

 PYTHON

人生苦短，我学 Python

掌握 Python 语言，拥有完美人生

# Python

## 程序设计

陈春晖 翁 恺 季江民 ◆ 编著

本书配套教学网站：<https://pintia.cn/>

  
浙江大学出版社  
ZHEJIANG UNIVERSITY PRESS

# Python 程序设计

陈春晖 翁 恺 季江民 编著



ZHEJIANG UNIVERSITY PRESS  
浙江大学出版社

## 图书在版编目(CIP)数据

Python 程序设计 / 陈春晖, 翁恺, 季江民编著. —  
杭州: 浙江大学出版社, 2019. 1

ISBN 978-7-308-18977-4

I. ①P… II. ①陈…②翁…③季… III. ①软件工  
具—程序设计—高等学校—教材 IV. ①TP311.561

中国版本图书馆 CIP 数据核字 (2019) 第 028536 号

## Python 程序设计

陈春晖 翁 恺 季江民 编著

---

策 划 黄娟琴

责任编辑 王元新 黄娟琴

责任校对 刘 郡 汪志强

封面设计 程 晨 黄小意

出版发行 浙江大学出版社

(杭州市天目山路 148 号 邮政编码 310007)

(网址: <http://www.zjupress.com>)

排 版 杭州中大图文设计有限公司

印 刷 杭州杭新印务有限公司

开 本 787mm×1092mm 1/16

印 张 16.25

字 数 375 千

版 印 次 2019 年 1 月第 1 版 2019 年 1 月第 1 次印刷

书 号 ISBN 978-7-308-18977-4

定 价 45.00 元

---

版权所有 翻印必究 印装差错 负责调换

浙江大学出版社市场运营中心联系方式: 0571-88925591; <http://zjdxcbbs.tmall.com>

# 前言

为了适应信息技术的发展,切实满足社会各个领域对计算机应用人才不断增长的需求,本教材设计了“Python 程序设计”通识课程教学方案,力求融入计算思维的思想,并将多年教学实践所形成的解决实际问题的思维模式和方法渗透到整个教学过程中。与传统的程序设计类教材不同,本教材在介绍程序设计的基本技能外,还着重介绍分析问题和解决问题的方法与思路,通过构建典型案例,为学生在未来利用 Python 程序设计语言解决各自专业中遇到的实际问题打下良好的基础。

本书具有以下特点:

## 1. 注重与中学内容的衔接

本教材的内容编排凝聚了作者多年的教学经验与体会,如第 2 章就引入列表解析并用于求和编程。新生很熟悉的求和形式是:

$$1+2+3+\cdots+100=\sum_{i=1}^{100}i$$

$\sum$  对应于 sum 函数, $\sum i$  就是  $\text{sum}(i)$ , $\sum_{i=1}^{100}$  对应  $\text{for } i \text{ in range}(1,101)$ ,合起来就是:

```
sum(i for i in range(1,101))
```

程序完成了? 是的! 这样编程简单明了,非常容易学习,还符合 Python 的编程习惯。这种情况在本教材中有不少,读者可自己研究。

## 2. 完善的配套练习

本教材配套的教学网站是 PTA(Programming Teaching Assistant),配套的练习都可在该网站上进行。PTA 官网网址:<https://pintia.cn/>,教师账号申请可与陈越老师联系,邮箱:chenyue@zju.edu.cn。

## 3. 精心选择第三方模块教学

Python 有十几万个官方认可的第三方模块,同一功能可能有好几个模块都

能完成,因此选择合适的模块进行教学就很重要。本教材尽量选择成熟、使用方便、功能强大的模块进行教学。网络爬虫教学通常会使用“requests”和“beautifulsoup”两个模块完成,但本教材选择了“requests-html”模块(与“requests”模块是同一作者),既方便教学,又能让学生掌握网络爬虫程序的更多技能。

#### 4. 教学内容符合社会实际需求

具有开发 Web 应用程序的能力是社会的普遍需求。本教材重点介绍了 Python 的 Web 应用程序开发,即用一个实例展示开发的整个过程。学生被要求结合本专业的实际情况,完成相应的 Web 应用程序,并部署到云端。从实际的教学效果看,各专业的同学都能完成,学生成就感满满!

本教材的教学内容是为 64 学时设计的,每周 4 节课,2 节理论教学,2 节上机实践。

如果教学安排是 32 学时,可选择完成第 1 至第 6 章和 7.1 节的教学内容。

本教材第 1、2、6、7、9 章和附录由陈春晖老师执笔,第 3、4、5 章由翁恺老师执笔,第 8 章由季江民老师执笔。全书由陈春晖老师统稿。

本教材是在许端清教授的大力支持下完成的,在教材的编写过程中,方宁老师提出了不少宝贵意见,陈越老师为教学工作提供了优质的教学实践平台,在此一并表示感谢!

由于时间仓促和作者水平有限,书中难免有不妥之处,恳请广大读者批评指正。

本书的配套课件、源代码等请联系责任编辑(邮箱:13588451163@163.com)。关于本教材及课件使用中的问题,请发邮件至 cchui@zju.edu.cn。

作者  
2019 年 1 月

# 目 录

<b>第 1 章 Python 语言概述</b> .....	1
1.1 计算机基础 .....	1
1.1.1 计算机特点 .....	1
1.1.2 计算机常用的数制及编码 .....	2
1.1.3 计算机系统组成 .....	5
1.1.4 操作系统 .....	7
1.1.5 程序设计语言 .....	8
1.2 Python 语言简介 .....	9
1.3 Python IDLE 开发环境 .....	11
1.3.1 Python IDLE 开发环境安装 .....	11
1.3.2 运行 Python 程序 .....	13
1.4 标识符和变量 .....	15
1.4.1 标识符和关键字 .....	15
1.4.2 常量和变量 .....	16
1.5 输入及输出函数 .....	17
1.5.1 输入函数 .....	17
1.5.2 输出函数 .....	18
本章小结 .....	20
习 题 .....	21
<b>第 2 章 用 Python 语言编写程序</b> .....	23
2.1 数字类型 .....	23
2.1.1 整数 .....	23
2.1.2 浮点数 .....	26
2.1.3 复数 .....	27

2.1.4 数学库(math 库)的使用 .....	28
2.2 字符串 .....	29
2.3 布尔值、空值和列表 .....	32
2.3.1 布尔值 .....	32
2.3.2 列表 .....	35
2.4 内置转换函数 .....	36
2.5 语句 .....	39
2.5.1 赋值语句 .....	39
2.5.2 if 语句 .....	41
2.5.3 for 语句 .....	42
2.5.4 列表推导式 .....	44
2.6 格式化输出 .....	46
本章小结 .....	48
习 题 .....	48
<b>第 3 章 使用字符串、列表和元组 .....</b>	<b>51</b>
3.1 序列的访问及运算符 .....	51
3.1.1 什么是序列 .....	51
3.1.2 通用序列操作 .....	51
3.1.3 序列的运算符 .....	55
3.1.4 计算序列的长度和最大值 .....	56
3.2 字符串的使用 .....	57
3.2.1 什么是字符串 .....	57
3.2.2 字符串常用方法或函数 .....	59
3.2.3 将数字转换成字符串 .....	62
3.3 列表和元组使用 .....	66
3.3.1 列表 .....	66
3.3.2 基本的列表操作 .....	67
3.3.3 列表的函数或方法 .....	69
3.3.4 字符串和列表的互相操作 .....	71
3.3.5 元组 .....	73
3.4 随机函数库(random 库) .....	74
本章小结 .....	77
习 题 .....	77

<b>第 4 章 条件、循环和其他语句</b> .....	79
4.1 条件语句 .....	79
4.1.1 基本的条件语句 .....	79
4.1.2 有分支的条件语句 .....	81
4.1.3 嵌套的条件语句 .....	83
4.1.4 连缀的 if-elif-else .....	84
4.1.5 条件表达式 .....	85
4.2 while 循环 .....	86
4.2.1 while 循环 .....	86
4.2.2 循环内的控制 .....	90
4.3 for 循环 .....	93
4.3.1 for...in 循环 .....	93
4.3.2 range()函数 .....	94
4.4 搜索和排序 .....	96
4.4.1 线性搜索 .....	96
4.4.2 搜索最值 .....	97
4.4.3 二分搜索 .....	97
4.4.4 选择排序 .....	99
4.4.5 冒泡排序 .....	101
4.5 异常处理 .....	102
本章小结 .....	106
习 题 .....	107
<b>第 5 章 集合和字典</b> .....	110
5.1 集合 .....	111
5.1.1 创建集合 .....	111
5.1.2 操作和访问集合的元素 .....	111
5.1.3 元素、子集、超集和相等判断 .....	112
5.1.4 集合运算 .....	113
5.2 字典 .....	114
5.2.1 创建字典 .....	114
5.2.2 字典的基本运算 .....	115
5.2.3 字典方法或函数 .....	116
5.3 集合和字典的应用 .....	117
5.3.1 数据结构 .....	118



5.3.2 过程 .....	119
本章小结 .....	121
习 题 .....	121
<b>第 6 章 函 数</b> .....	<b>123</b>
6.1 函数的定义和调用 .....	123
6.2 函数参数 .....	126
6.2.1 位置参数 .....	126
6.2.2 关键字参数 .....	126
6.2.3 默认值参数 .....	127
6.2.4 不定长数目参数 .....	130
6.3 函数的返回值 .....	135
6.4 命名空间和作用域 .....	136
6.5 递归 .....	137
6.6 内置函数 .....	139
6.6.1 sorted() 函数 .....	139
6.6.2 map() 函数 .....	140
6.6.3 zip() 函数 .....	140
6.6.4 eval() 和 exec() 函数 .....	141
6.6.5 all() 和 any() 函数 .....	141
6.7 程序结构 .....	142
6.7.1 模块和包 .....	142
6.7.2 sys 模块 .....	143
本章小结 .....	145
习 题 .....	146
<b>第 7 章 文 件</b> .....	<b>150</b>
7.1 文件读写 .....	150
7.2 用 Pandas 模块读写常见格式文件 .....	156
7.2.1 Python 第三方模块的安装 .....	156
7.2.2 Pandas 和 Plotly 模块 .....	158
7.2.3 用 Pandas 读写各种类型的文件 .....	163
7.2.4 JSON 文件读写 .....	167
7.3 数据可视化 .....	169
本章小结 .....	172
习 题 .....	173

<b>第 8 章 类和对象</b> .....	175
8.1 类和对象的概念 .....	175
8.2 类和对象的创建 .....	176
8.2.1 定义类 .....	176
8.2.2 创建对象 .....	177
8.2.3 访问对象成员 .....	177
8.2.4 属性值 .....	178
8.3 使用对象编写程序 .....	180
8.4 封装 .....	187
8.4.1 类成员 .....	187
8.4.2 私有成员与公有成员 .....	190
8.5 继承和多态 .....	191
8.5.1 继承 .....	191
8.5.2 多态 .....	192
本章小结 .....	192
习 题 .....	193
<b>第 9 章 Web 应用程序开发及网络爬虫</b> .....	199
9.1 Web 应用程序开发概述 .....	199
9.1.1 Web 应用程序运行原理 .....	199
9.1.2 超文本标记语言简介 .....	201
9.1.3 层叠样式表 .....	203
9.2 Web 应用框架 Flask .....	207
9.3 云上部署 Web 应用程序 .....	219
9.4 网络爬虫 .....	232
9.4.1 获取网页 .....	232
9.4.2 获取元素 .....	234
本章小结 .....	238
习 题 .....	238
<b>参考文献</b> .....	239
<b>附录 A Python 语言简明参考手册</b> .....	240
A.1 导引 .....	240
A.2 词法分析 .....	241

A.3	数据模型 .....	243
A.4	表达式 .....	244
A.5	语句 .....	245
A.6	顶层组件 .....	247
<b>附录 B</b>	<b>PTA 平台常见问题解答 .....</b>	<b>249</b>
B.1	评分 .....	249
B.2	常见问题 .....	250

# 第 1 章 Python 语言概述

## 1.1 计算机基础

### 1.1.1 计算机特点

自从 1946 年第一台电子计算机问世以来,计算机科学与技术已成为近代发展速度最快的一门学科,尤其是微型计算机的出现和计算机网络的发展,使计算机的应用渗透到社会的各个领域,有力地推动了信息社会的发展。计算机作为一种通用的信息处理工具,它具有极快的处理速度、很强的存储能力、精确的计算和逻辑判断能力。其主要特点如下。

#### 1. 运算速度快

当今计算机系统的运算速度已达到每秒几十亿亿次,微机也可达每秒亿次以上,使大量复杂的科学计算问题得以解决。例如,导弹轨道的计算、大型水坝的计算、长期天气预报的计算等,过去人工计算需要几年、几十年的时间,而现在使用计算机进行计算只需几天甚至几分钟的时间。

#### 2. 计算精确度高

科学技术的发展特别是尖端科学技术的发展,需要高度精确的计算。计算机控制的导弹之所以能准确地击中预定的目标,是与计算机的精确计算分不开的。一般计算机可以有几十位甚至几百位(二进制)有效数字,计算精度可由千分之几到亿分之几,这是任何其他计算工具所望尘莫及的。

#### 3. 具有存储和逻辑判断能力

随着计算机存储容量的不断增大,可存储记忆的信息越来越多。计算机不仅能进行计算,而且能把参加运算的数据、程序以及中间结果和最后结果保存起来,以供用户随时调用;还可以对各种信息(如语言、文字、图形、图像、音乐等)通过编码技术进行算术运算和逻辑运算,甚至进行推理和证明。

#### 4. 具有自动控制能力

计算机内部操作是根据人们事先编好的程序自动控制进行的。用户根据解题需要,事先设计好运行步骤与程序,计算机十分严格地按程序规定的步骤操作,整个过程无须人工干预。

### 1.1.2 计算机常用的数制及编码

数制也称计数制,是指用一组固定的符号和统一的规则来表示数值的方法。编码是采用少量的基本符号,选用一定的组合原则,以表示大量复杂多样的信息的技术。计算机是信息处理的工具,任何信息必须转换成二进制数据后才能由计算机进行处理、存储和传输。

#### 1. 二进制数

我们习惯使用的十进制数由 0、1、2、3、4、5、6、7、8、9 十个不同的符号组成,每一个符号处于十进制数中不同的位置时,它所代表的实际数值是不一样的。例如 1999 可表示成:

$$1 \times 1000 + 9 \times 100 + 9 \times 10 + 9 \times 1 = 1 \times 10^3 + 9 \times 10^2 + 9 \times 10^1 + 9 \times 10^0$$

式中:每个数字符号的位置不同,它所代表的数值也不同,这就是经常所说的个位、十位、百位、千位的意思。二进制数和十进制数一样,也是一种进位计数制,但它的基数是 2。二进制数中 0 和 1 的位置不同,它所代表的数值也不同。例如二进制数 1101 表示十进制数 13:

$$(1101)_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 8 + 4 + 0 + 1 = 13$$

一个二进制数具有下列两个基本特点:

- (1)有两个不同的数字符号,即 0 和 1。
- (2)逢二进一。

一般我们用<sub>角标</sub>表示不同进制的数。例如,十进制数用<sub>10</sub>表示,二进制数用<sub>2</sub>表示。

#### 2. 二进制数制与其他数制

数位是指数码在一个数中所处的位置。基数是指在某种进位计数制中,每个数位上所能使用的数码的个数。例如,二进制数基数是 2,每个数位上所能使用的数码为 0 和 1 两个数码。在数制中有一个规则,如果是 N 进制数,必须是逢 N 进 1。下面主要介绍与计算机有关的常用的几种计数制。

##### (1)十进制(十进位计数制)

十进制数具有十个不同的数码符号 0、1、2、3、4、5、6、7、8、9,其基数为 10,特点是逢十进一,例如:

$$(1011)_{10} = 1 \times 10^3 + 0 \times 10^2 + 1 \times 10^1 + 1 \times 10^0$$

##### (2)八进制(八进位计数制)

八进制数具有八个不同的数码符号 0、1、2、3、4、5、6、7,其基数为 8,特点是逢八进

一,例如:

$$(1011)_8 = 1 \times 8^3 + 0 \times 8^2 + 1 \times 8^1 + 1 \times 8^0 = (521)_{10}$$

(3)十六进制(十六进制计数制)

十六进制数具有十六个不同的数码符号 0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F(或小写 a、b、c、d、e、f),其基数为 16,特点是逢十六进一,例如:

$$(1011)_{16} = 1 \times 16^3 + 0 \times 16^2 + 1 \times 16^1 + 1 \times 16^0 = (4113)_{10}$$

如表 1-1 所示为 4 位二进制数的不同进制对照表。

表 1-1 4 位二进制数的不同进制对照表

二进制	十进制	八进制	十六进制	二进制	十进制	八进制	十六进制
0000	0	0	0	1000	8	10	8
0001	1	1	1	1001	9	11	9
0010	2	2	2	1010	10	12	A
0011	3	3	3	1011	11	13	B
0100	4	4	4	1100	12	14	C
0101	5	5	5	1101	13	15	D
0110	6	6	6	1110	14	16	E
0111	7	7	7	1111	15	17	F

### 3. ASCII 码

计算机中,对非数值的文字和其他符号进行处理时,要对文字和符号进行数字化处理,即用二进制编码来表示文字和符号。字符编码(Character Code)是用二进制编码来表示字母、数字以及专门符号。在计算机系统中,目前普遍采用的是 ASCII(American Standard Code for Information Interchange)码,即美国信息交换标准代码(见表 1-2)。ASCII 码有 7 位版本和 8 位版本两种,国际上通用的是 7 位版本,7 位版本的 ASCII 码有 128 个元素,只需用 7 个二进制位( $2^7 = 128$ )表示,其中控制字符 34 个,阿拉伯数字 10 个,大小写英文字母 52 个,各种标点符号和运算符号 32 个。在计算机中实际用 8 位表示一个字符,最高位为“0”。

表 1-2 ASCII 码

ASCII 值	字符	ASCII 值	字符	ASCII 值	字符	ASCII 值	字符
0	NUL	32	(space)	64	@	96	,
1	SOH	33	!	65	A	97	a
2	STX	34	"	66	B	98	b
3	ETX	35	#	67	C	99	c
4	EOT	36	\$	68	D	100	d

续表

ASCII 值	字符	ASCII 值	字符	ASCII 值	字符	ASCII 值	字符
5	ENQ	37	%	69	E	101	e
6	ACK	38	&	70	F	102	f
7	BEL	39	,	71	G	103	g
8	BS	40	(	72	H	104	h
9	HT	41	)	73	I	105	i
10	LF	42	*	74	J	106	j
11	VT	43	+	75	K	107	k
12	FF	44	,	76	L	108	l
13	CR	45	-	77	M	109	m
14	SO	46	.	78	N	110	n
15	SI	47	/	79	O	111	o
16	DLE	48	0	80	P	112	p
17	DC1	49	1	81	Q	113	q
18	DC2	50	2	82	R	114	r
19	DC3	51	3	83	S	115	s
20	DC4	52	4	84	T	116	t
21	NAK	53	5	85	U	117	u
22	SYN	54	6	86	V	118	v
23	TB	55	7	87	W	119	w
24	CAN	56	8	88	X	120	x
25	EM	57	9	89	Y	121	y
26	SUB	58	:	90	Z	122	z
27	ESC	59	;	91	[	123	{
28	FS	60	<	92	/	124	
29	GS	61	=	93	]	125	}
30	RS	62	>	94	^	126	~
31	US	63	?	95	_	127	DEL

#### 4. Unicode 编码和 UTF-8 编码

Unicode 编码(统一码、万国码、单一码)是计算机科学领域里的一项业界标准,包括字符集、编码方案等。Unicode 编码是为了解决传统的字符编码方案的局限而产生的,它为每种语言中的每个字符设定了统一且唯一的二进制编码,以满足跨语言、跨平台进

行文本转换与处理的要求。Unicode 编码通常用两个字节表示一个字符,原有的英文编码从单字节变成双字节,只需要把高字节全部填为 0 就可以了。

在 Unicode 编码中,“汉字”这两个字对应的 Unicode 编码是\u6c49\u5b57,“\u”表示 Unicode 编码。

在 Unicode 编码中,我们有很多方式将数字 23389(孝)表示成程序中的数据,包括 UTF-8、UTF-16、UTF-32。UTF 是“Unicode Transformation Format”的缩写,可以翻译成 Unicode 字符集转换格式,即怎样将 Unicode 定义的数字转换成程序数据。UTF-8 格式用得最多,其以字节为单位对 Unicode 字符进行编码。

UTF-8 的特点是对不同范围的字符使用不同长度的编码。对于 0x00~0x7F 的字符,UTF-8 编码与 ASCII 编码完全相同。UTF-8 编码的最大长度是 6 个字节。Python 3 中的字符串是 Unicode 字符串而不是字节数组,这是与 Python 2 的主要差别之一。

### 1.1.3 计算机系统组成

完整的计算机系统包括硬件和软件两大部分。所谓硬件,是指构成计算机的实体设备,即由物理器件构成的具有输入/输出、存储、计算、控制的实体部件。广义地说,软件是指系统中的程序以及所有相关文档的集合。我们平时讲到的“计算机”一词,都是指含有硬件和软件的计算机系统,其结构如图 1-1 所示。

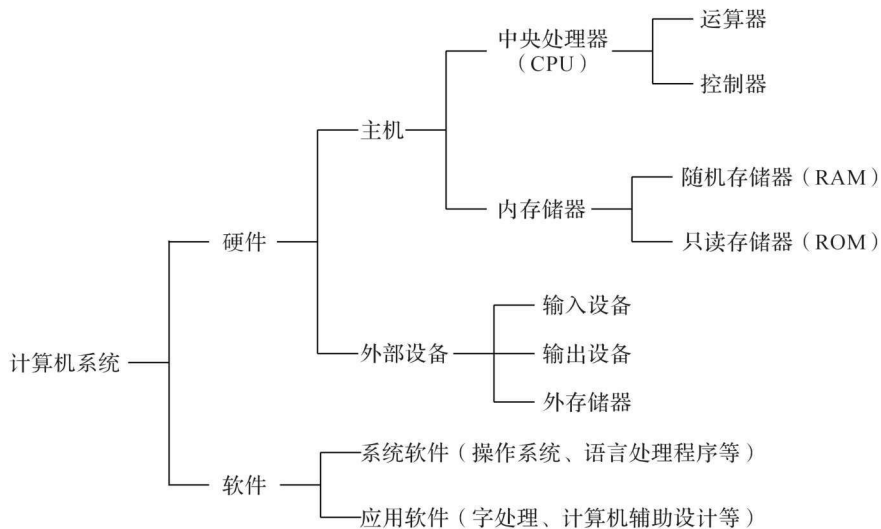


图 1-1 计算机系统组成

计算机硬件系统由运算器、控制器、存储器、输入设备/输出设备五个基本部分组成,也称计算机的五大部件。

运算器又称算术逻辑单元(Arithmetic Logic Unit, ALU),是计算机对数据进行运算处理的实体,它的主要功能是对二进制编码进行算术运算及基本逻辑运算,实现逻辑判断。运算器在控制器的控制下实现其功能,运算结果由控制器指挥送到内存储器中。



控制器主要由指令寄存器、译码器、程序计数器和操作控制器等组成。控制器用来控制计算机各部件协调工作,并使整个处理过程有条不紊地进行。它的基本功能就是从内存储器中取指令和执行指令,即控制器按程序计数器指出的指令地址从内存储器中取出该指令进行译码,然后翻译该指令,向有关部件发出控制命令,执行该指令。另外,控制器在工作过程中,还要接收各部件反馈回来的信息。

存储器具有记忆功能,用来保存信息,如数据、指令和运算结果等。存储器可分为两种:内存储器与外存储器。内存储器直接与 CPU 相连接,存储容量较小,但速度快,用来存放当前运行程序的指令和数据,并直接与 CPU 交换信息。内存储器由许多存储单元组成,每个单元能存放一个二进制数。存储器的存储容量以字节为基本单位,每个字节都有自己的编号,称为“地址”,如果要访问存储器中的某个信息,就必须知道它的地址,然后再按地址存入或取出信息。如果地址线是 8 位,则可以标识 256 单元,如表 1-3 所示。

表 1-3 存储器地址和单元内容

二进制内存储器单元地址	单元内容
00000000	01010001
00000001	11001110
00000010	00110011
...	...
...	...
...	...
11111110	10010000
11111111	01100010

为了度量信息存储容量,将 8 位二进制码(8bits)称为一个字节(Byte,简称 B)。字节是计算机中数据处理和存储容量的基本单位。1024 字节称为 1K 字节(1KB),1024K 个字节称 1 兆字节(1MB),1024M 个字节称为 1G 字节(1GB),1024G 个字节称为 1T 字节(1TB)。现在微型计算机主存容量大多数在几个 G 字节以上。

计算机处理数据时,一次可以运算的数据长度称为一个“字”(Word)。字的长度称为字长。一个字可以是一个字节,也可以是多个字节。常用的字长有 8 位、16 位、32 位、64 位等。如某一类计算机的字由 4 个字节组成,则字的长度为 32 位,相应的计算机称为 32 位机。

外存储器又称辅助存储器(简称辅存),它是内存储器的扩充。外存储器容量大,价格低,但存储速度较慢,一般用来存放大量暂时不用的程序、数据和中间结果,需要时,可成批地和内存储器进行信息交换。外存储器只能与内存储器交换信息,不能被计算机系统其他部件直接访问。常用的外存储器有硬盘、磁带、光盘等。

输入/输出设备简称 I/O(Input/Output)设备。用户通过输入设备将程序和数据输入计算机,输出设备将计算机处理的结果显示或打印出来。