计算机科学导论

——计算机算法

龚文引　博士

计算机学院

提纲

- 计算机算法的概念
- 计算机算法的表示

游戏-1

- 5个身高不同的学生，如何找出最高的学生？

游戏-1

```
#include <stdio.h>

int main()
{
        double a[5] = {15,37,86,98,69};
        double max;
        max = 0;                                    // 假设
        for (int i=0;i<5;i++)
        {
                if (a[i] > max)                     // 判断
                {
                        max = a[i];                 // 更新
                }
        }
        printf("The highest student = %f\n", max);
        return 0;
}
```

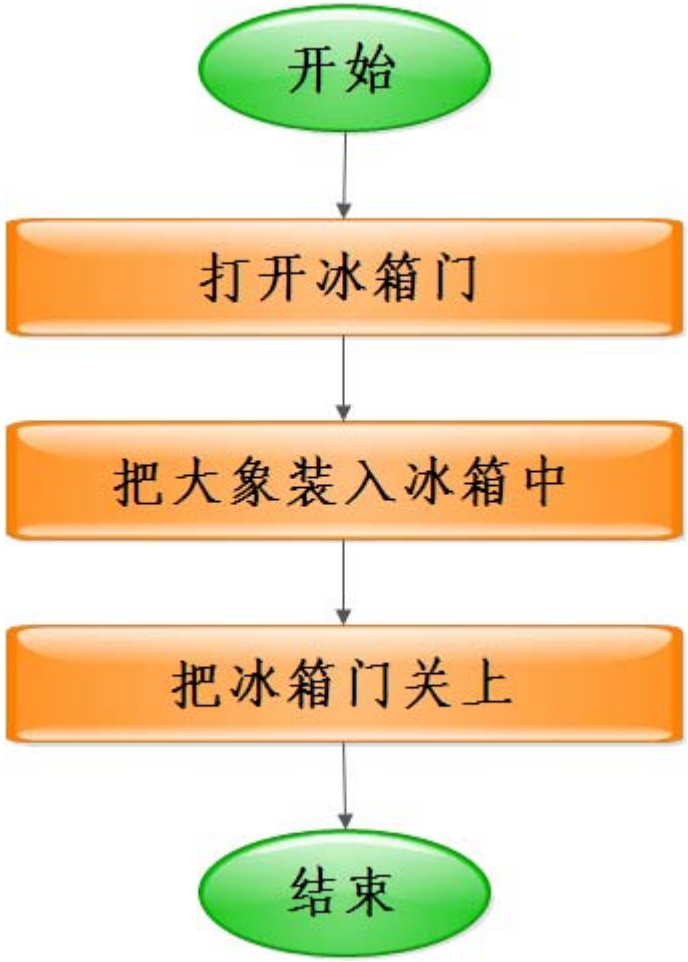
游戏-2

- 5个身高不同的学生，如何按照从高到矮的顺序排列好？

换位法

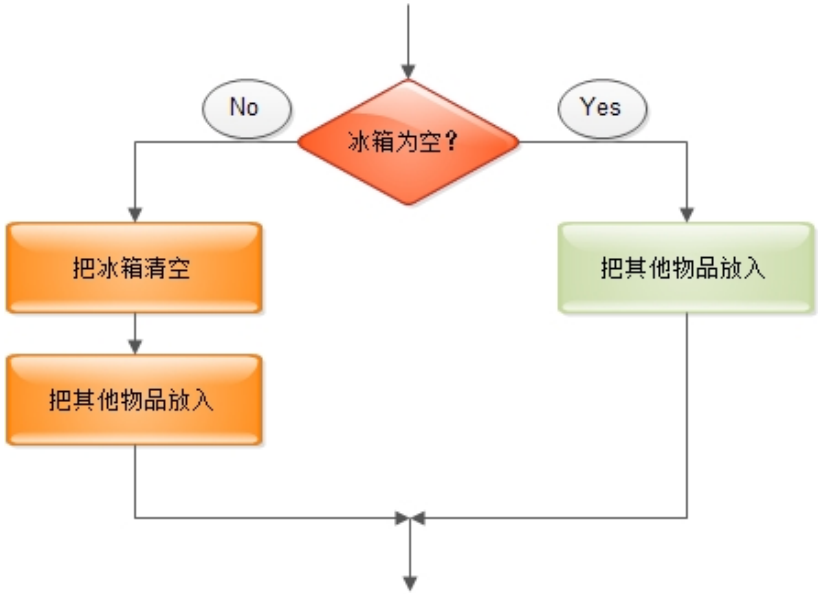
如何把大象装到冰箱里？

- 1. 把冰箱门打开；
- 2. 把大象装到冰箱里面去；
- 3. 把冰箱门关上。



如何把长颈鹿装进刚才的冰箱里？

- 1. 把冰箱门打开；
- 2. 把大象从冰箱中取出来；
- 3. 把长颈鹿装进冰箱里；
- 4. 把冰箱门关上；
- 5. 完成。



吃蟹黄汤包

- 方法：轻轻提，慢慢移，先开窗，再喝汤

吃蟹黄汤包

- 将吃一只蟹黄汤包的“算法”跟结构化程序设计的顺序结构联系起来——顺序很重要
  - (1) 将包子从蒸笼中慢慢提起,and then
  - (2) 将包子慢慢移动到面前的小碟子中,and then
  - (3) 在包子的正上方咬开一个小口,and then
  - (4) 通过小口吸食包子里的汤(当心别烫着),and then
  - (5) 将包子送入口中
  - (6) 完成!

吃蟹黄汤包

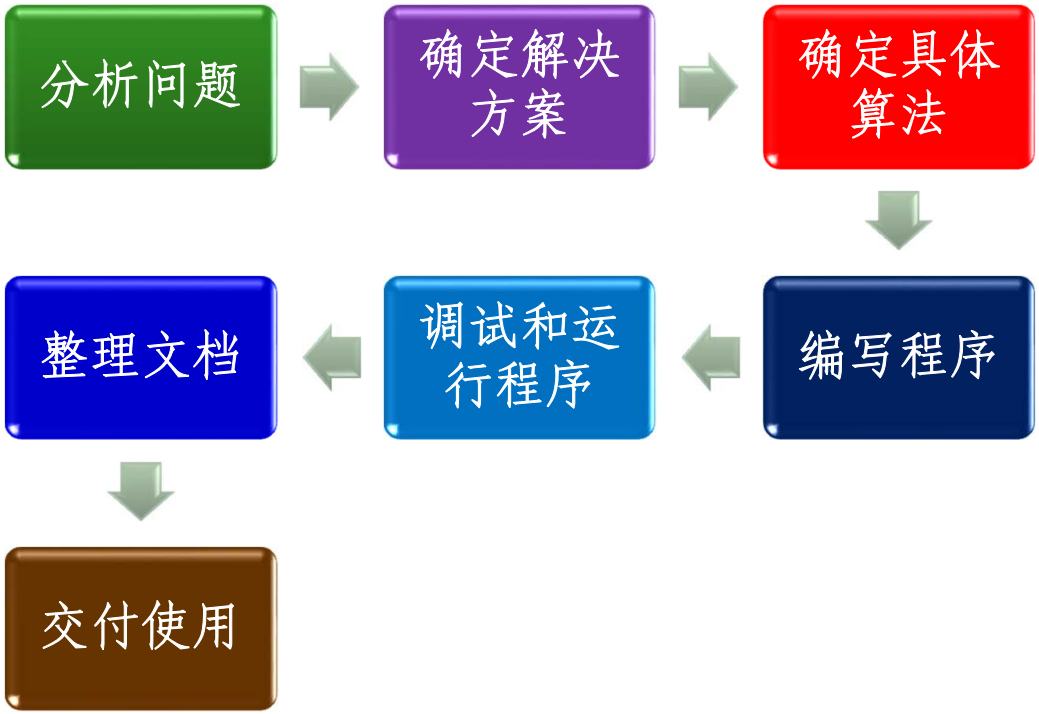
- 我们并不只是吃一只，那怎么办呢？比如说吃多只汤包或是说吃饱为止呢？
  - 策略一：控制数量
    - 假如规定吃8只：这就引入了循环结构和条件判断，达到8只，就终止循环，否则再执行上面吃一只汤包的步骤；
  - 策略二：是否吃饱
    - 将是否吃饱做为条件判断语句，若未吃饱，则跳转吃汤包的语句继续执行，一旦判断吃饱了，则跳出循环。

吃蟹黄汤包

- 我们并不只是吃一只，那怎么办呢？比如说吃多只汤包或是说吃饱为止呢？
  - 策略三：对于不同年龄段，对汤包的需求不同，采用多分支情况进行分类定量控制

计算机求解问题的步骤

- 用计算机解决一个实际应用问题时的整个处理过程称为程序设计



计算机算法概述

- 要利用计算机处理问题，光学习语言的语法规则还不够，最重要的是要学会针对各类型的问题，拟定出有效的解题方法和步骤。解题方法和步骤就是算法。
- 算法
  - 为了解决一个问题而采取的有限步骤。
- 计算机算法
  - 如何使计算机一步一步地工作的具体过程。

计算机算法概述

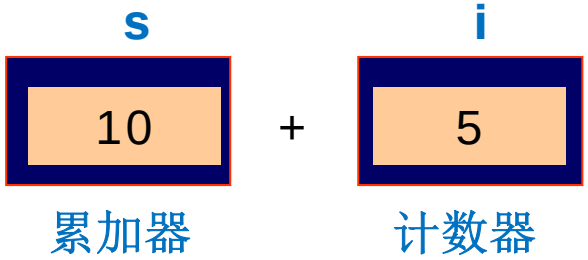
- 例：考虑 $s=\sum_{i=1}^{10} i$ 的算法
  - 算法一：直接表达

$$s = 1 + 2 + \cdots + 10$$

当项数较多时该算法不适用

计算机算法概述

- 算法二：迭代法（累加求和法）
  - 算法步骤：
    - ① 使s=0
    - ② 使i=1
    - ③ s+i→s
    - ④ i+1→i
    - ⑤ 若i≤10转③，否则转⑥
    - ⑥ 输出s



多个无规律的数求和，
怎样设计算法？

该算法通用，是好算法

如何表示算法

- 算法需要有统一的表示方法
  - 常见的表示方法有:
    - 自然语言
    - 流程图
    - 结构化程序图
    - N-S流程图

如何表示算法

- 自然语言表示

对于计算 s=1+2+3+4+5+6+7+8+9+10

用自然语言表示为:

① 使s=0　　（s为累加器）

② 使i=1　　（i为计数器）

③ s+i→s　　（累加求和公式）

④ i+1→i　　（计数器加1）

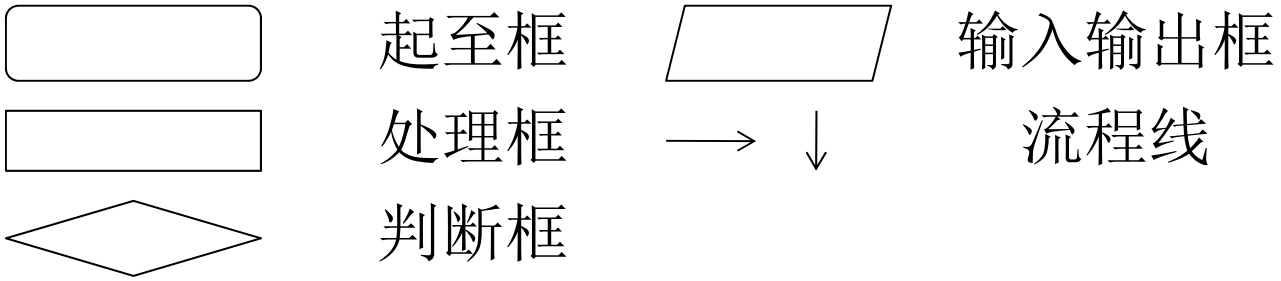
⑤ 若i≤10转③,否则转⑥

⑥ 输出s的值

特点：通俗易懂；文字冗长、含义不大严格。

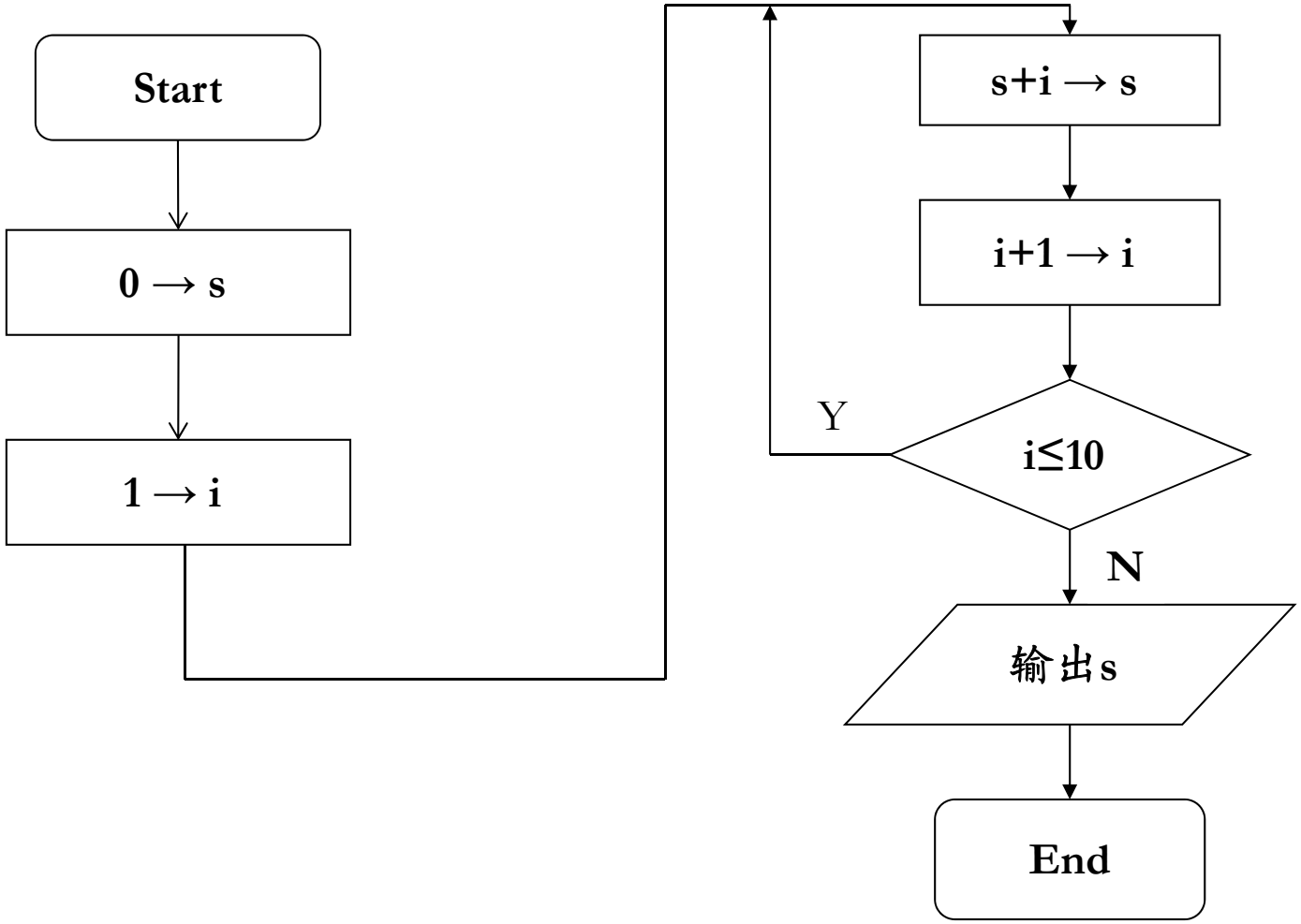
如何表示算法

- 流程图表示
  - 用流程图符号表示算法
  - 常用的流程图符号有:



对于计算 s=1+2+3+4+5+6+7+8+9+10

用流程图表示为：



直观形象，易于理解，次序清楚

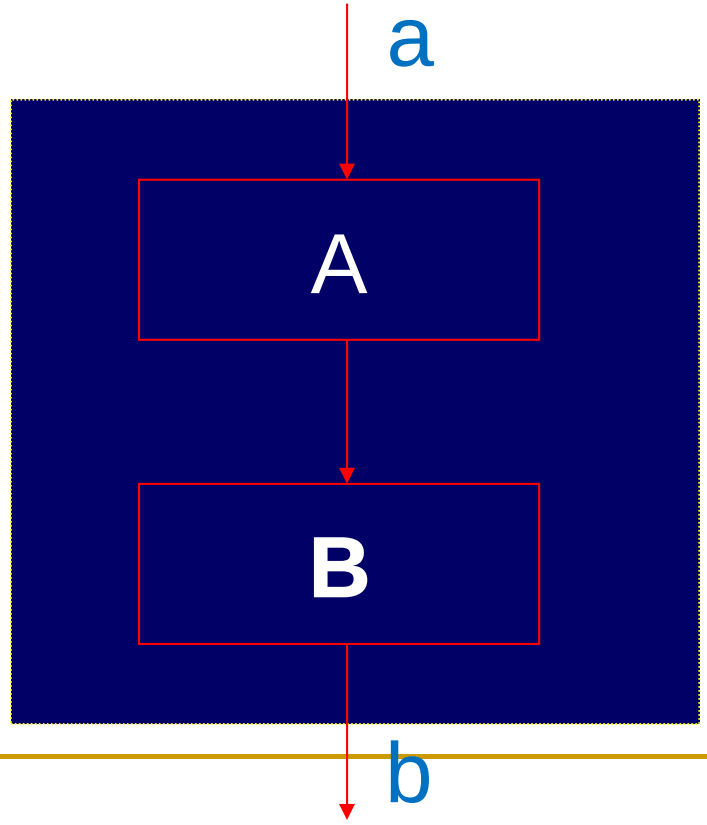
如何表示算法

- 流程图包含以下部分
  - 表示相应操作的框
  - 带有箭头的流程线
  - 框内外必要的问题说明

流程线不要忘记画箭头

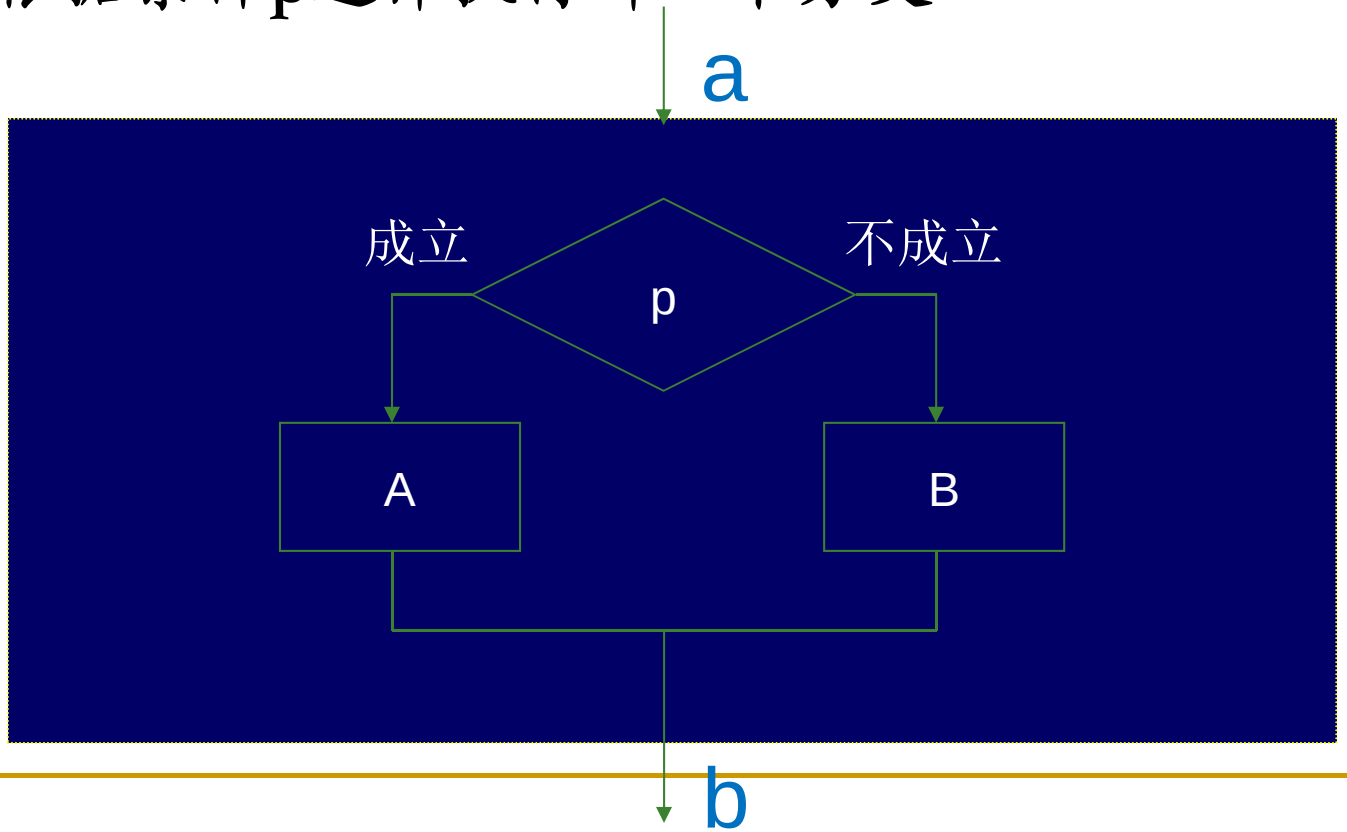
如何表示算法

- 结构化表示——顺序结构
  - 按固定顺序（从上到下或从左到右）执行的结构



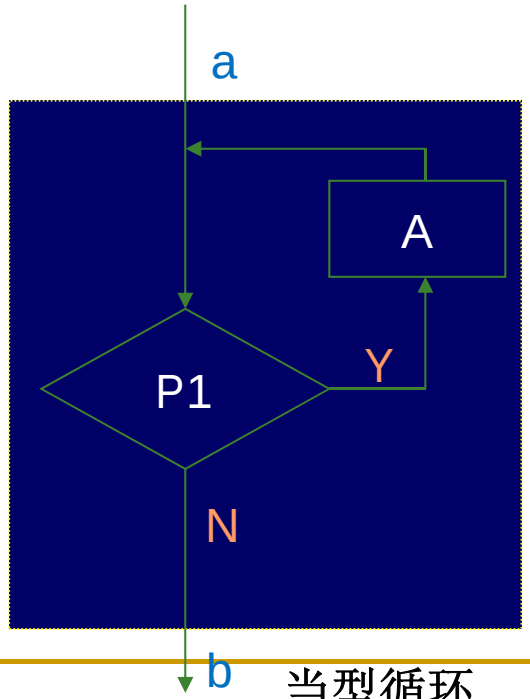
如何表示算法

- 结构化表示——选择结构
  - 根据条件p选择执行哪一个分支

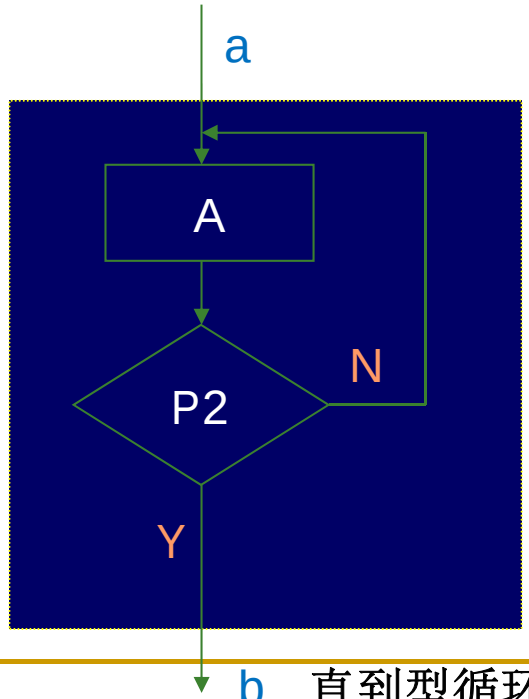


如何表示算法

- 结构化表示——循环结构
  - 重复执行某些操作的结构：
    - 当型(while)循环和直到型(until)循环



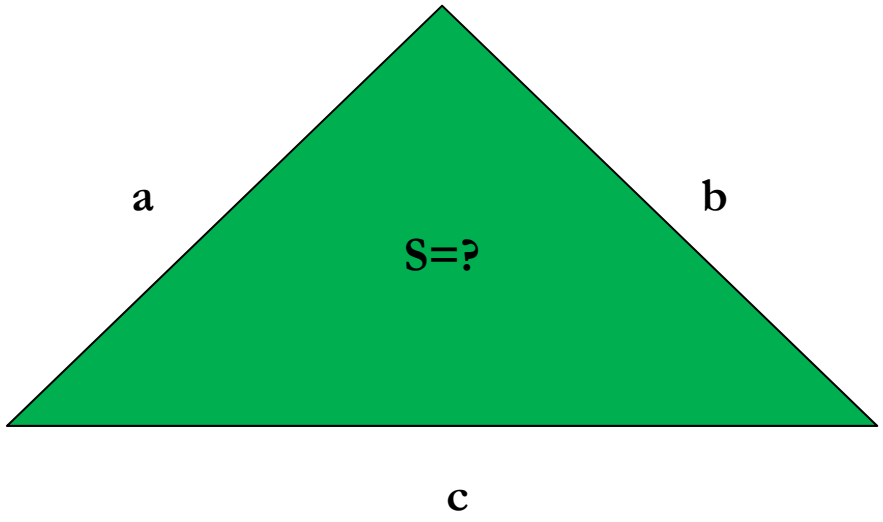
当型循环



直到型循环

顺序结构程序设计

- 输入三角形三边的长度，并计算其面积，然后输出到屏幕上



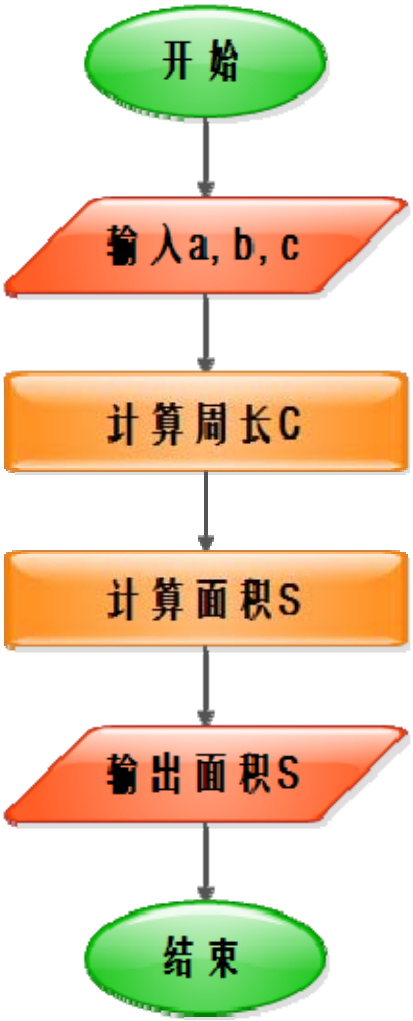
顺序结构程序设计

- 算法设计
  - 假设：三个边长a，b，c能构成三角形

$$C = \frac{a+b+c}{2}$$
$$S = \sqrt{C(C-a)(C-b)(C-c)}$$

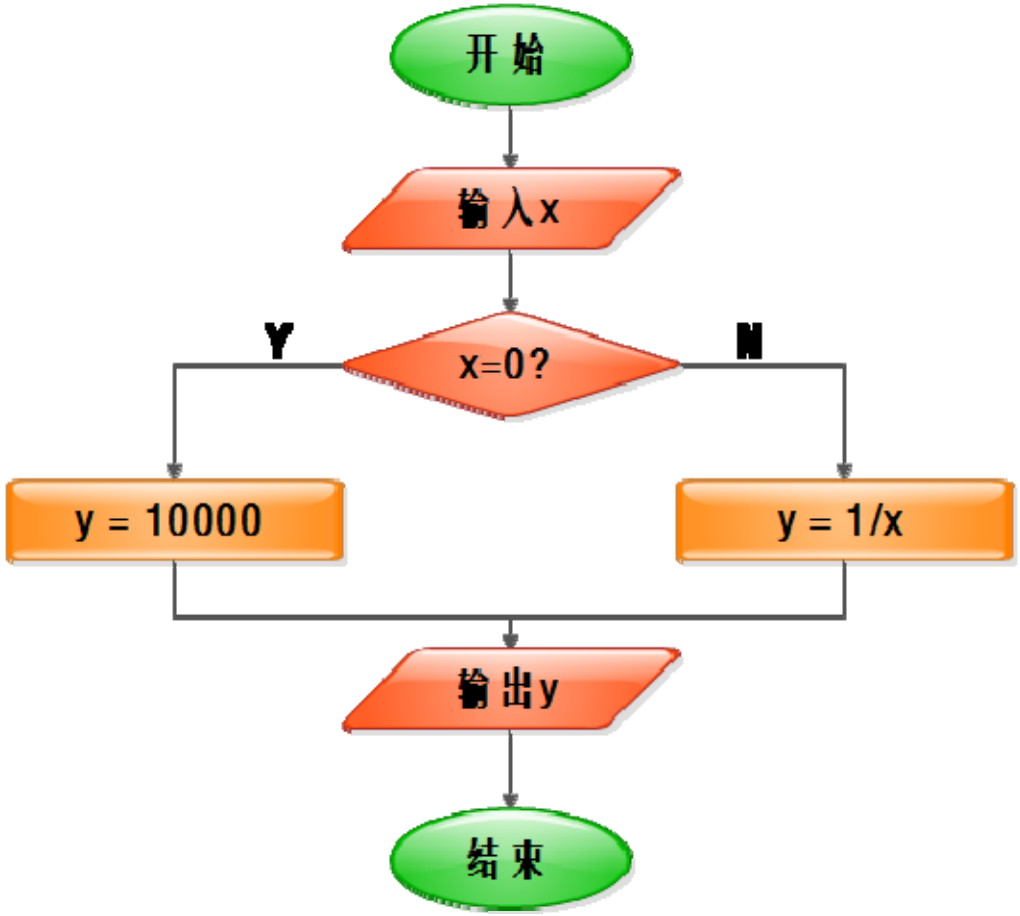
顺序结构程序设计

- 流程图



例：计算 $y=\begin{cases}\frac{1}{x}, & \text{if } x\neq 0\\ 10000, & \text{otherwise}\end{cases}$ 的算法流程图

选择结构



循环结构程序设计

- 求和

小结

- 了解算法设计的重要性
- 了解算法的基本表示方法