 4.2 分支程序设计技术

JMP DISP ;强制跳转到输出分支

EVE:

MOV AL,30H ;0的ASC码为30H

DISP:

MOV AH,02H ;调用DOS中断的02H功能，实现输出字符

MOV DL,AL ;输入字符即为AL，将其赋值给DL寄存器并输出

INT 21H

MOV AX,4C00H

INT 21H

CODE ENDS

END START