

可下载教学资料

<http://www.tup.tsinghua.edu.cn>



高等学校教材
计算机应用

C程序设计教程

姜桂洪 王军 黄宝香 谷亚鹏 编著



清华大学出版社

TP312/2682

2008

高等学校教材

计算机应用

C程序设计教程

姜桂洪 王军 黄宝香 谷亚鹏 编著

清华大学出版社

北京

内 容 简 介

本书采用 Turbo C++ 3.0 作为语言环境,系统地介绍了进行 C 程序设计的基本操作、算法描述、数据类型、流程控制、编程技巧等内容。语言通俗、简洁、流畅,内容论述深入浅出,例题讲解细致详尽且具有代表性,让具备一定的计算机基础知识的计算机语言初学者能够自己读明白。

全书共 12 章,知识体系完整,结构安排合理,内容深度适宜,并将习题分为课堂测试、课后习题和单元测试 3 个层次。

本书适合作为高等院校本、专科学生学习 C 语言的教材,也可作为计算机编程人员和相关领域的技术工作者自学与参考用书。

另外,本书还配有辅助教材《C 程序设计教程习题解答与上机指导》。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13501256678 13801310933

图书在版编目(CIP)数据

C 程序设计教程/姜桂洪,王军,黄宝香,谷亚鹏编著. —北京:清华大学出版社,2008.1
(高等学校教材·计算机应用)

ISBN 978-7-302-15913-1

I. C… II. ①姜… ②王… ③黄… ④谷… III. C 语言—程序设计—高等学校—教材
IV. TP312

中国版本图书馆 CIP 数据核字(2007)第 123156 号

责任编辑:魏江江 李玮琪

责任校对:时翠兰

责任印制:何 芊

出版发行:清华大学出版社 地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn> 邮 编:100084

c-service@tup.tsinghua.edu.cn

社 总 机:010-62770175 邮购热线:010-62786544

投稿咨询:010-62772015 客户服务:010-62776969

印 刷 者:北京国马印刷厂

装 订 者:三河市金元印装有限公司

经 销:全国新华书店

开 本:185×260 印张:25.75 字数:618 千字

版 次:2008 年 1 月第 1 版 印 次:2008 年 1 月第 1 次印刷

印 数:1~4000

定 价:38.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:025181-01

改革开放以来，特别是党的十五大以来，我国教育事业取得了举世瞩目的辉煌成就，高等教育实现了历史性的跨越，已由精英教育阶段进入国际公认的大众化教育阶段。在质量不断提高的基础上，高等教育规模取得如此快速的发展，创造了世界教育发展史上的奇迹。当前，教育工作既面临着千载难逢的良好机遇，同时也面临着前所未有的严峻挑战。社会不断增长的高等教育需求同教育供给特别是优质教育供给不足的矛盾，是现阶段教育发展面临的基本矛盾。

教育部一直十分重视高等教育质量工作。2001年8月，教育部下发了《关于加强高等学校本科教学工作，提高教学质量的若干意见》，提出了十二条加强本科教学工作提高教学质量的措施和意见。2003年6月和2004年2月，教育部分别下发了《关于启动高等学校教学质量与教学改革工程精品课程建设工作的通知》和《教育部实施精品课程建设提高高校教学质量和人才培养质量》文件，指出“高等学校教学质量和教学改革工程”是教育部正在制定的《2003—2007年教育振兴行动计划》的重要组成部分，精品课程建设是“质量工程”的重要内容之一。教育部计划用五年时间（2003—2007年）建设1500门国家级精品课程，利用现代化的教育信息技术手段将精品课程的相关内容上网并免费开放，以实现优质教学资源共享，提高高等学校教学质量和人才培养质量。

为了深入贯彻落实教育部《关于加强高等学校本科教学工作，提高教学质量的若干意见》精神，紧密配合教育部已经启动的“高等学校教学质量与教学改革工程精品课程建设工作”，在有关专家、教授的倡议和有关部门的大力支持下，我们组织并成立了“清华大学出版社教材编审委员会”（以下简称“编委会”），旨在配合教育部制定精品课程教材的出版规划，讨论并实施精品课程教材的编写与出版工作。“编委会”成员皆来自全国各类高等学校教学与科研第一线的骨干教师，其中许多教师为各校相关院系主管教学的院长或系主任。

按照教育部的要求，“编委会”一致认为，精品课程的建设工作从开始就要坚持高标准、严要求，处于一个比较高的起点上；精品课程教材应该能够反映各高校教学改革与课程建设的需要，要有特色风格、有创新性（新体系、新内容、新手段、新思路，教材的内容体系有较高的科学创新、技术创新和理念创新的含量）、先进性（对原有的学科体系有实质性的改革和发展、顺应并符合新世纪教学发展的规律、代表并引领课程发展的趋势和方向）、示范性（教材所体现的课程体系具有较广泛的辐射性和示范性）

和一定的前瞻性。教材由个人申报或各校推荐（通过所在高校的“编委会”成员推荐），经“编委会”认真评审，最后由清华大学出版社审定出版。

目前，针对计算机类和电子信息类相关专业成立了两个“编委会”，即“清华大学出版社计算机教材编审委员会”和“清华大学出版社电子信息教材编审委员会”。首批推出的特色精品教材包括：

（1）高等学校教材·计算机应用——高等学校各类专业，特别是非计算机专业的计算机应用类教材。

（2）高等学校教材·计算机科学与技术——高等学校计算机相关专业的教材。

（3）高等学校教材·电子信息——高等学校电子信息相关专业的教材。

（4）高等学校教材·软件工程——高等学校软件工程相关专业的教材。

（5）高等学校教材·信息管理与信息系统。

（6）高等学校教材·财经管理与计算机应用。

清华大学出版社经过 20 多年的努力，在教材尤其是计算机和电子信息类专业教材出版方面树立了权威品牌，为我国的高等教育事业做出了重要贡献。清华版教材形成了技术准确、内容严谨的独特风格，这种风格将延续并反映在特色精品教材的建设中。

清华大学出版社教材编审委员会
E-mail: dingl@tup.tsinghua.edu.cn

C 语言是一种描述能力强、可移植性好、硬件控制能力高、逻辑思维缜密的结构化程序设计语言。目前,许多高校将 C 语言选作学生学习计算机语言的启蒙语言,在“数据结构”课程中也已广泛采用 C 语言作为背景语言。此外,C 语言还广泛应用于电子信息领域。

由于 C 语言牵涉的概念比较复杂、规则繁多、使用灵活、容易出错,不少初学者感到学习困难,本书以“计算机文化基础”为先导课程,依据多年的实际教学经验,并参考借鉴了多种同类教材的优点,对该书的知识结构体系及内容讲述的逻辑顺序进行了精心的设计和安排。力求达到结构顺序合理,内容深度适宜。采用通俗易懂、简洁流畅的语言,论述深入浅出,例题典型全面且讲解细致详尽。

本书采用 Turbo C++ 3.0 作为语言环境,所有例题均在 Turbo C++ 3.0 语言环境下上机调试并通过。适合作为高等院校本科、专科学生的学习 C 语言的课程教材,也可以作为自学和参考用书。

全书共分 12 章,主要内容如下:

- C 语言运行环境的基本操作、C 语言的特点和学习 C 语言必备的基础知识。
- C 语言的基本数据类型、运算符、表达式及各种数值类型数据间的转换和运算。
- 算法概念、库函数的使用、数据输入输出。
- 选择结构的算法描述、关系运算、逻辑运算、if 语句和 switch 语句的应用。
- 循环结构的算法描述和 while、do... while、for 等循环语句及其循环嵌套的应用。
- 数组的概念、定义、引用、初始化及字符数组的输入输出。
- 函数的定义、返回值、嵌套调用、数组作函数参数、变量的存储类别。
- 宏定义、文件包含、条件编译等预处理命令。
- 指针的概念、指针变量的定义和使用、指针和数组、指针与字符串、指针和函数及指针数组的应用。
- 结构体、链表、共用体和枚举类型。
- C 文件的概念、分类及打开、关闭、读写、定位与随机读写等基本操作。
- C 程序的常用调试方法和步骤、C 程序的常见错误分析。

本书习题类型丰富,创新性地将习题按照学习 C 语言的认知规律分为 3 个层次。

- 课堂测试，有针对性地在介绍主要内容的小节后面配有课堂测试练习，检测读者是否理解了本小节的基本内容。
- 课后习题，每章配有类型丰富、题材典型的习题，帮助读者巩固所学知识。
- 单元测试，将全书分为 3 个单元，每单元后面配有 1 套测试练习题，用于检测读者对本部分内容掌握的情况。

另外，本书还配有辅助教材《C 程序设计教程习题解答与上机指导》。

本书主要由姜桂洪、谷亚鹏、王军、黄宝香等编写，全书由姜桂洪统稿。刘秋香、邢雪宁、潘金凤、孙小敏、崔国伟等老师，还有许多同学曾对本书编写给予大力支持和帮助，在此由衷地表示谢意。

由于编撰时间仓促，加之水平有限，书中难免有不妥之处，恳请读者批评指正。

编 者

2007 年 5 月

第 1 章 C 语言概述	1
1.1 C 语言的发展历程	1
1.2 数据在计算机内的存储形式和表示方法	2
1.2.1 内存的组织形式	2
1.2.2 数据的二进制表示方法	3
1.2.3 八进制数	5
1.2.4 十六进制数	6
1.2.5 原码、反码和补码	8
1.3 利用计算机处理问题的基本过程	10
1.4 最简单的 C 程序的构成	11
1.4.1 C 语言的特点	11
1.4.2 C 语言中的字符集	12
1.4.3 C 程序的构成	14
1.5 C 语言的运行环境和上机步骤	18
1.5.1 C 语言的运行环境	18
1.5.2 C 程序的上机步骤	20
1.6 小结	24
习题	24
第 2 章 数据类型、运算符和表达式	26
2.1 C 的数据类型	26
2.2 常量和变量	27
2.2.1 常量和符号常量	27
2.2.2 变量	27
2.3 整型数据	28
2.3.1 整型常量的表示方法	28
2.3.2 整型变量	29
2.3.3 整型常量的类型	33

2.4	实型数据	33
2.4.1	实型常量的表示方法	33
2.4.2	实型变量	34
2.4.3	实型常量的类型	35
2.5	字符型数据	36
2.5.1	字符型常量	36
2.5.2	字符型变量	38
2.5.3	字符串常量	40
2.6	变量初始化和变量赋初值	41
2.7	算术运算符和算术表达式	42
2.7.1	C 语言运算符简介	42
2.7.2	基本算术运算符和算术表达式	43
2.7.3	自增、自减运算符	44
2.7.4	有关算术表达式使用中的问题说明	45
2.8	各种数值类型数据间的转换和运算	47
2.8.1	隐式转换	47
2.8.2	强制类型转换	48
2.9	赋值运算符和赋值表达式	48
2.10	位运算	53
2.10.1	位运算符及其功能	53
2.10.2	位运算举例	57
2.11	逗号运算符和逗号表达式	59
2.12	小结	60
	习题	60
第 3 章	算法和顺序结构程序设计	64
3.1	算法概念和算法的表示方法	64
3.1.1	算法概念	64
3.1.2	算法的表示方法	66
3.2	C 语句概述	70
3.3	数据的输入输出和库函数的使用	73
3.3.1	数据的输入输出概念	73
3.3.2	库函数的使用	73
3.4	putchar 函数和 getchar 函数	76
3.4.1	putchar 函数	76
3.4.2	getchar 函数	76
3.5	数据的格式输出 printf 函数	77
3.6	数据的格式输入 scanf 函数	84
3.7	结构化程序设计的一般步骤	88

3.8 顺序结构程序设计举例	91
3.9 小结	93
习题	93
第4章 选择结构程序设计	96
4.1 选择结构能够解决的问题及算法表示	96
4.2 关系运算符和关系表达式	99
4.2.1 关系运算符及其优先次序	99
4.2.2 关系表达式	100
4.3 逻辑运算符和逻辑表达式	101
4.3.1 逻辑运算符及其优先次序	101
4.3.2 逻辑表达式	102
4.4 if 语句	104
4.4.1 if 语句的基本形式	104
4.4.2 if 语句嵌套	107
4.5 条件运算符和条件表达式	111
4.6 switch 语句	113
4.7 选择结构程序设计举例	115
4.8 小结	116
习题	117
第5章 循环结构程序设计	121
5.1 循环结构能够解决的问题及算法表示	121
5.2 while 语句	124
5.3 do...while 语句	126
5.4 for 语句	129
5.5 循环嵌套	133
5.6 break 语句和 continue 语句	135
5.7 循环结构程序举例	137
5.8 小结	139
习题	140
单元自测练习 (1)	144
第6章 数组	147
6.1 数组的概念及数组在内存中的存储形式	147
6.1.1 数组概念	147
6.1.2 数组在内存中的存储形式	148
6.2 一维数组的定义和引用	148

6.2.1	一维数组的定义	148
6.2.2	一维数组元素的引用	149
6.2.3	一维数组的初始化	150
6.2.4	一维数组程序举例	151
6.3	二维数组的定义和引用	153
6.3.1	二维数组的定义	153
6.3.2	二维数组元素的引用	154
6.3.3	二维数组的初始化	155
6.3.4	二维数组程序举例	157
6.4	字符数组的定义和引用	158
6.4.1	字符数组的定义	159
6.4.2	字符数组的初始化	159
6.4.3	字符数组元素的引用	160
6.4.4	字符串及其结束标志	161
6.4.5	字符数组的输入输出	163
6.4.6	常用字符串处理函数	165
6.4.7	字符数组应用举例	169
6.5	小结	171
	习题	172
第 7 章	函数	175
7.1	C 程序的基本结构	175
7.2	函数的定义和函数的返回值	178
7.2.1	函数定义的一般形式	178
7.2.2	函数参数	181
7.2.3	函数的返回值	183
7.2.4	函数的调用	184
7.3	函数的嵌套调用和递归调用	187
7.3.1	函数的嵌套调用	187
7.3.2	函数的递归调用	189
7.4	数组作函数参数	192
7.4.1	数组元素作函数参数	192
7.4.2	一维数组名作函数参数	194
7.4.3	多维数组名作函数参数	200
7.5	局部变量和全局变量	202
7.5.1	局部变量	202
7.5.2	全局变量	204
7.6	变量的存储类别	208
7.6.1	动态存储和静态存储	208

7.6.2 auto 变量	208
7.6.3 利用 static 声明局部变量	209
7.6.4 register 变量	210
7.6.5 利用 extern 声明外部变量	211
7.6.6 利用 static 声明全局变量	213
7.7 内部函数和外部函数	214
7.7.1 内部函数	214
7.7.2 外部函数	215
7.8 小结	215
习题	217
第 8 章 预处理命令	219
8.1 宏定义	220
8.1.1 不带参数的宏定义	220
8.1.2 带参数的宏定义	223
8.2 文件包含	229
8.3 条件编译	232
8.4 小结	234
习题	235
单元自测练习 (2)	237
第 9 章 指针	241
9.1 有关指针的基本概念	241
9.2 指针变量的定义和使用	244
9.2.1 指针变量的定义	244
9.2.2 指针变量的引用	245
9.2.3 指针变量作函数参数	248
9.2.4 多级指针	250
9.3 数组和指针	251
9.3.1 数组元素与指针变量	252
9.3.2 数组名作函数参数	256
9.3.3 多维数组与指针	263
9.4 字符串与指针	272
9.4.1 字符数组、字符串和字符指针的关系	272
9.4.2 字符串指针作函数参数	274
9.4.3 字符串指针和字符数组的区别	276
9.5 函数和指针	277
9.5.1 指向函数的指针变量	277

9.5.2 用函数指针变量调用函数	278
9.5.3 用指向函数的指针变量作函数参数	279
9.5.4 返回指针值的函数	281
9.6 指针数组及应用	282
9.6.1 指针数组	282
9.6.2 指针数组作 main 函数的参数	285
9.7 void 指针类型	287
9.8 小结	288
习题	290
第 10 章 结构体、共用体和枚举类型	294
10.1 结构体类型数据的定义和使用	294
10.1.1 结构体类型的定义	294
10.1.2 结构体变量的定义	296
10.1.3 结构体变量的引用	297
10.1.4 结构体变量的初始化	299
10.1.5 结构体数组	300
10.1.6 位段	304
10.2 结构体类型数据与指针	307
10.2.1 指向结构体变量的指针	307
10.2.2 指向结构体数组的指针	309
10.2.3 用指向结构体变量的指针作函数参数	311
10.3 链表	314
10.3.1 链表基本结构与定义	314
10.3.2 处理动态链表所需的函数	317
10.3.3 对链表的基本操作	318
10.3.4 对链表的综合操作	323
10.4 共用体	325
10.4.1 共用体概念	325
10.4.2 共用体变量的引用	327
10.4.3 共用体类型数据的特点	328
10.5 枚举类型	329
10.5.1 枚举类型的定义	329
10.5.2 枚举类型数据的特点	330
10.6 利用 typedef 定义类型	332
10.7 小结	334
习题	334
第 11 章 文件	338
11.1 C 文件概述	338

11.2 文件的打开和关闭	341
11.2.1 文件的打开	341
11.2.2 文件的关闭	343
11.3 文件的读写	344
11.3.1 单个字符的读写函数	344
11.3.2 fread 函数和 fwrite 函数	348
11.3.3 fprintf 函数和 fscanf 函数	351
11.3.4 其他常用读写函数	351
11.4 文件的定位与随机读写	354
11.5 出错的检测	356
11.6 小结	357
习题	358
第 12 章 C 程序调试和常见错误	361
12.1 程序调试	361
12.2 C 程序常见错误分析	364
12.3 小结	375
单元自测练习 (3)	376
附录 A 常用字符与 ASCII 码对照表	380
附录 B C 运算符的优先级和结合性	382
附录 C 常用库函数	384
附录 D C 语言中的关键字及其用途	389
附录 E 课堂测试题答案	390

C 语言概述

1.1 C 语言的发展历程

从电子计算机诞生到现在,程序设计语言先后经历了机器语言、汇编语言、面向过程的程序设计语言和面向对象的程序设计语言 4 个发展阶段。在目前数百种计算机语言中,C 语言仍然是国际上广泛流行的计算机语言。它适合作为系统描述语言,既可以用来编写系统软件,也可用来编写应用软件。

C 语言的发展过程可以粗略地分为诞生(1970 年—1973 年)、发展(1973 年—1988 年)和成熟(1988 年以后)3 个阶段。

C 语言是在 B 语言的基础上发展起来的。1970 年美国贝尔实验室的 Ken Thompson 设计出了很简单的而且很接近硬件的 B 语言,并用 B 语言编写了第一个 UNIX 操作系统,在 PDP-7 上实现。1972 年到 1973 年间,贝尔实验室的 D. M. Ritchie 在 B 语言的基础上设计出了 C 语言。C 语言既保持了 B 语言精练、接近硬件的优点,又克服了过于简单、数据无类型等缺点。

1973 年, Ken Thompson 和 D. M. Ritchie 合作把 UNIX 的 90% 以上代码用 C 语言改写(即 UNIX 第 5 版)。后来, C 语言做了多次改进,但主要还是在贝尔实验室内部使用。直到 1975 年 UNIX 第 6 版发布后, C 语言的突出优点才引起人们的普遍注意。

1977 年出现了不依赖于具体机器的 C 语言编译文本——《可移植 C 语言编译程序》,使 C 语言移植到其他机器时所需做的工作大大简化了,这也推动了 UNIX 操作系统迅速地在各种机器上应用。随着 UNIX 的日益广泛使用, C 语言也迅速得到推广。C 语言和 UNIX 操作系统在发展过程中可以说是相辅相成的。

1978 年以后, C 语言先后移植到大、中、小、微型计算机上,并独立于 UNIX 操作系统和 PDP 计算机。于是, C 语言便很快风靡全世界,成为世界上应用最广泛的几种计算机语言之一。以 1978 年发布的 UNIX 第 7 版中的 C 语言编译程序为基础, Brian W. Kernighan 和 D. M. Ritchie 合著了影响深远的经典著作《The C Programming Language》,这本书成为后来广泛使用的 C 语言版本的基础。

1983 年,美国国家标准化协会(ANSI)根据 C 语言的各种版本对 C 语言的发展和扩充,制定了新的标准,称为 ANSI C。ANSI C 比原来的标准 C 有了很大的发展。

1987 年, ANSI 又公布了新标准——87 ANSI C。1988 年, 美国国家标准协会 (ANSI) 在综合各种 C 语言版本的基础上制定了 C 语言文本标准, 称为 ANSI C 标准。ANSI C 实现了 C 语言的规范化和统一化。Brian.W.Kernighan 和 Dennis.M.Ritghie 按照 ANSI C 标准重写了《The C Programming Language》一书, 并于 1990 年正式发表了《The C Programming Language Second Edition》。

1990 年国际标准化组织 (ISO) 公布了以 ANSI C 为基础制定的 C 语言的国际标准 ISO C, 人们通常称之为标准 C。

C 标准的制定标志着 C 语言的成熟, 1988 年以后推出的各种 C 语言版本对标准 C 是兼容的。1994 年 ISO 修订了 C 语言标准。目前流行的各种 C 语言编译系统的版本多是以 ANSI C 标准为基础开发的, 但不同版本之间语言功能与语法规则略有差别, 详细情况可以查阅有关手册。本书的叙述基本上以 ANSI C 为基础。

1.2 数据在计算机内的存储形式和表示方法

利用 C 语言编程, 必须了解数据在计算机内存中的组织形式。了解计算机内存的组织结构和数据在内存中存储的形式是学好计算机语言的必要条件, 也是学习包括 C 语言在内的所有计算机语言的基础。

1.2.1 内存的组织形式

1. 内存存储器 RAM

内存存储器 RAM (Random Access Memory) 是直接和 CPU 相联系的存储设备, 是计算机工作的存储区域, 一切要执行的程序和数据都要先装入该存储器内, 然后再进行一系列的操作。CPU 在工作时直接从 RAM 中读取数据, 而 RAM 中的数据来自外存, 并随着计算机的工作随时变化。

RAM 中的数据可以反复读出, 只有向存储器写入新数据时存储器中的内容才会被更新。RAM 中的数据随着计算机的断电自然消失。因此, RAM 是计算机处理数据的临时存储区。

2. 内存存储器的组织形式

内存存储器是由微小的电子线路单元组成的, 每一个单元称为一个位 (bit, b)。由于可以以 0 和 1 标记常用电器元件的两个稳定工作状态, 如电压的高和低、电灯的亮和灭、电容的充电和放电、脉冲的有和无、晶体管的导通和截止等, 所以计算机内存中的信息是以二进制形式存在, 用二进制存储数据可以准确表达电器元件的两种稳定工作状态。

内存存储器通常是由集成电路组成的, 它可以由十几万到上万亿个位组成, 为便于管理, 常将若干个位组成一个字节 (Byte, B), 多数计算机以 8 个位作为一个字节。一个字节可以存放 8 个位的二进制数, 如 0110 0111。数据在内存中存储时是以字节为单位的 (例如在微机中, 一个字符可以用一个字节存储, 一个整数可以用 2 个字节存储, 一个实数可以用

4 个字节存储)。若干个字节组成一个字 (word)，一个字中可以存放一条计算机指令或一个数据。

如果一个计算机系统以 64 位二进制信息表示一条指令，就称这台计算机的“字长”为 64 位。通常所说的“64 位机”就是以 64 位作为一个字的，一次传输的信息为 64 个位。每个字节有一个“地址”。只有通过地址才能找到某个存储单元，并从中存取数据或地址。存储器中存储的数据与地址的关系，就如同教室一样，教学楼的教室的编号就是地址，只有通过教室的编号才能找到上课的同学。请注意区分“存储单元的地址”与“存储单元中的内容”。如图 1.1 所示。

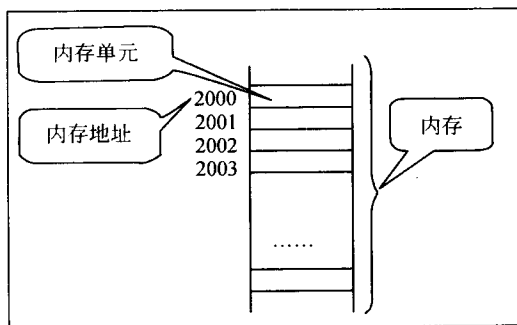


图 1.1 内存存储器的结构示意图

电子计算机中的基本存储单元为字节，并以字节为单位进行计量，以 B (Byte) 为字节的标志。目前常用的计量单位还有 KB、MB、GB 和 TB 等，其大小分别为 2^{10} 、 2^{20} 、 2^{30} 、 2^{40} 个字节。单位之间的换算关系如下：

$$\begin{aligned} 1\text{B} &= 8\text{b} & 1\text{KB} &= 1024\text{B} & 1\text{MB} &= 1024\text{KB} \\ \text{GB} &= 1024\text{MB} & 1\text{TB} &= 1024\text{GB} \end{aligned}$$

1.2.2 数据的二进制表示方法

数据在计算机的内存中是用二进制 (Binary) 形式存储的。为什么要用二进制形式存储数据呢？人们习惯于用十进制，逢十进一。其实这完全是因为人们的习惯和计量的需要，而并非天经地义的。事实上，日常生活中也存在着其他一些常用进位制度，如七进制（每星期七天）、六十进制（一小时等于六十分，一分等于六十秒）、十二进制（一打等于十二个，一英尺等于十二英寸，一年等于十二个月）等。也有用二进制的，如鞋、袜、手套、筷子等，都是逢二进一。在计算机语言中除十进制外，常用到的还有二进制、八进制、十六进制。

在十进制 (Decimal) 数中，数值由 0、1、2、3、4、5、6、7、8、9 十个数字符号表示；基数为 10；按逢 10 进 1、借 1 算 10 的规则计数；采用位置记数法（或带权记数法）。所谓基数，就是在某进位制度下用到的数字个数。所谓权，就是幂次。

所有进位制度的数在各位上的权的定义：小数点左边，从右向左分别是基数的 0 次方，1 次方，…，n-1 次方，小数点右边，从左向右分别是基数的 -1 次方，-2 次方，…，-n 次