



算法与流程图

BASIC PROGRAM DISEGN 
■ 程序设计基础

程序构成公式

- **Nikiklaus Wirth 提出的公式：**

程序 = 数据结构 + 算法

- **目前已经修改为**

程序 = 算法 + 数据结构 + 程序设计方法 + 语言工具和环境

算法及其表示

- **算法：解决问题所采取的一系列步骤**

例，要求计算圆的面积，算法为：

设置（输入或指定）半径值 r ；

使用公式 $s = \pi r^2$ ；

显示（输出） S 的值。

算法的特点

(1)有输入

算法可以有零个或多个输入。在一个算法的执行过程中，提供需处理的数据或控制算法执行过程的信息。

(2)有输出

算法必须具有一个或多个执行结果的输出。没有输出的算法是一个无效的算法。

算法的特点

(3)有穷性

任何算法都应该在执行有穷步骤之后结束。

(4)确定性

算法不能具有二义性。算法中每一步的语义都应该清晰明了，明确指出应该执行什么操作，如何执行操作。

算法的特点

(5) 高效性

根据算法编写出来的程序应具有较高的时空效率:

执行时间短

不占用过多内存。

算法的表示（描述）

- **自然语言表示**

易理解和交流，但易产生二义性。

- **伪代码表示**

伪代码使用介于自然语言和计算机语言之间的文字和符号来描述算法

算法的表示（描述）

- **程序流程图**

用图形符号和文字说明来表示数据处理的过程和步骤。

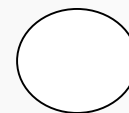
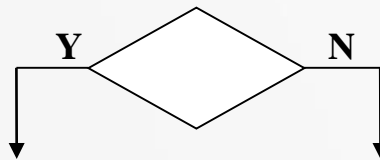
- **N-S流程图**

也称方框图；适于结构化程序设计的算法描述工具。

程序流程图

- 描述算法的良好工具。
- 逻辑框和流向线组成。
- 逻辑框是表示功能的图形符号
- 流向线指示逻辑处理顺序

逻辑框和流向线

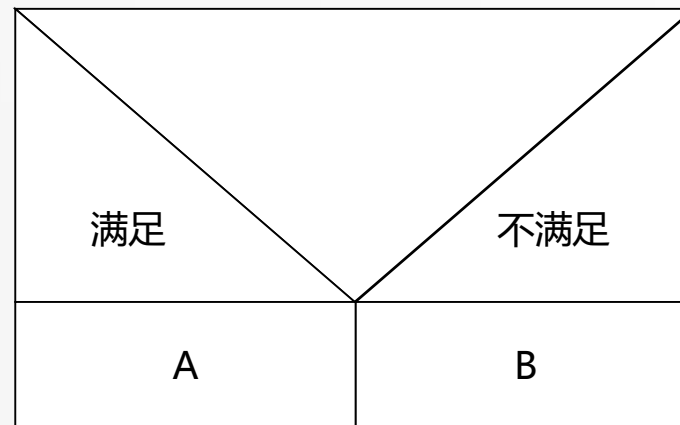
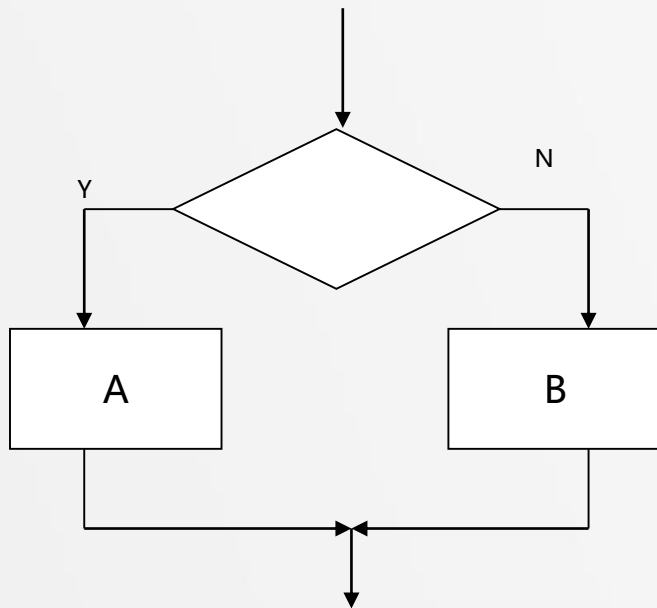


N-S流程图

1973年美国学者Nassi和Schneiderman提出N-S流程图（也称为方框图），它是一种适于结构化程序设计的算法描述工具。

由于流程图各步骤之间，一般总是按照从上到下顺序执行(N-S流程图中取消了流向线)

二选一结构的表示



例子1

输入1个整数，输出其绝对值

- 自然语言描述：

- 1 输入一个整数;

- 2 若该数为正数，则其绝对值取该数的值

否则，其绝对值取该数的相反数值；

- 3 输出其绝对值。

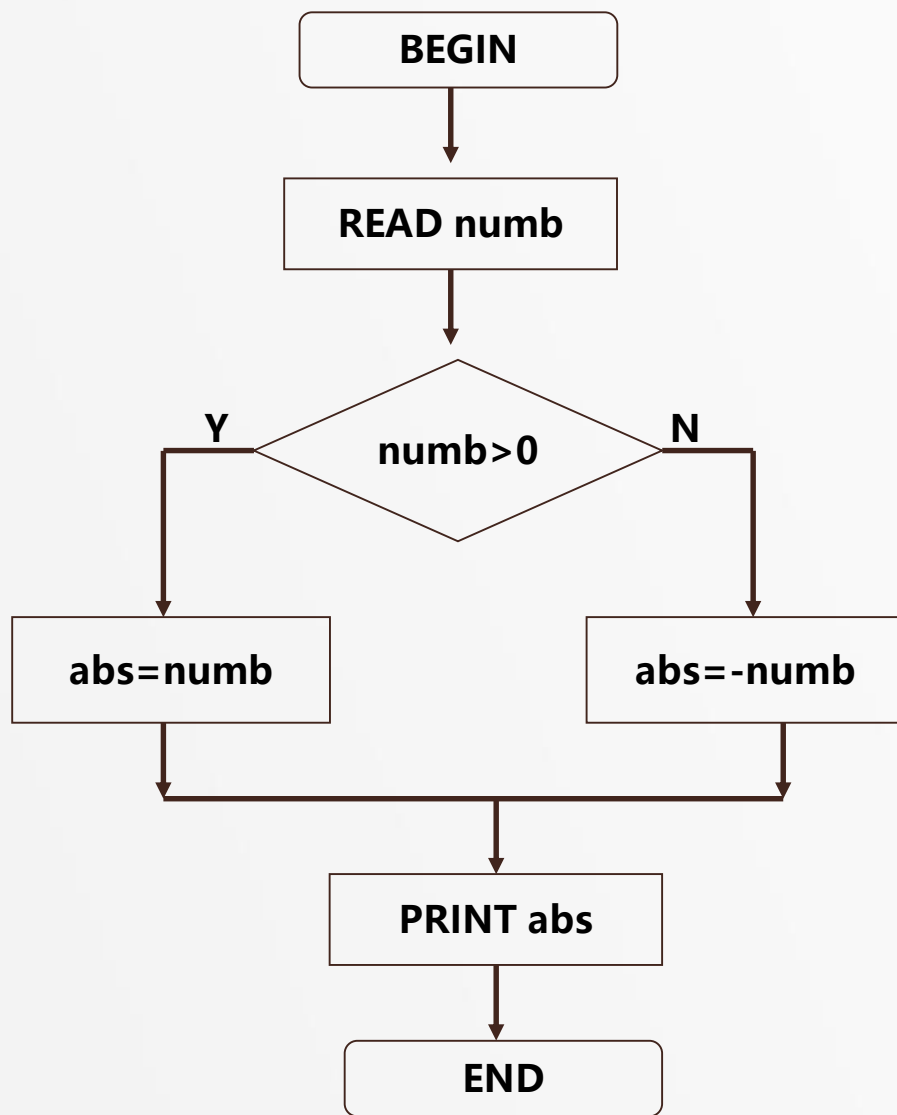
伪码表示

```
BEGIN
  READ numb
  IF numb > 0
    abs = numb
  ELSE
    abs = -numb
  PRINT abs
END
```

对比

```
BEGIN
  READ numb
  IF numb < 0
    numb = -numb
  PRINT numb
END
```

程序流程图



```
#include <stdio.h>
void main( )
{
    int numb; /*代表一个整数*/
    int abs; /*代表运算结果：绝对值*/
    scanf_s( "%d" ,&numb); /*输入整数的值*/
    if ( numb>0) abs=numb;
        else abs= -numb;
    printf ( "numb 的绝对值是： %d\n" ,abs);
}
```


练习题

求长方体的体积

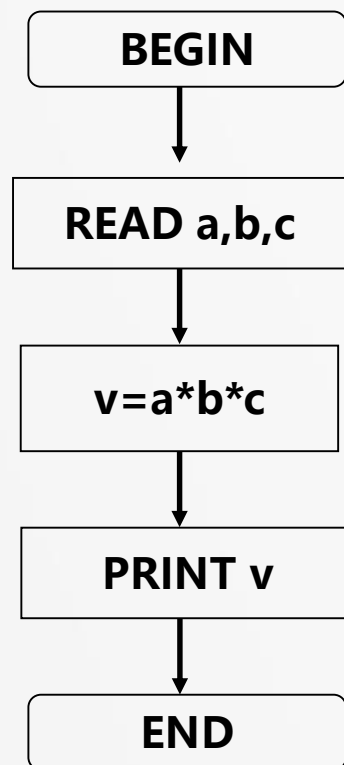
要求：

根据算法思想画出流程图，根据流程图写出程序代码并测试程序

算法：

- 1 输入三个实数 a, b, c ;
- 2 根据长方体体积计算公式： $V = a * b * c$ 计算体积；
- 3 输出体积。

程序流程图



```
/*This program calculates the volume*/
2    #include <stdio.h>
3    void main() /* 主函数 */
4    {
5        int ,b,c,v; /*定义整型变量*/
6
7        scanf("%d %d %d",&a,&b,&c); /*用键盘输入数据*/
8        v=a*b*c; /*求体积*/
9        printf("v=%d\n",v); /*输出体积v的值*/
10    }
```