教育部高职高专规划教材
Jiaoyubu Gaozhi Gaozhuan Guihua Jiaocai

高等数学训练教程

侯风波 主编

高等教育出版社
HIGHER EDUCATION PRESS

教育部高职高专规划教材

高等数学训练教程

侯风波　主编
张学奇　孟庆才　焦万堂　徐志义　编

高等教育出版社

内容提要

本书是教育部高职高专规划教材，是根据教育部最新制定的《高职高专教育高等数学课程教学基本要求》编写的《高等数学》的配套教材。内容包括 Mathematica 中的程序设计、函数训练、极限训练、导数训练、导数应用训练、一元函数积分学训练、常微分方程训练、向量与空间解析几何训练、多元函数微分学训练、多元函数积分学训练及级数训练，共十一讲。

本书以高等数学的基本概念与基本方法为训练重点，特别关注高等数学的思想方法及用高等数学解决实际问题的能力的训练。每讲包含基本训练、典型习题分析及解答、自测题、自测题答案四项内容。在基本训练中，通过精选例题与习题对基本概念与基本方法进行训练；在典型习题分析及解答中，对主教材中的大部分习题作了详尽的分析解答。

本书可作为高职、高专、成人高校高等数学教材的配套教材。

图书在版编目（CIP）数据

高等数学训练教程/侯风波主编．—北京：高等教育出版社，2001

教育部高职高专规划教材

ISBN 7-04-009310-3

Ⅰ．高… Ⅱ．侯… Ⅲ．高等数学—高等学校：技术学校—教材 Ⅳ．O13

中国版本图书馆 CIP 数据核字（2001）第 029019 号

高等数学训练教程

侯风波 主编

出版发行	高等教育出版社		
社　址	北京市东城区沙滩后街 55 号	邮政编码	100009
电　话	010-64054588	传　真	010-64014048
网　址	http://www.hep.edu.cn http://www.hep.com.cn		
经　销	新华书店北京发行所		
印　刷	北京印刷一厂		
开　本	787×1092 1/16	版　次	2001 年 7 月第 1 版
印　张	13.25	印　次	2001 年 7 月第 1 次印刷
字　数	320 000	定　价	11.70 元

前　言

本书是教育部高职高专规划教材，是根据教育部最新制定的《高职高专教育高等数学课程教学基本要求》编写的教育部高职高专规划教材《高等数学》的配套教材。本书内容包括Mathematica程序设计、函数训练、极限训练、导数训练、导数应用训练、一元函数积分学训练、常微分方程训练、向量与空间解析几何训练、多元函数微分学训练、多元函数积分学训练、级数训练，共十一讲。本书以高等数学的基本概念与基本方法为训练重点，并特别关注高等数学的思想方法及用高等数学解决实际问题的能力的训练。每讲训练含基本训练、典型习题分析解答、自测题、自测题答案四项内容。本书具有如下特点：

1. 严格按照《高职高专教育高等数学课程教学基本要求》，遵照高职高专高等数学的教学规律，以掌握概念、强化应用为重点，加强学生数学文化素质的培养。

2. 大部分内容采用一例（一个例题）、一练（一个练习题）、一说明（一个解题说明）的格式编写，有助于学生系统掌握有关知识。

3. 为了培养学生应用数学解决实际问题的意识、兴趣和能力，本书编入了较多的应用实例。

4. 对重要概念与重要方法不但进行了深入浅出的分析，而且还通过典型习题分析解答的形式，对主教材中的大部分习题做了较详尽的分析解答，以便于学生自学。

5. 对Mathematica程序设计做了简要介绍，以配合高等数学主教材的有关内容，进一步培养学生借助于计算机与数学软件包求解数学模型的能力。

6. 通过一题多解等形式的训练，培养学生分析问题、解决问题所必需的抽象思维能力、联想能力及逻辑推理能力。

本书可单独使用，也可作为高职高专高等数学教材的配套教材。建议教师结合习题课教学对学生使用本书进行必要的指导，切实提高高职高专学生的数学素养。

参加本书编写的有侯风波（承德石油高等专科学校）、张学奇（承德石油高等专科学校）、孟庆才（河北工程技术高等专科学校）、焦万堂（郑州工业高等专科学校）、徐志义（本溪冶金高等专科学校），全书的框架结构安排、统稿、定稿由侯风波承担。

教育部高等学校数学与力学教学指导委员会成员、北京航空航天大学教授李心灿和北方工业大学数学学科主任、副教授宋瑞霞承担了本教材的审稿工作。他们认真审阅了本书的全部原稿，并提出了许多有价值的意见。本书的编写和出版，得到了高等教育出版社有关领导及该社高职高专编辑室张思華主任的重视，并给予了大力支持与帮助。责任编辑李艳馥和杨树东两位同志为本书的编辑出版付出了辛勤的劳动，并提出了许多好的建议。在此一并致以最诚挚的谢意。

由于我们水平所限，书中若有不当之处，恳请同仁和读者给予批评指正。

编　者

2001年2月

第一讲　Mathematica 中的程序设计

为使读者能够利用 Mathematica 系统求解较复杂的数学问题，本讲给大家介绍一下 Mathematica 中的程序设计.

何谓程序？通常由 Mathematica 的一组命令（或语句）结合在一起，共同完成一个或多个问题的求解，这一组命令叫一个程序. 见下例.

例 1　设 $f(x)=x^2, g(x)=2^x$, 求 $f(g(x))$ 及 $g(f(x))$.

解　为让 Mathematica 给出该问题的运算结果，必须先定义 $f(x)$ 及 $g(x)$，其程序如下：

```
In[1]:= f[x_]:= x^2
In[2]:= g[x_]:= 2^x
In[3]:= f[g[x]]
In[4]:= g[f[x]]
```

Out[3] = 2^{2x}

Out[4] = 2^{x^2}

上面的 4 个输入语句就构成了求 $f(g(x))$ 及 $g(f(x))$ 的一个程序. 实际问题中，需要求解的数学模型比该例复杂得多，其 Mathematica 系统中的运算程序也要复杂得多. 特别是，当数据比较大时，灵活运用建表函数 Table，往往能收到事半功倍的效果. 另外，几乎所有综合问题都离不开循环及条件判断等操作，因此，本讲要重点研究 Mathematica 系统的循环与条件控制语句的使用方法.

一、Table 函数的灵活运用

Table 为一建表函数，其格式如下：

Table [通项公式, {循环范围}, {循环范围}, …], 其中 {循环范围} 的描述形式由表 1.1 给出：

表 1.1

{循环范围} 描述形式	意　　义
{i, min, max, step}	i 从 min 到 max，以步长 step 增加
{i, min, max}	同上，步长 step 为 1 时，可省略不写
{i, max}	i 从 1 到 max，初值为 min = 1 时，可省略
{max}	重复 max 次
{i, imin, imax}, {j, jmin, jmax}	i 从 imin 到 imax，并对 i 的每一个值，j 从 jmin 到 jmax，步长为 1

建表函数的常用形式由表 1.2 给出：

表 1.2

建表函数	意　　义
Table [f [i], {i, min, max, step}]	给出 f [i] 的数值表，i 从 min 变到 max，以 step 为步长
Table [f [i], {i, min, max}]	给出 f [i] 的数值表，i 从 min 变到 max，步长为 1
Table [f [i], {i, imin, imax}, {j, jmin, jmax}, …]	生成一个多维表，i，j，…的步长皆为 1
Table [Table [f [i, j], {i, imin, imax}], {j, jmin, jmax}]	以嵌套方式生成一个二维表，i，j 的步长为 1

为了灵活运用表进行计算，还必须对表的元素能准确操作.

若 b 为一维表，则b[[i]]表示 b 的第 i 个元素；若 b 为二维以上的表，则 b[[i]]表示 b 的第 i 个子表；若 b 为二维表，则其第 i 个子表中的第 j 个元素用 b[[i,j]]表示. 如果 s = {{1,2,3},{2,3,1},{3,2,1}}，则 s[[2,3]]表示 s 中的第 2 个子表{2,3,1}的第 3 个元素 1，即 s[[2,3]] = 1；若 b 为三维表，则 b[[[1,2,3]]]表示 b 的第 1 个分表中第 2 个子表的第 2 个元素，如：

b = {{{1, 2}, {2, 3}}, {{4, 5}, {6, 8}}}

b [[1, 2, 1]] = 2

b [[2, 2, 2]] = 8

例 2　设平面上点 $A_i(i=1,2,\cdots,10)$ 的坐标如下：$A_1(1,0)$，$A_2(1,1)$，$A_3(0,1)$，$A_4(0,2)$，$A_5(-1,2)$，$A_6(-1,0)$，$A_7(-2,0)$，$A_8(-2,-1)$，$A_9(0,-1)$，$A_{10}(1,-1)$. 求相邻两点的距离，并求出 A_1 到 A_{10} 的总路程.

解

In [1]: = a: = {{1, 0}, {1, 1}, {0, 1}, {0, 2}, {-1, 2}, {-1, 0}, {-2, 0}, {-2, -1}, {0, -1}, {1, -1}}

b = Table [Sqrt [(a [[k + 1, 1]] - a [[k, 1]]) ^2 + (a [[k + 1, 2]] - a [[k, 2]]) ^2], {k, 1, 9}]

c = Sum [b [[k]], {k, 1, 9}]

Out [2] = {1, 1, 1, 1, 2, 1, 1, 2, 1}

Out [3] = 11

在上面的计算过程中，首先利用了平面直角坐标系中两点间距离公式，求出了相邻两点 A_k (a [[k, 1]], a [[k, 2]]) 与 A_{k+1} (a [[k + 1, 1]], a [[k + 1, 2]]) 间的距离，由表 b 给出，随后用 Sum 函数对表 b 的 11 个元素进行求和，得到了从 A_1 到 A_{10}的总路程为 11.

练习　求以例 2 中的 10 个点为顶点的多边形的周长与面积.

二、条件控制语句

Mathematica 系统提供了 3 种描述条件分支的结构，这些条件语句在编程中经常用到.

1. If 语句

If 语句的结构与一般程序设计语言中的 If 结构类似. 由于 Mathematica 中逻辑表达式的值有 3 个：真 (True)，假 (False) 和“非真非假”，因此 If 语句有下面三种格式：

If [逻辑表达式，表达式 1]，其意义为：当逻辑表达式的值为真（True）时，计算表达式 1，且表达式 1 的值是整个 If 语句的值.

If [逻辑表达式，表达式 1，表达式 2]，其意义为：当逻辑表达式的值为真（True）时，计算表达式 1，并将表达式 1 的值作为整个 If 语句的值；当逻辑表达式的值为假（False）时，计算表达式 2，并将表达式 2 的值作为该 If 语句的值.

If [逻辑表达式，表达式 1，表达式 2，表达式 3]，其意义为：当逻辑表达式的值为真（True）时，计算表达式 1，并将表达式 1 的值作为该 If 语句的值；当逻辑表达式的值为假（False）时，计算表达式 2，并将表达式 2 的值作为该 If 语句的值；当逻辑表达式的值非真（True）非假（False）（即真假不确定的情况）时，计算表达式 3，并将所计算的值作为该 If 语句的值.

例 3 定义函数 $f(x)=\begin{cases}x^2, & x<0;\\ 2x+1, & x\geqslant 0,\end{cases}$ 并求 $f(-2)$，$f(0)$ 及 $f(5)$ 的值.

解

In [1]: =f [x_]: =If [x<0, x^2, 2*x+1]

In [2]: = {f [-2], f [0], f [5]} （*在一个表中完成 3 个函数值的计算*）

Out [2] = {4, 1, 11} （*将所求的 3 个函数值在一个表中输出*）

2．Which 语句

Which 语句也是条件分支语句．该语句可以对多个条件依次进行判断，并对第一个为真的条件所对应的表达式进行求值．Which 语句的一般表达式如下：

Which [条件 1，表达式 1，条件 2，表达式 2，…，条件 n，表达式 n]，其意义为：从左到右，依次执行条件 i 并对第一个为真的条件所对应的表达式求值，该值就作为该 Which 语句的值.

例 4 定义分段函数

$$g(x)=\begin{cases}-1, & x<0;\\ x^2, & 0\leqslant x<1;\\ \cos x, & 1\leqslant x<2;\\ 10x, & 2\leqslant x<10;\\ 0, & \text{其他},\end{cases}$$

并求 $g(-10)$, $g\left(\frac{1}{2}\right)$, $g(5)$, $g(10)$.

解

In[1]: = g[x_]: = Which[x < 0, -1,x > = 0&&x < 1,x^2,x > = 1&&x < 2,Cos[x], x > = 2&&x < 10,10 * x,True,0]

In[2]: = {g[-10],g[1/2],g[5],g[10]}

Out[2] = $\{-1,\frac{1}{4},50,0\}$

例 4 中的 Which 语句中的最后一个条件用的逻辑值 True（真），其目的是为了适应当自变量的值不满足前边 4 个条件时，即作为其他情况处理，在这里函数值为 0．通常用 True 作为 Which 语句的最后一个条件时，可用于处理其他情况.

3. Switch 语句

Switch 语句的一般形式为：

Switch [expr，模式 1，表达式 1，模式 2，表达式 2，…]，其意义是：将表达式 expr 的值与模式 1，模式 2，…的值依次进行比较，给出第一个与 expr 相匹配的模式 i 所对应的表达式 i 的值，并将此值作为 Switch 语句的值；若没有与 expr 相匹配的模式，则 Switch 语句的值为 Null.

例 5 给出下面两个语句的输出结果：

In[1]: = h[x_]: = Switch[Mod[x,4],0,x,1,x^2,2,x^3,3,x^4];

In[2]: = {h[5],h[8],h[10]}.

解 首先，Mod [x，4] 给出 x 被 4 除的余数，Switch 语句指出，当 Mod [x，4] =0（即 x 被 4 整除）时，Switch 语句的值为 x；当 Mod [x，4] =1（即 x 被 4 除余 1）时，Switch 语句的值为 x^2；当 Mod [x，4] =2（即 x 被 4 除余 2）时，Switch 语句的值为 x^3；当 Mod [x，4] =3（即 x 被 4 除余 3）时，Switch 语句的值为 x^4.

由于 Mod [5，4] =1，所以，h [5] $=5^2=25$.

由于 Mod [8，4] =0，所以，h [8] =8.

由于 Mod [10，4] =2，所以，h [10] $=10^3=1000$.

故 Out [2] = {25，8，1000}.

三、循环控制语句

Mathematica 系统有许多种语句可以描述循环. Do，While 和 For 语句为描述循环的基本语句.

Do 语句先计算循环次数，再执行循环体，常用于有确定循环次数的循环结构；在 While 和 For 语句中，先作一次条件确认后，再执行一次循环体的表达式，但在用逻辑表达式构造条件时，要注意避免空循环和死循环.

1. Do 语句

Do 语句的一般格式为：

Do [循环体，{循环范围}]，其意义为：对循环范围内的每一个循环值执行一次循环体，循环体由一个或多个表达式组成，表达式之间用分号分离.

Do 语句的常用形式如下：

Do [expr，{i，i0，i1，step}]，其意义为：对循环变量 i 从 i0 到 i1，每次 i 增加 step，计算表达式 expr.

Do [expr，{i，i1}]，其意义为：i 从 1 到 i1，步长为 1，计算表达式 expr.

Do [expr，{n}]，其意义为：对表达式 expr 计算 n 次.

Do [expr，{i，i0，i1，istep}，{j，j0，j1，is}]，其意义为：i 从 i0 到 i1，每次增加 istep，j 从 j0 到 j1，每次增加 js，计算表达式 expr.

例 6 分别求 100 以内所有偶数之和及 20 以内所有偶数之积，即分别求出下面两个表达式的值：

(1) $2+4+6+\cdots+100$;

(2) $2\cdot4\cdot6\cdot\cdots\cdot20$.

解

In [1]: = s = 0; Do [s = s + i, {i, 2, 100, 2}]; s

In [2]: = t = 1; Do [t = t * i, {i, 2, 20, 2}]; t

Out [1] = 2550

Out [2] = 3715891200

在该例中，In [1] 与 In [2] 分别由 3 个语句组成，若将这 3 个语句写成 3 行，运算结果不变. 在 In [1] 中，将 100 以内的偶数之和用变量 s 表示，在循环累加之前，先将 s 清零，这是计算连加的常用技巧；在 In [2] 中，用变量 t 表示 20 以内所有偶数之积，在循环相乘之前，先将 t 赋值 1，即 t = 1，这也是计算连乘积的常用技巧.

2. While 语句

While 语句的一般形式为：

While [条件，循环体]，其意义为：当条件为真（True）时，对循环体表达式求值，然后重复条件判断的循环体求值过程，直到条件非真时停止；当条件的值非真非假时，While 语句不做任何工作. While 语句的循环次数完全取决于其条件的设置，条件设置不当，可能一次不执行，也可能无限地执行下去.

例 7 输出介于 90 与 100 之间的 4 个偶数.

解

In [1]: = n = 100; While [(n = n - 2) > 90, Print [n]]

在运行 In [1] 的过程中，在屏幕上输出 98，96，94，92 这 4 个数. 当 While 语句执行完后，n = 90，请想一想，为什么？另外，在 While 语句中，若将 Print [n] 换成 n，运行结果会怎样？借助于“帮助”来研究此问题.

3. For 语句

For 语句的一般形式为：

For [初始值，条件，修正循环变量，循环体].

For 语句的计算步骤为：

(1) 先执行初始条件（循环变量的初始值及循环体中有关变量的初值）；

(2) 再对循环条件进行判定，若循环条件为真，则执行一次循环体（如循环条件不真，则结束循环计算，跳到下一个语句）；

(3) 修正循环变量的值后再转到 (2).

下面将修正循环变量的操作方式说明如下：

k + + : k 的值增加 1；

+ + k: k 先增加 1；

k - - : k 的值减少 1；

- - k: k 先减少 1.

例 8 利用 For 循环求 10!.

解

In [1]: = For [i = 1; t = 1, i ≤ 10, i + + , t = t * i]; t

Out [1] = 3628800

注意，在 In [1] 中，之所以后面加上一个变量 t 是为了将 t 的计算结果在 Out [1] 中输出，这是因为在 Mathematica 中每个循环结构整体的值都是 Null. 另外，循环变量的修正 i++ 相当于 i=i+1，实际上，在 For 循环中可以通过赋值的形式来对循环变量进行修正，见下例.

例 9 利用 For 循环计算 10 以内的所有奇数之积.

解

In [1]: = For [i=1; t=1, i<10, i=i+2, t=t*i], t

Out [1] = 945

练习 编程计算 $\sum_{i=1}^{100} i!$.

四、过程程序与函数程序

在用 Mathematica 求解某一数学问题时，往往需要一系列表达式（或语句）才能完成，将这些表达式用分号隔开构成一个表达式序列.

一个表达式序列也称为一个过程. 在运行时，过程中的表达式依次求值，像 Mathematica 的表达式一样，过程的定义和调用轻便灵活（如，可以把一个过程定义给一个变量或一个函数），在一个输入行中就可以放一个过程，调用一个过程就像调用一个函数那样简单.

在过程中，经常要用工作变量保存计算的中间结果，这类变量称为局部变量. 局部变量在过程中定义和使用，在过程之外不再保留过程计算之中局部变量的值. 使用局部变量有利于过程的模块化.

在 Mathematica 中，可用函数 Block 或 Module（模块）来实现将局部变量说明与表达式序列组成一个程序块. 这里仅介绍 Module（模块）的使用方法.

1. Module 程序块

Module 的格式为：

Module [{局部变量表}, 表达式序列], 其中 {局部变量表} 中可以说明零个或多个局部变量，说明变量时，只列出局部变量名，不需对变量类型进行说明，局部变量名之间用逗号分隔，并可在说明时赋予初值；表达式序列可以由一个或多个表达式组成，表达式之间要用分号分隔，并以最后一个表达式的值作为 Module 语句的值. 也就是说，当表达式为复合表达式时，最后一个表达式的值为复合表达式的值.

例 10 在一个 Module 块中，将多项式 $(1+x)^2$ 与 $(1-2x)^2$ 的差分解因式，并将最终结果赋予变量 u.

解

In [1]: = u = Module [{s}, s = (1+x) ^2 - (1-2*x) ^2; Factor [s]]

Out [1] = -3 (-2+x) x

请上机验证一下，当把 x 也放入局部变量表中，该问题的运行结果怎样?

如果一个过程没有局部变量说明，则可省略 Module 的框架. 在函数定义中，如果要用一串命令完成计算，需要将这一串命令用圆括弧括起来，并以最后一个表达式的值作为函数值.

如

```
In [1]: =f [x]: = (z=6*x; y=10*x)
In [2]: =f [10]
Out [2] =100
```

请进行上机实验，在上面的程序中，若在 In [1] 中去掉圆括弧，该程序中的运算结果又怎样？

2．函数程序

从程序设计的角度看，编制程序时总要精心地设置一些局部变量、循环控制结构和子程序等．过程程序的优点是有利于充分展示程序设计者的个人才能．

在 Mathematica 符号计算系统中，编制程序以函数程序为主．编程的重点放在分析要解决的问题上，编制程序以调用和组合相应的系统函数为主．如果找不到解答问题的系统函数，则需编制函数程序．

例 11 设 $a_i=3i^2+1$，$i=1, 2, 3, \cdots, 20$，用多种方法计算 $\sum\limits_{i=1}^{20} a_i$.

解

```
In [1]: =b=Table [3*i^2+1, {i, 1, 20}]; s=Apply [Plus, b]
In [2]: =Sum [3*i^2+1, {i, 1, 20}]
In [3]: =s=0; Do [s=s+3*i^2+1, {i, 1, 20}]; s
In [4]: =s=0; i=21; While [(i=i-1) >=1, s=s+3*i^2+1]; s
In [5]: =For [i=1; s=0, i<=20, i++, s=s+3*i^2+1]; s
```

上面五种算法结构均给出 $\sum\limits_{i=1}^{20} a_i=8\,630$，请上机检验．

在 In [1] 中，Plus 表示加法运算，Apply [f, expr] 表示将函数 f 作用于 expr，这里 Apply [Plus, b] 表示将 b 的各元素加到一起．语句 In [1] 与 In [2] 属于函数程序，In [3]，In [4]，In [5] 属于过程程序．

值得指出的是，在 Mathematica 中，也需要对循环结构进行退出循环，结束本层循环，转向下层循环等操作．下面提供与此有关的几个函数，需要时，借助于“帮助”再予以熟悉．

Return [expr]，其意义是：退出函数中的所有过程和循环，返回 expr 的值．

Break [ ]，其意义是：结束本层循环，并以 Null 为该循环结构的值．

Continue [ ]，其意义是：转向本层 For（或 While）结构中的下一层循环．

练习 编程计算 $\sum\limits_{i=1}^{10}\left(\prod\limits_{\substack{j=1\\ j\neq i}}^{i}\frac{ij}{i-j}\right)\cdot i$，其中 $\prod\limits_{k=1}^{10} f(k)$ 表示 k 从 1 到 10 取值时，$f(k)$ 的 10 个值的连乘积．

五、Mathematica 中的常见命令

1．数值求值命令 N

N [expr]　求 expr 的近似值，有效位数由系统内定.

N [expr, n]　求 expr 具有 n 位有效数字的近似值.

2. 变量替换规则

表达式/.x－＞a　将表达式中的变量 x 用 a 代替.

3. 方程求解

Solve [eqn, var]　指定未知量为 var，求多项式方程或方程组 eqn 的解.

NSolve [eqn, var]　指定未知量为 var，求多项式方程或方程组 eqn 的近似解，有效位数由系统内定.

FindRoot [eqn, {x, x0}]　求方程 eqn 在 x0 附近的一个根.

4. 求极限

Limit [expr, x－＞x0]　求表达式当 $x \to x0$ 时的极限.

Limit [expr, x－＞x0, Direction－＞1]　求表达式 expr 当 $x \to x0$ 时的左极限.

Limit [expr, x－＞x0, Direction－＞－1]　求表达式 expr 当 $x \to x0$ 时的右极限.

5. 求导数

f' [x]　求一元函数 f (x) 的一阶导数.

D [f, x]　求函数 f 的导数$\frac{df}{dx}$或偏导数$\frac{\partial f}{\partial x}$.

D [f, {x, n}]　求函数 f 的 n 阶导数$\frac{d^n t}{dx^n}$或 f 关于 x 的 n 阶偏导数$\frac{\partial^n f}{\partial x^n}$.

D [f, x1, x2, …]　求多重偏导数$\frac{\partial}{\partial x1}\frac{\partial}{\partial x2}\cdots f$.

Dt [f]　求全微分 df.

Dt [f, x]　求全导数$\frac{df}{dx}$.

6. 积分

Integrate [f, x]　求不定积分$\int f dx$.

Integrate [f, {x, a, b}]　求定积分$\int_a^b f dx$.

Integrate [f, {x, a, b}, {y, c, d}]　求累次积分$\int_a^b dx \int_c^d f dy$.

NIntegrate [f, {x, a, b}]　求定积分$\int_a^b f dx$的近似值.

7. 解微分方程

DSolve [eqns, y, x]　求常微分方程 eqns 的纯函数形式解.

DSolve [eqns, y [x], x]　求常微分方程 eqns 的普通形式解.

DSolve [eqns, y, {x, xmin, xmax}]　求常微分方程 eqns 的数值解，自变量的取值范围为 [xmin, xmax].

DSolve [eqns, {y_1, y_2, …}, {x, xmin, xmax}]　求常微分方程 eqns 的数值解，其中未知函数 y_1, y_2, …均以 x 为自变量.

8. 数据处理

Fit [{{x1, f1}, {x2, f2}, …}, {1, x, x^2}, x]　求数据表{{x1, f1}, {x2, f2}, …}的一个二次函数.

Min [x1, x2, …]　求 x1, x2, …中的最小值.

Max [x1, x2, …]　求 x1, x2, …中的最大值.

9. 优化

FindMinimum [f, {x, x0}]　以 x0 为初值寻求函数 f 的一个局部极小值及极小值点.

ConstrainedMin [f, {inequalities}, {x, y, …}]　在不等式组 {inequalities} 所围区域内求函数 f 的最小值.

ConstrainedMax [f, {inequalities}, {x, y, …}]　在不等式组 {inequalities} 所围区域内求函数 f 的最大值.

10. 输入与输出

Print [expr1, expr2, …]　在同一行中打印表达式 expr1, expr2, …的值, 中间无空格.

Input [ ]　交互式地读入一个 Mathematica 表达式或值.

上面仅列出了 Mathematica 系统的部分函数或命令. 建议读者在学习本书过程中, 要尽量结合 Mathematica 软件包在计算机上将本书的例题与练习做一遍, 以便提高在计算机上用数学解决实际问题的能力.

第二讲　函 数 训 练

客观世界中，许多变量之间的关系都可用函数描述．本讲重点训练函数的基本概念与基本特征．通过本讲的训练，要进一步加深对函数概念的理解；强化对函数的周期性、奇偶性、单调性等几种特性及分段函数、复合函数等概念的认识；提高用函数认识实际问题的能力．

一、基 本 训 练

1．基本概念训练

近代函数模型 f: $D\to M$（如图 2-1）明确表明了函数模型的结构，它由定义域 D，对应规律 $y=f(x)$，值域 $Y=\{y\mid y=f(x),x\in D\}$ 三个要素所构成，定义域和对应规律是主导要素，其结构如图 2-2 所示，更突出了对应规律．

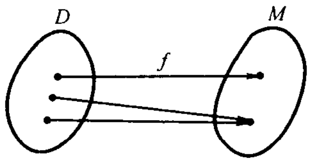

图 2-1

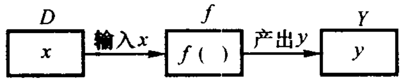

图 2-2

复合函数是刻画变量相依关系的更为复杂的一种数学模型，其结构如图 2-3 所示．若产出 u 的过程和输入 u 的过程都可实现，就可产生复合函数 $y=f[\varphi(x)]$．

例 1　下列数学结构为什么不符合函数的定义？

(1) $f=\begin{cases}-x, & x\leqslant 0;\\ 1, & x\geqslant 0;\end{cases}$

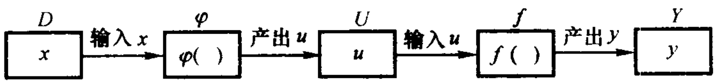

图 2-3

(2) 自变量 x 的取值范围为 $\{1,2,3\}$，f: $\begin{matrix}1 & 2\\ \downarrow & \downarrow\\ 1 & 1\end{matrix}$．

解　(1) 在 $x=0$ 处，对应的函数值不唯一，不符合函数定义．

$f=\begin{cases}-x, x\leqslant 0;\\ 1, x>0\end{cases}$ 或 $f=\begin{cases}-x, x<0;\\ 1, x\geqslant 0\end{cases}$ 才是函数，而题中所示的只是数学结构，不是函数．

(2) 数学结构 f 不符合函数定义．因为定义域中的 3 未被定义函数值，这与函数定义中

"自变量 x 在定义域中任意取定一值，因变量有唯一确定的值与之对应"不符，所以此数学结构不是函数，而

$$f:\begin{matrix} 1 & 2 & 3 \\ \downarrow & \downarrow & \downarrow \\ 1 & 1 & 1 \end{matrix}$$

就表示函数.

例 2 下列两组函数是否恒等?

(1) $y=\cos x$ 与 $y=\sqrt{1-\sin^2 x}$;

(2) $y=\ln\dfrac{1+x}{1-x}$ 与 $y=\ln(1+x)-\ln(1-x)$.

解 (1) $y=\cos x$ 与 $y=\sqrt{1-\sin^2 x}=|\cos x|$ 的定义域都是 $(-\infty,+\infty)$，但对应规律不同，故它们不恒等.

(2) $y=\ln\dfrac{1+x}{1-x}$的定义域为 $(-1,1)$，且 $y=\ln\dfrac{1+x}{1-x}=\ln(1+x)-\ln(1-x)$，而 $y=\ln(1+x)-\ln(1-x)$ 的定义域也为$(-1,1)$，说明两个函数的定义域与对应规律都相同，故它们恒等.

说明 函数定义中，最本质的是两个要素：对应规律与定义域，故考察函数模型的结构要从这两个要素着手.

> **练习** 判断下列各组函数是否恒等:
>
> (1) $y=2\ln|x|$ 与 $y=\ln x^2$; (2) $y=2\ln x$ 与 $y=\ln x^2$;
>
> (3) $y=1$ 与 $y=\sin^2 x+\cos^2 x$; (4) $y=\dfrac{x^2-1}{x-1}$ 与 $y=x+1$.

2. 基本理论训练

判断函数奇偶性的方法主要是根据定义及奇偶函数的运算性质.

(1) 两个奇函数之和是奇函数；两个偶函数之和是偶函数；奇函数与偶函数之和是非奇非偶函数；

(2) 两个奇函数的乘积是偶函数；两个偶函数的乘积是偶函数；奇函数与偶函数的乘积是奇函数.

另外，应掌握一些常见函数的奇偶性. 如，x^{2n+1} (n 为正整数)，$\sin x$，$\arctan x$ 等是奇函数；而常数函数 C，x^{2n} (n 为正整数)，$\cos x$ 等是偶函数.

例 3 证明:

(1) $f(x)=\ln(x+\sqrt{x^2+1})$ 为奇函数;

(2) $f(x)=\left(\dfrac{1}{a^x+1}-\dfrac{1}{2}\right)F(x)$(其中 $a>0$ 且 $a\neq 1$, $F(x)$ 是奇函数) 为偶函数.

证 (1) 因为 $f(-x)=\ln(-x+\sqrt{x^2+1})=\ln\dfrac{(-x+\sqrt{x^2+1})(x+\sqrt{x^2+1})}{x+\sqrt{x^2+1}}$

$$=\ln\frac{1}{x+\sqrt{x^2+1}}=-\ln(x+\sqrt{x^2+1})=-f(x),$$

所以 $f(x)$ 是奇函数.

说明

(1) 当运算式中出现 $a\pm\sqrt{b}$时，通常在运算前先对分子、分母乘以其共轭形式 $a\mp\sqrt{b}$，进行有理化简后计算；

证明 $f(x)$ 是奇函数时，往往直接计算 $f(x)+f(-x)=0$ 成立更简便.

由

$$f(x)+f(-x)=\ln(x+\sqrt{x^2+1})+\ln(-x+\sqrt{x^2+1})$$
$$=\ln(x+\sqrt{x^2+1})\cdot(-x+\sqrt{x^2+1})=\ln 1=0,$$

从而证明 $f(x)=-f(-x)$,即 $f(x)$ 为奇函数.

(2) 令 $g(x)=\dfrac{1}{a^x+1}-\dfrac{1}{2}$, 因为

$$g(-x)=\frac{1}{a^{-x}+1}-\frac{1}{2}=\frac{a^x}{1+a^x}-\frac{1}{2}=\frac{a^x+1-1}{a^x+1}-\frac{1}{2}=-\frac{1}{a^x+1}+\frac{1}{2}$$
$$=-\left(\frac{1}{a^x+1}-\frac{1}{2}\right)=-g(x),$$

所以 $g(x)$ 为奇函数. 又 $F(x)$ 是奇函数，故 $f(x)=g(x)F(x)$ 是偶函数.

例 4 判断下列函数的奇偶性：

(1) $f(x)=\dfrac{e^{-x}-1}{e^{-x}+1}$; (2) $f(x)=\begin{cases}1-x, x\leqslant 0;\\ 1+x, x>0.\end{cases}$

解 (1) $f(-x)=\dfrac{e^x-1}{e^x+1}=\dfrac{(e^x-1)e^{-x}}{(e^x+1)e^{-x}}=\dfrac{1-e^{-x}}{1+e^{-x}}=-\dfrac{e^{-x}-1}{e^{-x}+1}=-f(x)$,

所以 $f(x)$ 是奇函数.

(2) 因为

$$f(x)=\begin{cases}1-x, x\leqslant 0;\\ 1+x, x>0\end{cases}=\begin{cases}1-x, x<0;\\ 1, \quad x=0;\\ 1+x, x>0,\end{cases}$$

所以

$$f(-x)=\begin{cases}1+x, -x<0;\\ 1, \quad x=0;\\ 1-x, -x>0\end{cases}=\begin{cases}1+x, x>0;\\ 1, \quad x=0;\\ 1-x, x<0\end{cases}=\begin{cases}1-x, x\leqslant 0;\\ 1+x, x>0\end{cases}=f(x).$$

因此 $f(x)$ 是偶函数.

说明 判断分段函数的奇偶性，在分段点处存在“$\geqslant$”或“$\leqslant$”时，先分出“<”，“=”，或“>”，然后再判断.

练习 判断下列函数的奇偶性：

(1) $f(x)=x\left(\dfrac{1}{2^x+1}+\dfrac{1}{2}\right)$; (2) $f(x)=\begin{cases}1, x>0;\\ 0, x=0;\\ -1, x<0.\end{cases}$

3. 基本方法训练

例 5 求函数 $y=\sqrt{16-x^2}+\arcsin\dfrac{2x-1}{7}+\dfrac{\sqrt{\ln(x-1)}}{x(x-3)}$ 的定义域.

解 要使 y 有意义，x 应满足

$$\begin{cases}16-x^2\geqslant 0\\ \left|\dfrac{2x-1}{7}\right|\leqslant 1\\ \ln(x-1)\geqslant 0,\\ x-1>0\\ x(x-3)\neq 0\end{cases}\text{即}\begin{cases}|x|\leqslant 4\\ -3\leqslant x\leqslant 4\\ x-1\geqslant 1\\ x>1\\ x\neq 0\text{且}x\neq 3\end{cases},\quad\text{即}\begin{cases}|x|\leqslant 4\\ -3\leqslant x\leqslant 4\\ x\geqslant 2\\ x\neq 3\end{cases},$$

其交集为$[2,3)\cup(3,4]$,故函数定义域为$[2,3)\cup(3,4]$.

说明 由解析式表示的函数的定义域是使该解析表达式有意义的一切实数所构成的集合，求定义域时应注意以下几点：

(1) 若函数的表达式中含有分式，则分式的分母不能为零；

(2) 若函数的表达式中含有偶次方根，则根式下的表达式必须大于零或等于零；

(3) 若函数的表达式中含有对数，则真数必须大于零；

(4) 若函数的表达式中含有 $\arcsin\varphi(x)$ 或 $\arccos\varphi(x)$，则必须满足 $|\varphi(x)|\leqslant 1$;

(5) 分段函数的定义域是各个部分自变量的取值范围之并；

(6) 若函数式是由几个函数经过四则运算构成，其定义域是各个函数的定义域的公共部分.

例 6 设 $f(x)=\begin{cases}-x^2, & x\geqslant 0;\\ -e^x, & x<0,\end{cases}\quad\theta(x)=\ln x$,

(1) 求 $f[\theta(x)]$ 及其定义域；

(2) 可以复合成形如 $\theta[f(x)]$ 的复合函数吗?

解 (1) 因为 $\theta(x)$ 的定义域是 $(0,+\infty)$，值域为 $(-\infty,+\infty)$，而 $f(x)$ 的定义域是 $(-\infty,+\infty)$，所以 $\theta(x)$ 的值域在 $f(x)$ 的定义域内，故 $f[\theta(x)]$ 有意义，因此

$$f[\theta(x)]=\begin{cases}-\theta^2(x), & \theta(x)\geqslant 0;\\ -e^{\theta(x)}, & \theta(x)<0,\end{cases}$$

即

$$f[\theta(x)]=\begin{cases}-\ln^2 x, & x\geqslant 1;\\ -x, & 0<x<1.\end{cases}$$

故 $f[\theta(x)]$ 的定义域为$(0,+\infty)$.

说明 复合函数中，内层函数的值域与外层函数的定义域之交集必须为非空集.

(2) 由于$f(x)$的值域是$(-\infty,0)$,$\theta(x)$的定义域是$(0,+\infty)$,它们无公共部分,所以不能复合成形如 $\theta[f(x)]$ 的函数.

例 7 已知$f\left(\dfrac{1}{x}\right)=\dfrac{x}{1+x}$,(1) 求 $f(x)$;(2) 以 $f(x)$ 表示 $f(3x)$;(3) 求 $f[f(x)]$.

解 (1) $f\left(\dfrac{1}{x}\right)=\dfrac{x}{x+1}=\dfrac{1}{\dfrac{1}{x}+1}$,所以 $f(x)=\dfrac{1}{x+1}$.

(2) $f(3x)=\dfrac{1}{3x+1}$,又由 $f(x)=\dfrac{1}{x+1}$,得 $x=\dfrac{1}{f(x)}-1$,代入 $f(3x)$,有

$$f(3x)=\frac{1}{3\left[\dfrac{1}{f(x)}-1\right]+1}=\frac{1}{\dfrac{3}{f(x)}-2}=\frac{f(x)}{3-2f(x)}.$$