

基本算法语句 (1)

高一数学备课组 房

军
南京市东山外国语学校

问题情境

已知我班某学生上学期期末考试语文、数学和英语学科成绩分别为127、122、108，试设计适当的算法求出这名学生三科的平均分。

算法：

S1 $a \leftarrow 127$

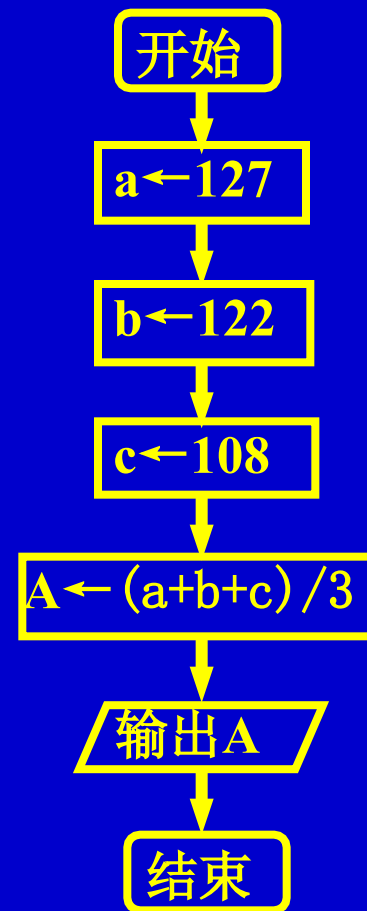
S2 $b \leftarrow 122$

S3 $c \leftarrow 108$

S4 $A \leftarrow (a+b+c)/3$

S5 输出A

流程图：



怎样将以上算法转换成计算机能理解的语言呢？



你知道计算机是怎样工作的吗？

- 计算机完成任何一项任务都需要算法。
- 算法要用计算机能够理解的程序设计语言（programming language）翻译成计算机程序。
- 程序设计语言有很多种。如BASIC, Foxbase, C语言, C++, J++, VB等。
- 为了实现算法中的三种基本的逻辑结构，各种程序设计语言中都包含下列基本的算法语句：

A decorative vertical bar on the left side of the slide, featuring a gradient from blue at the bottom to yellow at the top, with a red arrow pointing right and a yellow arrow pointing up.

基本的算法语句

输入语句

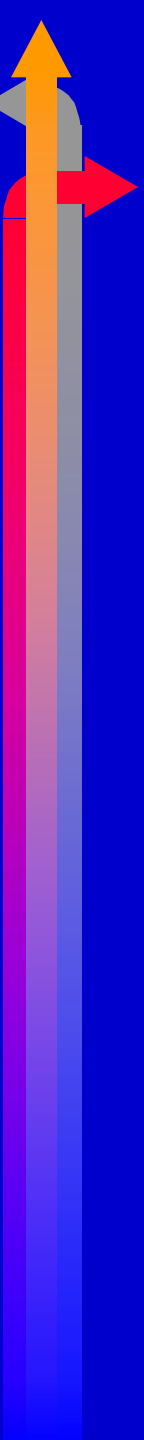
输出语句

赋值语句

条件语句

循环语句

本节主要通过**伪代码**学习基本的算法语句。



伪代码是介于自然语言和计算机语言之间的文字和符号,是表达算法的简单而实用的好方法.

建构数学

赋值语句的定义：

将表达式所代表的值赋给变量的语句。

赋值语句的一般格式：

变量名 \leftarrow 表达式，“ \leftarrow ”叫做赋值号。

例如：

“ $x\leftarrow y$ ”表示将 y 的值赋给 x ，其中 x 是一个变量， y 是一个与 x 同类型的变量或表达式。



说明：

- ①赋值语句中的赋值号“←”的左右两边不能对换，它将赋值号右边的表达式的值赋给赋值号左边的变量；
- ②赋值语句左边只能是变量名字，而不是表达式，右边表达式可以是一个数据、常量或表达式；
- ③对于一个变量可以多次赋值。



例1: 写出求 $x=23$ 时多项式 $7x^3+3x^2-5x+11$ 的值的算法.

算法1: $x \leftarrow 23;$

$$p \leftarrow 7x^3 + 3x^2 - 5x + 11$$

算法2: $x \leftarrow 23;$

$$p \leftarrow ((7x + 3)x - 5)x + 11;$$



两种算法比较:

算法1要做6次乘法，算法2只需做3次乘法。由此可见，算法的好坏会影响运算速度。

算法2称为**秦九韶算法**。

其**算法特点**是：通过一次式的反复计算，逐步得出高次多项式的值；对于一个 n 次多项式，只要做 n 次乘法和 n 次加法。

问题情境

例2:今有鸡兔同笼，上有三十五头，下有九十四足，问鸡兔各有多少？

用方程的思想不难解决这一问题。

设有 x 只鸡， y 只兔，则

$$\begin{cases} x + y = 35 \\ 2x + 4y = 94 \end{cases}$$

设计一个解二元一次方程组的通用算法

设二元一次方程组为：

$$\begin{cases} a_1x + b_1y = c_1 \\ a_2x + b_2y = c_2 \end{cases} \quad (a_1b_2 - a_2b_1 \neq 0)$$

用消元法解得

$$\begin{cases} x = \frac{b_2c_1 - b_1c_2}{a_1b_2 - a_2b_1} \\ y = \frac{a_1c_2 - a_2c_1}{a_1b_2 - a_2b_1} \end{cases}$$

建构数学

因此只要输入相应的未知数的系数和常数项，就能计算出方程组的解，即可以输出 x , y 的值.

我们用输入语句 (input statement)

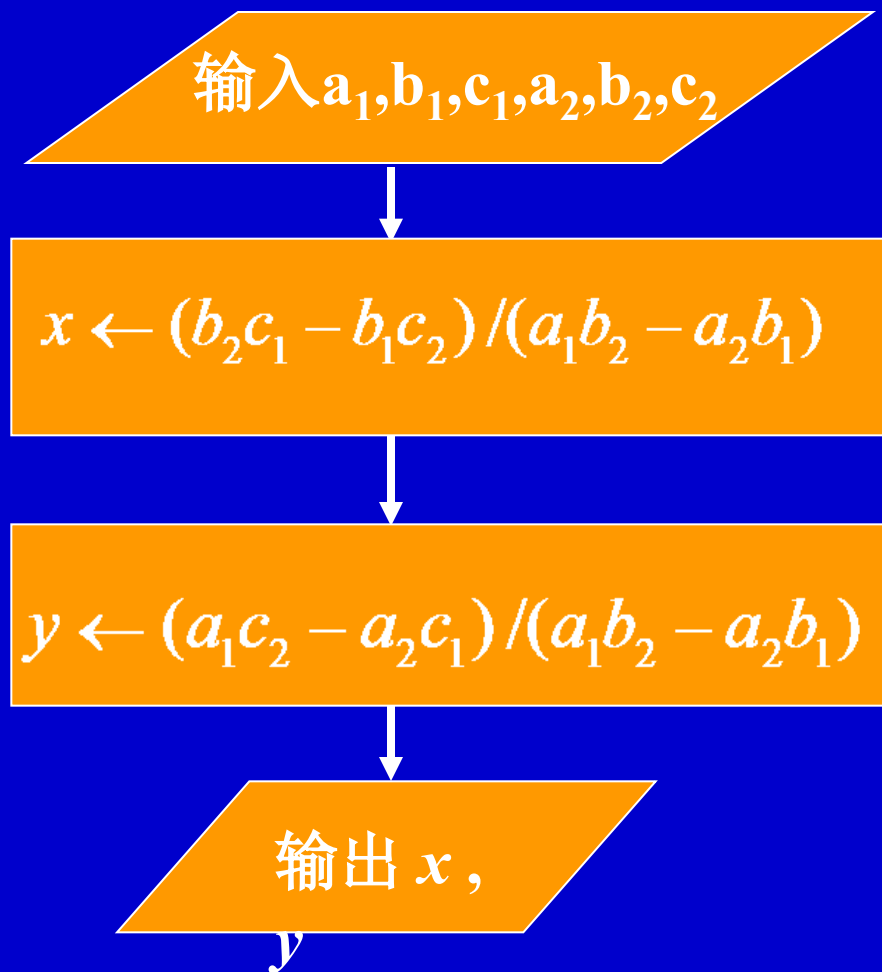
“Read a, b ”表示输入的数据依次送给 a, b

用输出语句 (output statement) “Print

x ”表示输出运算结果，这样上述解二元一次方程组的算法的流程图与相应的伪代码就可以表示为：

解二元一次方程组的通用算法

★ 流程图



★ 伪代码

Read

$a_1, b_1, c_1, a_2, b_2, c_2$

$x \leftarrow (b_2c_1 - b_1c_2) / (a_1b_2 - a_2b_1)$

$y \leftarrow (a_1c_2 - a_2c_1) / (a_1b_2 - a_2b_1)$

Print x, y

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/738066037010006123>