

AutoLISP

机械设计

高级应用实例

胡仁喜 胡星 史青录等编著

AutoLISP

COMPUTER AID DESIGN FOR MECHANISM

翔实的基础理论阐述
循序渐进的深入分析
丰富典型的实例引导



机械工业出版社
CHINA MACHINE PRESS

计算机辅助机械设计高级应用实例
COMPUTER AID DESIGN FOR MECHANISM

系列



计算机辅助机械设计高级应用实例系列

AutoLISP 机械设计高级应用实例

胡仁喜 胡星 史青录等编著

机械工业出版社

本书由浅入深,循序渐进地介绍了 AutoLISP 的基础知识和利用 AutoLISP 进行开发的方法与技巧。全书按知识的结构顺序分为三篇:第一篇为基础知识篇,主要介绍 AutoLISP 语言概述,AutoLISP 的基本要素,AutoLISP 基本函数,实用工具函数,操作 AutoCAD 对象等 AutoLISP 的基本知识和基本概念。第二篇为开发环境篇,主要介绍 Visual LISP 集成开发环境和对话框设计等知识。第三篇为开发应用篇,主要介绍用 AutoLISP 生成零件图和 AutoLISP 在 3D 绘图中的应用等知识。

本书语言简洁明了,讲解深入浅出。示例典型,详略得当。既可作为机械设计辅助设计课程与计算机应用课程的学习教材,也可作为 CAD 二次开发人员的自学或参考工具书。

图书在版编目(CIP)数据

AutoLISP 机械设计高级应用实例/胡仁喜等编著. —北京:机械工业出版社, 2005.1

(计算机辅助机械设计高级应用实例系列)

ISBN 7-111-15781-8

I. A… II. 胡… III. 计算机辅助设计-应用软件, AutoLISP
IV. TP391.72

中国版本图书馆 CIP 数据核字(2005)第 128653 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑:曲彩云 责任印制:李妍

北京蓝海印刷有限公司印刷·新华书店北京发行所发行

2005 年 1 月第 1 版第 1 次印刷

787mm×1092mm 1/16·22 印张·544 千字

0001-4000 册

定价:38.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话(010)68993821、88379646

68326294、68320718

封面无防伪标均为盗版

出版说明

机械设计是一门古老而成熟的学科，自第一次工业革命以来，经过几百年的发展，现在已经臻于完善。然而，传统的手工进行计算绘图的机械设计方法在日新月异的社会发展需求面前显得捉襟见肘，力不从心。以计算机为代表的信息技术推动整个社会各方面发展的同时也为机械设计这门古老的学科带来了新的生机，这就是计算机辅助设计（CAD）。

借助计算机提供的信息化平台，机械设计得以摆脱传统手工绘图和计算设计的烦琐和落后，变得高度自动化和精确化。可以说，CAD 技术在继承成熟的机械设计理论的同时已经彻底颠覆了传统机械设计几百年的人工模式。

目前，我国的机械设计学科也正在进行这场深刻的计算机革命。世界和国产的各种优秀 CAD 软件正以前所未有的速度迅速进入机械设计和制造行业的各个领域。由于目前 CAD 技术在我国机械设计工程界正处于一个快速导入期，各种 CAD 软件由于其自身的性能优势拥有不同的应用人群。本《计算机辅助机械设计高级应用实例系列》丛书针对机械设计行业不同应用对象，精选了目前国内应用最广泛的 12 种 CAD 软件，结集成书，力图全景式地介绍目前主流 CAD 软件的功能和特点，为不同需求人群设计一套完整的学习和应用指导工具书。

这套丛书所属各书目具有以下特点：

◆ 版本前沿

本丛书所有讲解对象软件都是世界或国内对应软件的最新版本，编者力图将目前世界上最新的，功能最强大的 CAD 软件介绍给读者，使读者能够学习到最前沿的知识。

◆ 内容精深

编者力图跳出目前世面低中端书籍的俗套，站在一个比较高的起点上，对每一个软件进行全貌式的讲解，从低端的基本功能介绍入手，循序渐进地逐步深入，直至对各种软件的高端分析与开发功能进行详细而具体地剖析。在此过程中，贯穿大量而又有机联系的实例，帮助读者在有限的篇幅内轻松而又深入地掌握本软件的知识精髓。

◆ 作者权威

本丛书各书目的作者都是相应软件使用方面的专家和技术权威，都有过相关软件的多年使用或教学经验，也是利用该软件进行 CAD 设计的高手，他们集中自己多年的心血，融化于字里行间，有很多地方都是他们经过反复研究得出的经验总结。

借这套丛书的出版，希望能够对广大读者的能力提高有所裨益。

前言

LISP(List Processing Language)是一种计算机的表处理语言,是在人工智能学科领域广泛应用的一种程序设计语言。AutoLISP 语言是嵌套于 AutoCAD 内部,将 LISP 语言和 AutoCAD 有机结合的产物。美国 AutoDesk 公司在 AutoCAD 内部嵌入 AutoLISP 的目的是使用户充分利用 AutoCAD 进行二次开发:实现直接增加和修改 AutoCAD 命令,随意扩大图形编辑功能,建立图形库和数据库并对当前图形进行直接访问和修改,开发 CAD 软件包等。

AutoLISP 语言最典型的应用之一是实现二维和三维参数化绘图程序设计,包括尺寸驱动程序,鼠标拖动程序等。另一个 AutoLISP 的典型应用就是驱动利用 AutoCAD 提供的 PDB 模块构成 DCL(Dialog Control Language)文件创建自己的对话框。

使用 AutoLISP 可直接调用几乎全部 AutoCAD 命令。AutoLISP 语言既具备一般高级语言的基本结构和功能,又具有一般高级语言所没有的强大的图形处理功能,是当今世界上 CAD 软件中被广泛采用的语言之一。

自从 AutoLISP 嵌入 AutoCAD 以后,使仅仅作作为交互式图形编辑软件的 AutoCAD 变成能真正进行计算机辅助设计、绘图的 CAD 软件,由于 LISP 灵活多变,又易于学习和使用,因而使 AutoCAD 成为功能很强的工具性软件。

本书按知识的结构顺序分为三篇:第一篇为基础知识篇,主要介绍 AutoLISP 语言概述,AutoLISP 的基本要素,AutoLISP 基本函数,实用工具函数,操作 AutoCAD 对象等 AutoLISP 的基本知识和基本概念。通过本篇的学习,读者可以掌握操作 AutoLISP 所必备的基础知识。第二篇为开发环境篇,主要介绍 Visual Lisp 集成开发环境和对话框设计等知识。通过本篇的学习,读者可以了解 AutoLISP 自身执行的环境和利用 AutoLISP 设计对象操作环境的方法。第三篇为开发应用篇,主要介绍用 AutoLISP 生成零件图和 AutoLISP 在 3D 绘图中的应用等知识。通过本篇的学习,读者可以掌握利用 AutoLISP 具体进行 AutoCAD 绘图的方法和技巧。

在编写过程中,考虑到 AutoLISP 知识的复杂性,为了使本书具有更好的亲和力和可读性,本书作者对全书知识进行了精心地编排和设计。在全面讲述 AutoLISP 知识的基础上突出了两大重点:

◆ 思维的连贯性:本书从简单到复杂,从基础到应用,层层深入地介绍了 AutoLISP 语言本身的函数结构和语法、AutoLISP 的开发环境和 AutoLISP 开发应用等知识。让读者在学习本书的过程中能够循序渐进,由浅入深地掌握利用 AutoLISP 进行开发和设计的技巧和方法。

◆ 知识的独立性:一般人认为,AutoLISP 作为 AutoCAD 的高级开发工具,必须在熟练掌握和应用 AutoCAD 的基础上才能掌握 AutoLISP 有关知识。而本书则极力突破这种传统思维,在知识讲解和内容设计上,力求作到使学习 AutoLISP 能够独立于 AutoCAD 之外,即读者不必对 AutoCAD 非常熟悉,就可以直接利用 AutoLISP 进行 AutoCAD 绘图和二次开发。

本书由胡仁喜、胡星、史青录主编,路纯红、辛文彤、王佩楷、周冰、张俊生、董伟、李瑞、周广芬、李鹏、王敏、陈丽芹、李世强、王兵学、袁涛、王渊峰、杨立辉、王玮、郑长松、王文平、孟清华、王涛等参与了部分章节编写。由于作者水平有限,书中错误之处在所难免。如有不当之处,敬请广大读者发送邮件至 hurenxi2000@163.com 批评指正。

作者

2005.1

目 录

出版说明

前言

第一篇 基础知识篇

第 1 章 AutoLISP 语言概述	2
本章重点	2
1.1 AutoLISP 语言简介	2
1.1.1 开发 AutoCAD 的重要工具	2
1.1.2 AutoLISP 的特点	3
1.2 AutoLISP 数据类型	3
1.2.1 原子	4
1.2.2 表和点对	5
1.3 AutoLISP 的程序结构	6
1.4 AutoLISP 的运行环境	7
1.5 AutoLISP 的内存分配	7
1.6 AutoLISP 程序的执行过程	8
1.6.1 加载和卸载 AutoLISP 文件	8
1.6.2 运行 AutoLISP 程序	10
1.7 完整的 AutoLISP 例程	10
本章小结	13
第 2 章 AutoLISP 的基本要素	14
本章重点	14
2.1 理解 AutoLISP 的变量和表达式	14
2.2 表达式的结构	15
2.2.1 数学表达式	16
2.2.2 矢量表达式	16
2.2.3 函数表达式	17
2.3 AutoLISP 的变量与类型	20
2.3.1 字符串型变量	20
2.3.2 整型变量	20
2.3.3 实型变量	21
2.3.4 表型变量	21
2.3.5 其他类型	21
2.4 变量的应用	22
2.4.1 使用 AutoLISP 变量	23

2.4.2 使用 AutoCAD 系统变量	24
2.5 创建用户自己的变量和表达式	24
本章小结	25
第 3 章 AutoLISP 基本函数	26
本章重点	26
3.1 数值函数	26
3.1.1 计算函数	28
3.1.2 布尔运算函数	33
3.1.3 三角函数	34
3.1.4 综合举例	35
3.2 字符串处理函数	36
3.2.1 求字符串长度函数 strlen (string length)	36
3.2.2 字符串链接函数 strcat (string catenation)	37
3.2.3 子串提取函数 substr (substring)	37
3.2.4 字母大小写转换函数 strcase	38
3.2.5 字符串模式匹配函数 wcmatch	38
3.2.6 综合举例	39
3.3 条件和循环函数	40
3.3.1 关系运算函数	40
3.3.2 逻辑运算函数	42
3.3.3 EQ 函数与 EQUAL 函数	43
3.3.4 条件函数	44
3.3.5 循环函数	47
3.4 表处理函数	50
3.4.1 表处理的基本函数	51
3.4.2 表的构造函数	53
3.4.3 表的循环处理函数	55
3.4.4 表的关联	56
3.4.5 综合举例	57
3.5 符号和函数处理函数	58
3.5.1 赋值函数	58
3.5.2 其他符号处理函数	60
3.5.3 函数处理函数	62
3.6 错误处理函数	66
3.7 应用程序处理函数	71
3.7.1 ADS 应用程序	71
3.7.2 ARX 应用函数	72
3.7.3 其他应用函数	73
3.8 综合举例	73

本章小结	79
第 4 章 实用工具函数	81
本章重点	81
4.1 查询和命令函数	81
4.1.1 command 函数	82
4.1.2 系统及环境变量	84
4.1.3 系统配置控制	85
4.1.4 综合举例	87
4.2 显示控制函数	89
4.2.1 文字输出函数	90
4.2.2 控制菜单	92
4.2.3 控制图形和文本窗口	93
4.2.4 控制低层图形	94
4.3 获取用户输入函数	96
4.3.1 用户输入函数	97
4.3.2 用户提示函数	106
4.3.3 用户输入函数条件的控制	109
4.4 几何实用函数	111
4.4.1 几何计算函数	111
4.4.2 对象捕捉	113
4.4.3 文本范围	113
4.5 转换函数	114
4.5.1 字符串转换	114
4.5.2 角度转换函数	117
4.5.3 ASCII 转换	119
4.5.4 单位转换	120
4.5.5 坐标系转换	121
4.6 文件处理函数	123
4.6.1 文件的搜索与获取	123
4.6.2 文件的打开与关闭	124
4.6.3 文件的输入输出函数	126
4.7 设备的访问和控制	128
4.7.1 获取用户输入	128
4.7.2 数字校准	130
4.8 综合举例	130
本章小结	134
第 5 章 操作 AutoCAD 对象	135

本章重点	135
5.1 选择集处理函数	135
5.2 对象处理函数	140
5.2.1 实体名操作函数	140
5.2.2 实体数据操作	141
5.2.3 多段线和轻装多段线	149
5.3 扩展数据处理函数	150
5.4 获取符号表和词典	151
5.4.1 符号表	151
5.4.2 词典条目	154
5.6 综合举例	155
本章小结	156
 第二篇 开发环境篇	
第 6 章 Visual LISP 集成开发环境	158
本章重点	158
6.1 Visual LISP 概述	158
6.2 启动和退出 Visual LISP 集成开发环境	159
6.2.1 启动 Visual LISP 集成开发环境	159
6.2.2 退出 Visual LISP 集成开发环境	159
6.3 Visual LISP 集成开发环境介绍	160
6.3.1 Visual LISP 菜单介绍	161
6.3.2 工具栏介绍	163
6.3.3 控制台操作	164
6.3.4 文本编辑器概述	166
6.3.5 加载和运行 AutoLISP 程序	167
6.4 文本编辑器主要功能及实现	169
6.4.1 文件/编辑/保存操作	169
6.4.2 代码分色显示	173
6.4.3 文本格式化	174
6.4.4 语法检查	178
6.5 调试程序	181
6.5.1 监视窗口	181
6.5.2 断点	183
6.5.3 单步调试程序	183
6.5.4 调试举例	184
本章小结	186
第 7 章 对话框设计	187

本章重点	187
本章典型效果图	187
7.1 对话框概述	188
7.2 对话框组件	188
7.3 用 DCL 定义对话框	189
7.3.1 base.DCL 和 acad.DCL 文件	189
7.3.2 引用 DCL 文件	190
7.3.3 DCL 语法	190
7.4 用 Visual LISP 显示对话框	193
7.4.1 显示对话框	193
7.4.2 预览错误处理	194
7.5 调整对话框的布局	195
7.5.1 在控件组中分配控件	195
7.5.2 调整控件间距	196
7.5.3 调整右端和底部的空间	196
7.5.4 调整加框行和列周围的空间	197
7.5.5 自定义退出按钮文本	197
7.6 设计指南	199
7.6.1 美观和功效	199
7.6.2 前后一致的设计和清晰明了的语言	199
7.6.3 用户控制	200
7.6.4 容错处理	200
7.6.5 提供帮助	201
7.6.6 考虑残疾用户	201
7.6.7 单词字母的大写	202
7.6.8 避免缩略语	202
7.6.9 布局	202
7.6.10 控件的大小和位置	202
7.6.11 禁用控件	203
7.6.12 嵌套对话框	203
7.6.13 隐藏对话框	203
7.6.14 提供默认值	203
7.6.15 处理键盘输入	203
7.6.16 预定义控件和控件组的指导	204
7.6.17 错误处理	206
7.7 对话框语言 DCL 详解	207
7.7.1 控件属性	207
7.7.2 DCL 属性目录	208
7.7.3 对话框控件的 DCL 语法	214

7.7.4 设计对话框举例	231
7.8 对话框驱动程序	234
7.8.1 在 AutoLISP 中调用设计的对话框	234
7.8.2 动作表达式和回调	237
7.8.3 列表框/下拉框处理	241
7.8.4 图像处理	246
7.8.5 对话框嵌套	248
7.8.6 隐藏对话框	250
7.8.7 特定应用数据	252
7.9 综合举例	253
本章小结	256
第三篇 开发应用篇	
第 8 章 用 AutoLISP 生成零件图	258
本章重点	258
本章典型效果图	258
8.1 参数化绘图概述	259
8.2 一个简单的参数化绘图的例子	259
8.3 理解参数化设计	263
8.4 开发参数化系统的基本要素	263
8.5 综合举例	264
8.5.1 参数化设计的一般步骤	264
8.5.2 绘制轴	264
8.5.3 绘制带轮	274
本章小结	285
第 9 章 AutoLISP 在 3D 绘图中的应用	286
本章重点	286
本章典型效果图	286
9.1 三维绘图概述	287
9.1.1 三维造型的优越性	287
9.1.2 用 AutoLISP 在三维绘图中的应用	287
9.2 三维造型基础	288
9.2.1 UCS 三维用户坐标系统	288
9.2.2 建立用户坐标系	289
9.2.3 创建三维实体	295
9.3 用 AutoLISP 进行三维参数化绘图	299
9.4 用 AutoLISP 进行曲线绘制	308
本章小结	314

附录	315
附录 A command 函数	315
附录 B AutoCAD 单位定义	321
附录 C 错误提示	326
附录 D AutoLISP 的系统变量	329

第一篇

基础知识篇

计算机辅助机械设计高级应用实例系列
COMPUTER AID DESIGN FOR MECHANISM

本篇介绍以下主要知识点:

- ❖ *AutoLISP* 语言概述
- ❖ *AutoLISP* 的基本要素
- ❖ *AutoLISP* 基本函数
- ❖ 实用工具函数
- ❖ 操作 *AutoCAD* 对象

第 1 章 AutoLISP 语言概述

内 容 提 要

LISP 是一种计算机的表处理语言,是在人工智能学科领域广泛应用的一种程序设计语言。本章介绍了 *AutoLISP* 语言的优点以及使用场合, *AutoLISP* 的基本数据结构、程序结构,以及 *AutoLISP* 程序的编辑、加载和运行方法。最后,演示了两个小的 *AutoLISP* 程序。



本章重点



了解 AutoLISP 的基本数据类型



了解 AutoLISP 的程序结构



掌握如何执行 AutoLISP 程序

1.1 AutoLISP 语言简介

LISP(List Processing Language)是一种计算机的表处理语言,是在人工智能学科领域广泛应用的一种程序设计语言。AutoLISP 语言嵌套于 AutoCAD 内部,将 LISP 语言和 AutoCAD 有机结合的产物。AutoLISP 是开发 AutoCAD 的重要工具之一。

1.1.1 开发 AutoCAD 的重要工具

使用 AutoLISP 可直接调用几乎全部 AutoCAD 命令。AutoLISP 语言既具备一般高级语言的基本结构和功能,又具有一般高级语言所没有的强大的图形处理功能,是当今世界上 CAD 软件中被广泛采用的语言之一。

美国 AutoDesk 公司在 AutoCAD 内部嵌入 AutoLISP 的目的是使用户充分利用 AutoCAD 进行二次开发:实现直接增加和修改 AutoCAD 命令,随意扩大图形编辑功能,建立图形库和数据库并对当前图形进行直接访问和修改,开发 CAD 软件包等。

AutoLISP 语言最典型的应用之一是实现参数化绘图程序设计,包括尺寸驱动程序、鼠标拖动程序等。尺寸驱动是指通过改变实体标注的尺寸值来实现图形的自动修改;鼠标拖动即利用 AutoLISP 语言提供的(`GRREAD[<track>]`)函数,让用户直接读取 AutoCAD 的输入设备(如

鼠标),任选项追踪光标移动存在且为真时,通过鼠标移动光标,调整所需的参数值而达到自动改变屏幕图形大小和形状。

到目前为止,大多数参数化程序都是针对二维平面图编制的。实际上,立体图同样可以实现参数化绘图,在 AutoCAD 1 3 中编制实体的立体图参数化程序比其平面三视图程序更简单,而且立体图生成后,可以很方便地生成三视图、剖面图和轴侧图等。

另一个 AutoLISP 的典型应用就是驱动利用 AutoCAD 提供的 PDB 模块构成 DCL(Dialog Control Language)文件创建自己的对话框。

自从 AutoLISP 嵌入 AutoCAD 以后,使仅仅作作为交互式图形编辑软件的 AutoCAD 变成能真正进行计算机辅助设计、绘图的 CAD 软件,由于 LISP 灵活多变,又易于学习和使用,因而使 AutoCAD 成为功能很强的工具性软件。

1.1.2 AutoLISP 的特点

AutoLISP 具有如下功能:

(1) AutoLISP 语言是在普通的 LISP 语言基础上,扩充了许多适用于 CAD 应用的特殊功能而形成的,一种仅能以解释方式运行于 AutoCAD 内部的解释性程序设计语言。

(2) AutoLISP 语言中的一切成分都是以函数的形式给出的,它没有语句概念或其他语法结构。执行 AutoLISP 程序就是执行一些函数,再调用其他函数。

(3) AutoLISP 把数据和程序统一表达为表结构,即 S—表达式,故可把程序当作数据来处理,也可把数据当作程序来执行。

(4) AutoLISP 语言中的程序运行过程就是对函数求值的过程,是在对函数求值的过程中实现函数的功能。

(5) AutoLISP 语言的主要控制结构是采用递归方式。递归方式的使用,使得程序设计简单易懂。

1.2 AutoLISP 数据类型

AutoLISP 语言主要用到如下数据类型:

- ◆ 符号 (SYM)
- ◆ 字符串 (STR)
- ◆ 表(及用户定义的函数) (LIST)
- ◆ 文件描述符 (FILE)
- ◆ AutoLISP 的内部函数 (SUBR)
- ◆ AutoCAD 的选择集 (PICKSET)
- ◆ AutoCAD 的实体名 (ENAME)
- ◆ 函数分页表 (PAGETB)

本节只介绍前五种数据类型,其他类型将在相应的章节中介绍。

在这五种数据类型中,前四种称为原子(ATOM),原子包括数字原子(整形数和实型数)、符号原子和串原子。所以 AutoLISP 最基本的数据类型是原子和表,它们又总称为符号表达式(Symbolic-Expression),也称为 S—表达式。

1.2.1 原子

1. 整数

整数是由 0、1、2……9、+、- 等字符组成，整数的大小与所使用的计算机系统有关。

2. 实型数

实型数用双精度浮点数表示，并且至少有 14 位的精度，即整数后跟小数。如果实数的绝对值小于 1，小数点前必须加 0，不能直接以小数点开头，否则被误认为点对而出错。实型数的范围比整型数大得多，如对于 16 位微机，实型数范围约为 $-1.797693 \times 10^{308} \sim 1.79793 \times 10^{308}$ 。它不易超界，故用户可以尽量采用实型数。

实型数也可采用科学记数法表示，如 0.12×10^9 可表示为 0.12E9。

3. 符号

(1) 符号(symbol)包括除 “)”、“(”、“.”、“,”、“'”、“;” 之外的任何打印字符。

(2) 符号原子的长度没有限制，但尽量不要超过 6 个字符，否则要占用额外的内存，降低运行速度。

(3) 在 AutoLISP 中符号的大小写是等效的，如以下的符号原子都是合法的：

A A12 PC x-38-6 *A

(4) AutoLISP 中的任何符号都是有值的，即符号都要赋以一定的数值，或者说符号总是约束在一定值上。一般用赋值函数 `setq` 进行赋值。

例如：

`(setq x 25.0)`

意思是将 25.0 赋给 x，这时 x 的当前约束值即为 25.0。一个符号在使用前如没有赋以任何值，则该符号的值为 nil(空)，它不占用内存空间。



注意

为区别起见，我们常用术语“符号”来指存储静态数据的一个符号名，例如内建式函数和用户定义函数名的一个符号。

用术语“变量”来指存储程序数据的符号名，如上述 `(setq x 25.0)` 中的变量名为 x，它的值为 25.0。AutoLISP 程序中每一个变量都要消耗少量内存，故当变量值不再有用时，重复使用变量名或将变量值设置成 nil 是良好的程序设计习惯。符号名或变量名不能包含空格字符或分隔符，并总是以字母开头。其值保持不变的变量称为常量，AutoLISP 仅用一个常量 Pi。Pi 的值约为 3.14159。

4. 字符串

字符串是由包括在一对双引号内的一组字符组成的，如：

“ABC” “135” “AbC” “ ”

字符串包括任何可打印的字符。字符串中字母的大小写及空格都是有意义的。字符串的最大长度为 100 个字符, 位于 100 之后的字符无效。若字符串中没有任何字符, 则为空串 “”。

1.2.2 表和点对

1. 表

在 AutoLISP 语言中, 表有如下特点:

- (1) 表是指放在一对相匹配的左、右括号中的一个或多个元素的有序集合。
- (2) 表中的每一个元素可以是任何类型的 S—表达式, 即可以是数字、符号、字符串, 也可以是表。
- (3) 元素与元素之间要用空格隔开, 而元素与括弧之间可不用空格, 因为括弧本身就是有效的分隔号。如: (15 (a b) c d), 在此例中, 表内有 4 个元素, 即 15、(a b)、c 和 d, 其中第二个元素是表。
- (4) 表是可以任意嵌套的, 上例表中即嵌套了一个表 (a b)。表可以嵌套很多层, 从外层向里依次称为 0 层 (也称顶层)、1 层、2 层……我们所指的表中的元素是指表的顶层元素。
- (5) 表中的元素是有顺序的, 为便于对表中元素进行存取, 每个元素都有一个序号。从左向右, 第一个元素的序号为 0, 第二个元素的序号为 1, 第 i 个元素序号为 $i-1$ 。
- (6) 表的大小为表的长度, 即表中顶层元素的个数。没有任何元素的表称为空表。空表用 () 或 nil 表示。在 AutoLISP 语言中, nil 是一个特殊符号原子, 它既是原子又是表。
- (7) 表有两种类型: 标准表和引用表。

- ◆ 标准表: 标准表是 AutoLISP 程序的基本结构形式, AutoLISP 程序就是由标准表组成的。标准表是用于函数的调用, 其中第一个元素必须是系统内部函数或用户定义的函数, 其他的元素为该函数的参数, 如上面提到的赋值函数的调用, 即采用标准表的形式。(setq x 25.0) 表中第一个元素 setq 为系统内部定义的赋值函数, x 和 25.0 均为 setq 的参数。
- ◆ 引用表: 这种表第一个元素不是函数, 即不作为函数调用, 常作为数据处理。引用表的一个重要应用是表示图中的点的坐标。当表示点的坐标时, 表中的元素是用实型数构成的。表示二维点的坐标是用两个实型数构成的表, 如 (20.0 30.5), 其中第一个元素表示点的 x 轴坐标, 第二个元素表示点的 y 轴坐标。三维点的坐标表示, 是用三个实型数构成的表, 如 (20.0 82.5 1.0), 其中三个元素依次表示点的 x 轴坐标、y 轴坐标和 z 轴坐标。

2. 点对 (dotted pair)

点对也是一种表, 该表中有两个元素, 两元素中间为一圆点 “·”, 且圆点与元素之间必须用空格分开。

例如:

(A · B)

就是一个点对, A、B 与圆点均用空格分开, 其中第一个元素 A 为该点对的左元素, 第二个元 B 为点对的右元素。点对亦可任意嵌套。当使用点对时, 切记要注意它的书写格式。