

第2章 算法——程序的灵魂

1. 什么是算法？试从日常生活中找3个例子，描述它们的算法。

解：略。

2. 什么叫结构化的算法？为什么要提倡结构化的算法？

解：略。

3. 试述3种基本结构的特点，请自己另外设计两种基本结构（要符合基本结构的特点）。

解：见图2.1和图2.2。

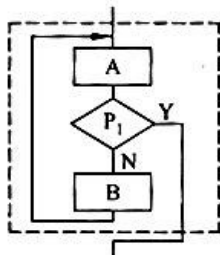


图 2.1

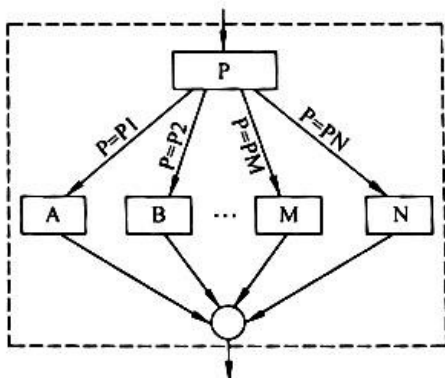


图 2.2

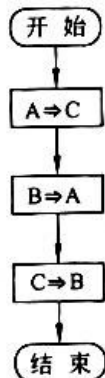


图 2.3

4. 用传统流程图表示求解以下问题的算法。

(1) 有两个瓶子A和B，分别盛放醋和酱油，要求将它们互换（即A瓶原来盛醋，现改盛酱油，B瓶则相反）。

解：显然，如果只有两个瓶子，肯定不能完成此任务，必须有一个空瓶C作为过渡，其步骤见图2.3。

(2) 依次将10个数输入，要求将其中最大的数输出。

解：流程图见图2.4。

(3) 有3个数 a, b, c ，要求按大小顺序把它们输出。

解：流程图见图2.5。

(4) 求 $1+2+3+\cdots+100$ 。

解：流程图见图2.6。

(5) 判断一个数 n 能否同时被3和5整除。

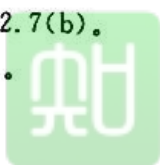
解：流程图见图2.7(a)或图2.7(b)。

(6) 将100~200的素数输出。

解：流程图见图2.8。



成惠资料订购链接



知否大学
— 微信公众号同名 —

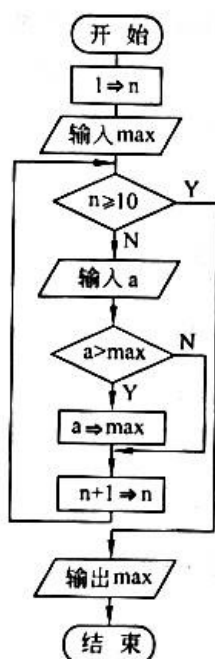


图 2.4

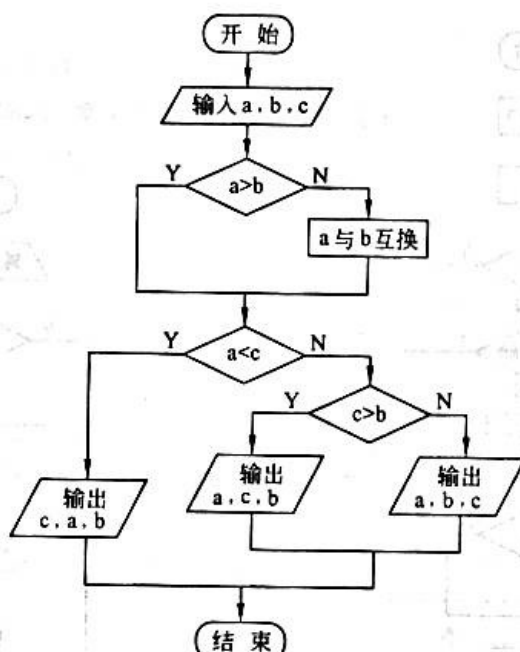


图 2.5

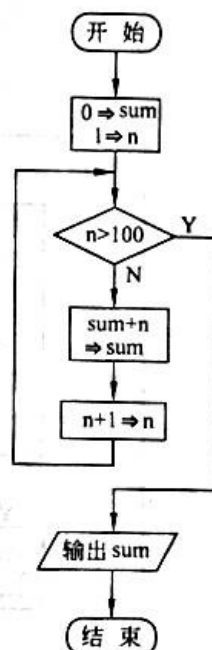
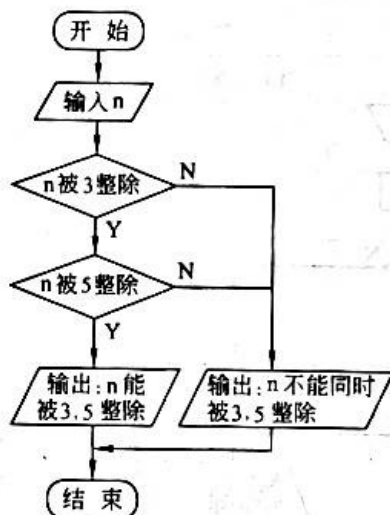
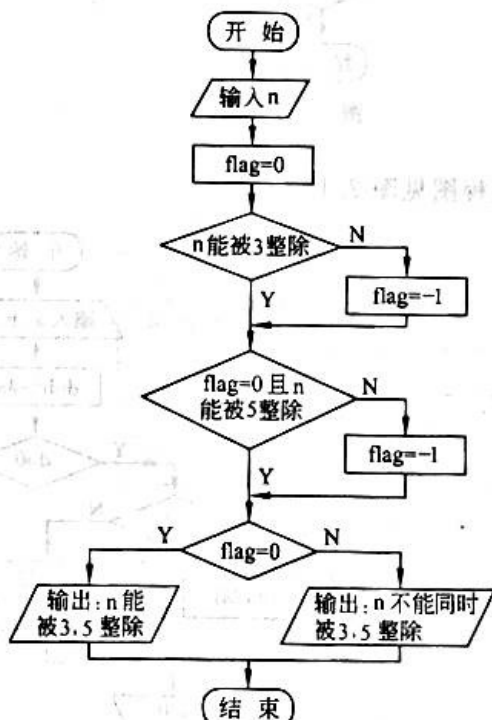


图 2.6



(a)



(b)

图 2.7

(7) 求两个数 m 和 n 的最大公约数。

解: 流程图见图 2.9。

(8) 求方程 $ax^2+bx+c=0$ 的根。分别考虑: ①有两个不等的实根; ②有两个相等的实根。

找课后习题答案

下载「知云大学」APP

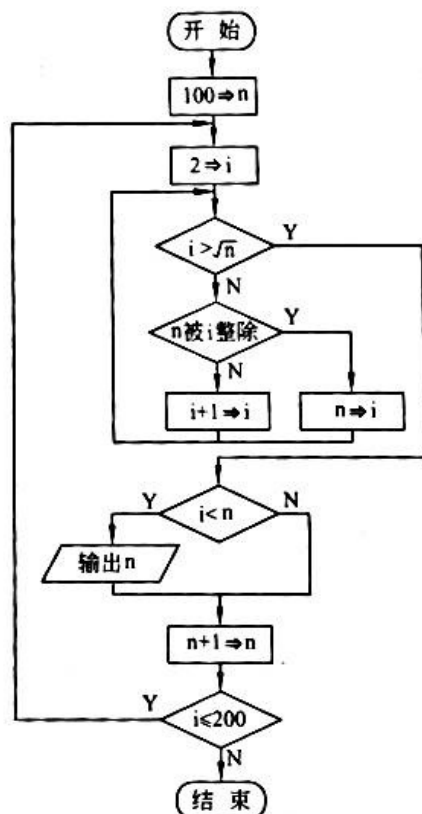


图 2.8

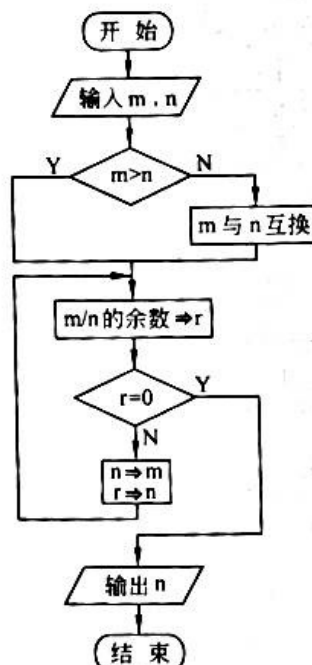


图 2.9

解: 流程图见图 2.10。

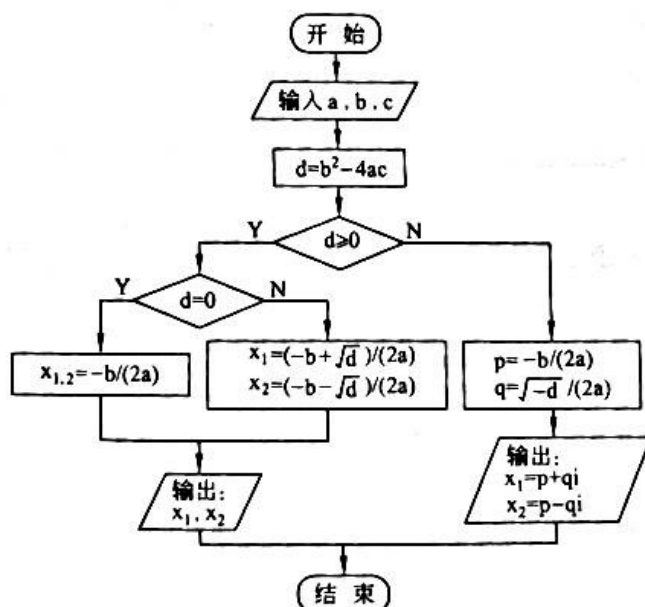


图 2.10

5. 用 N-S 图表示第 4 题中各题的算法。

(1) 有两个瓶子 A 和 B, 分别盛放醋和酱油, 要求将它们互换(即 A 瓶原来盛醋, 现改

— 微信公众号同名 —

盛酱油,B瓶则相反)。

解: N-S流程图见图 2.11。

(2) 依次将 10 个数输入,要求将其中最大的数输出。

解: N-S流程图见图 2.12。

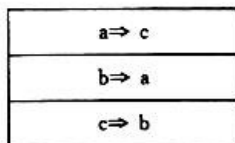


图 2.11

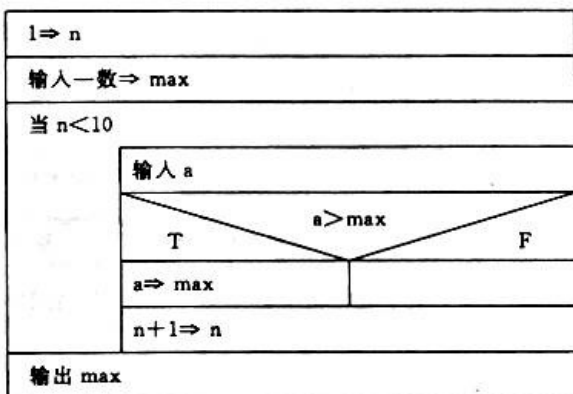


图 2.12

(3) 有 3 个数 a, b, c , 要求按大小顺序把它们输出。

解: N-S流程图见图 2.13。

(4) 求 $1+2+3+\cdots+100$ 。

解: N-S流程图见图 2.14。

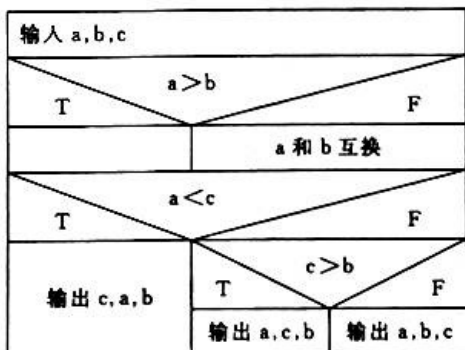


图 2.13

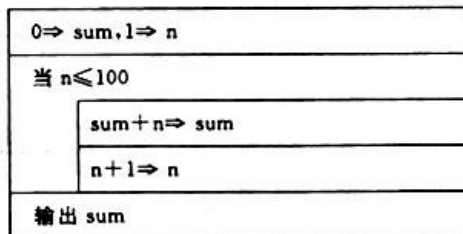


图 2.14

(5) 判断一个数 n 能否同时被 3 和 5 整除。

解: N-S流程图见图 2.15。

(6) 将 100~200 的素数输出。

解: 流程图见图 2.16。

(7) 求两个数 m 和 n 的最大公约数。

解: 流程图见图 2.17。

(8) 求方程式 $ax^2+bx+c=0$ 的根。分别考虑: ①有两个不等的实根; ②有两个相等的实根。

解: 流程图见图 2.18。

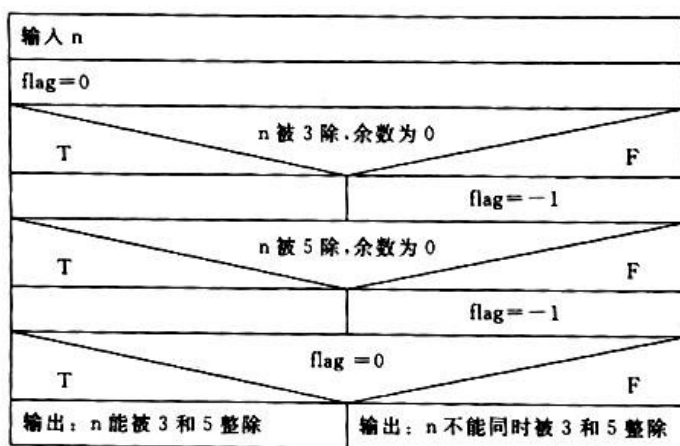


图 2.15

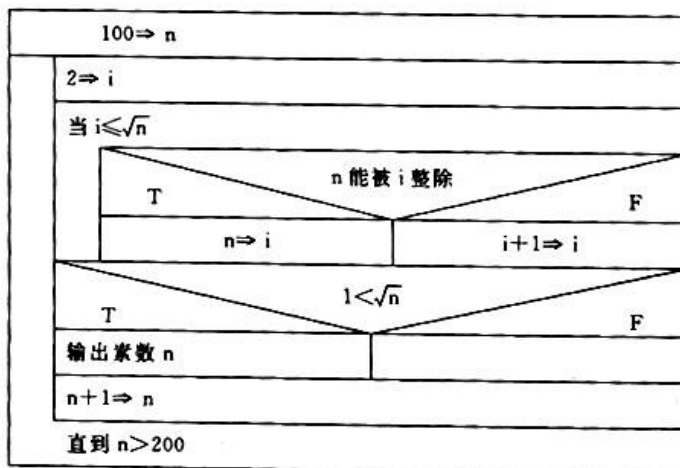


图 2.16

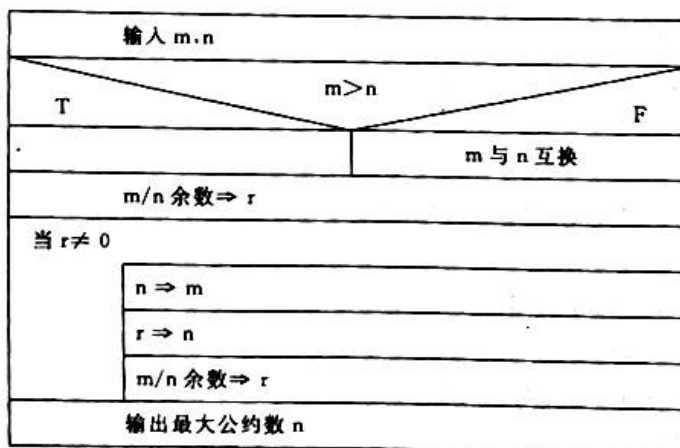
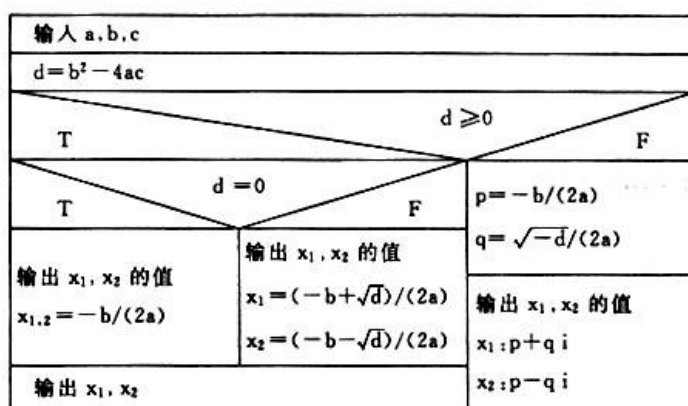


图 2.17



知否大学
— 微信公众号同名 —



注: i 为 $\sqrt{-1}$ 。

图 2.18

6. 用伪代码表示第4题中各题的算法。

(1) 有两个瓶子 A 和 B, 分别盛放醋和酱油, 要求将它们互换(即 A 瓶原来盛醋, 现改盛酱油, B 瓶则相反)。

解:

$c = a$

$a = b$

$b = c$

(2) 依次将 10 个数输入, 要求将其中最大的数输出。

解:

$n = 1$

input max

while $n < 10$ do

input a

if $a > \text{max}$ then $\text{max} = a$

$n = n + 1$

end do

print max

(3) 有 3 个数 a, b, c , 要求按大小顺序把它们输出。

解:

input a, b, c

if $a < b$ then swap a, b

(swap a, b 表示 a 和 b 互换)

if $a < c$ then

print c, a, b

else

if $c > b$ then

print a, c, b

```
    else  
        print a,b,c  
    end if  
end if
```

(4) 求 $1+2+3+\cdots+100$ 。

解:

```
sum=0  
n=1  
while n≤100 do  
    sum=sum+n  
    n=n+1  
end do  
print sum
```

(5) 判断一个数 n 能否同时被 3 和 5 整除。

解:

```
input n  
flag=0  
if mod(n,3)≠0 then flag=-1  
if mod(n,5)≠0 then flag=-1  
if flag=0 then  
    print n "能被 3 和 5 整除"  
else  
    print n "不能同时被 3 和 5 整除"  
end if
```

(6) 将 100~200 的素数输出。

解:

```
n=100  
while n≤200 do  
    i=2  
    while i≤√n  
        if mod(n,i)=0 then  
            i=n  
        else  
            i=i+1  
        end if  
    end do  
    if i≤√n then print n  
    n=n+1  
end do
```



知否大学

- 微信公众号同名 -

(7) 求两个数 m 和 n 的最大公约数。

解:

```
input m,n
if m<n then swap m,n
t=mod(m,n)
while t≠0 do
    m=n
    n=t
    t=mod(m,n)
end do
print n
```

(8) 求方程式 $ax^2+bx+c=0$ 的根。分别考虑: ①有两个不等的实根; ②有两个相等的实根。

解:

```
int a,b,c
disc=b2-4ac
if disc≥0 then
    if disc=0 then
        x1,x2=-b/(2a)
    else
        x1=(-b+√disc)/(2a)
        x2=(-b-√disc)/(2a)
    end if
    print x1,x2
else
    p=-b/(2a)
    q=√disc/(2a)
    print p+q,"+",p-q,"i"
end if
```

7. 什么叫结构化程序设计? 它的主要内容是什么?

解: 略。

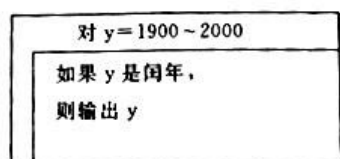
8. 用自顶向下、逐步细化的方法进行以下算法的设计:

(1) 输出 1900~2000 年中是闰年的年份, 符合下面两个条件之一的年份是闰年: ①能被 4 整除但不能被 100 整除; ②能被 100 整除且能被 400 整除。

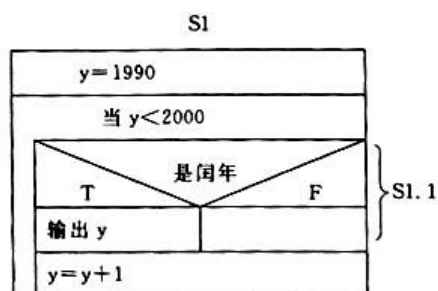
解: 先画出图 2.19(a), 对它细化得图 2.19(b); 对图 2.19(b) 中的 S1.1 细化得图 2.19(c)。

(2) 求 $ax^2+bx+c=0$ 的根。分别考虑 $d=b^2-4ac$ 大于 0、等于 0 和小于 0 这 3 种情况。

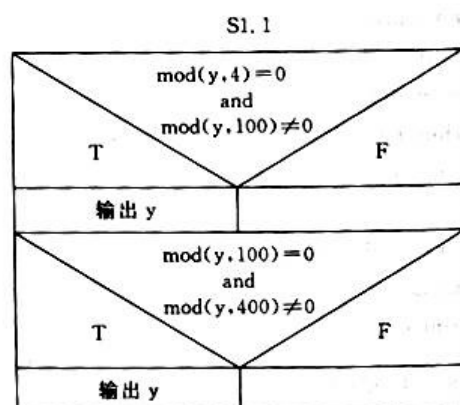
解: 先画出图 2.20(a), 对其中的 S3 细化为图 2.20(b), 对图 2.20(b) 中的 S3.1 细化为图 2.20(c), 对图 2.20(c) 中的 S3.1.1 细化为图 2.20(d), 对图 2.20(c) 中的 S3.1.2 细化为



(a)



(b)



(c)

图 2.19

图 2.20(e),再对图 2.20(b)中的 S3.2 细化为图 2.20(f)。请读者将它们合成一个总的 N-S 图。

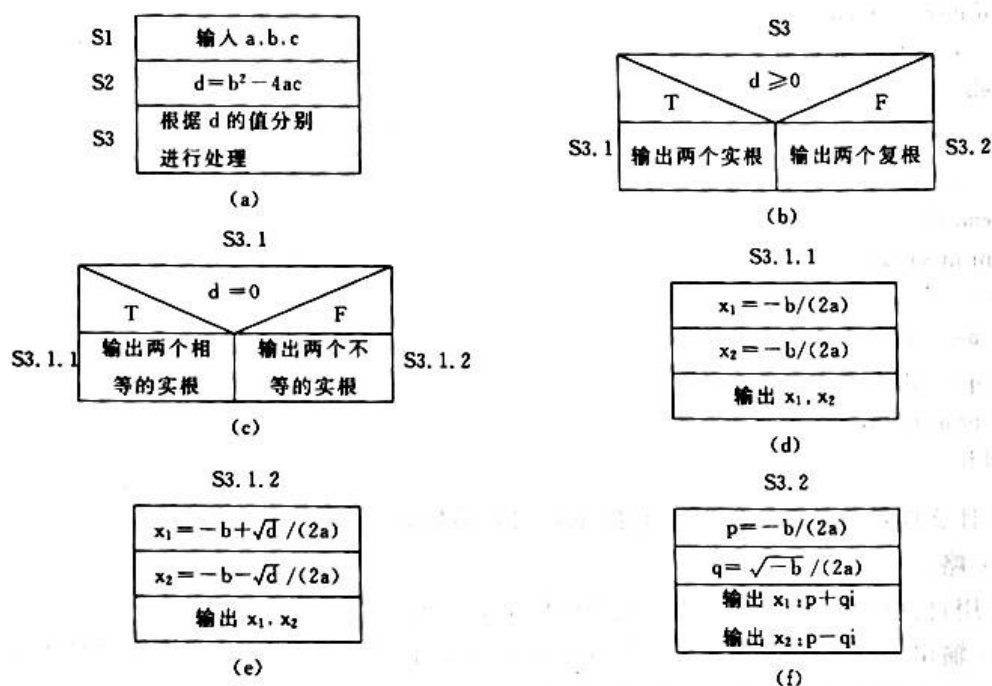
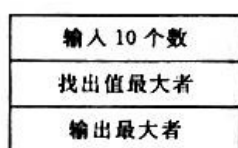


图 2.20

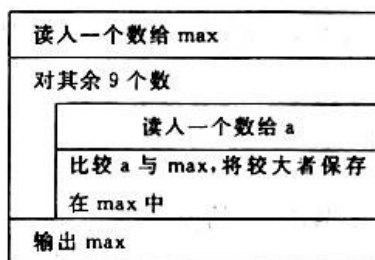
(3) 输入 10 个数,输出其中最大的一个数。

解: 先初步画出图 2.21(a)。考虑到还没有学习数组的知识,因而不能做到将 10 个数全部输入给数组中各个元素,然后再从中找最大者。由于不采用数组这种数据结构,算法也应与采用数组时有所不同。现在只用普通变量,逐个读入数据,将当时各数中的最大者保留

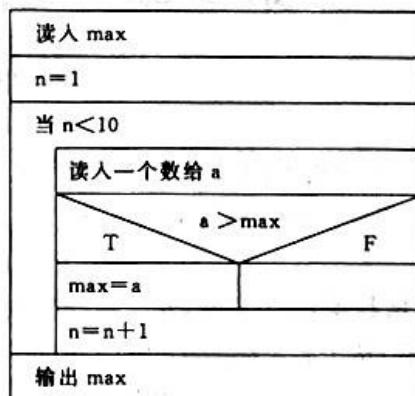
下来存放在 \max 中,以便再与后面读入的数比较。将图 2.21(a)细化为图 2.21(b),再细化为图 2.21(c),



(a)



(b)



(c)

图 2.21



知否大学
— 微信公众号同名 —

