



21世纪高等学校计算机
专业实用规划教材

Python程序设计 与算法基础教程

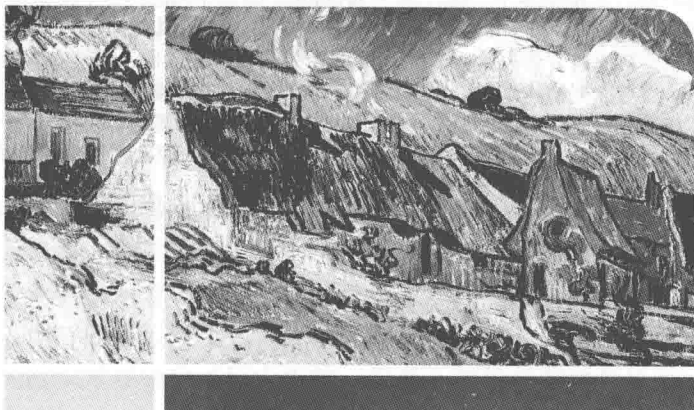
© 江红 余青松 主编



清华大学出版社



21世纪高等学校计算机
专业实用规划教材



Python程序设计 与算法基础教程

◎ 江红 余青松 主编



清华大学出版社
北京

内 容 简 介

本教程集教材、练习册、上机指导于一体，基于 Windows 10 和 Python 3.5.2 构建 Python 开发平台，阐述 Python 语言的基础知识，以及使用 Python 语言的实际开发应用实例，具体内容包括：Python 概述、Python 语言基础、程序流程控制、常用内置数据类型、系列数据类型、输入和输出、错误和异常处理、函数、类和对象、模块和客户端、算法与数据结构基础、图形用户界面、图形绘制、数值日期和时间处理、字符串和文本处理、文件、数据库访问、网络和 Web 编程、多线程编程以及系统管理等。

本教程作者结合多年的程序设计、系统开发以及授课经验，由浅入深、循序渐进地介绍 Python 程序设计语言，让读者能够较为系统全面地掌握程序设计的理论和应用。

本教程可以作为高等学校各专业的计算机程序设计教材，同时也可作为广大程序设计开发者、爱好者的自学参考书。

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 13701121933

图书在版编目 (CIP) 数据

Python 程序设计与算法基础教程/江红，余青松主编. —北京：清华大学出版社，2017
(21 世纪高等学校计算机专业实用规划教材)

ISBN 978-7-302-46683-3

I. ①P… II. ①江… ②余… III. ①软件工具—程序设计—教材 IV. ①TP311.561

中国版本图书馆 CIP 数据核字 (2017) 第 035909 号

责任编辑：魏江江 薛 阳

封面设计：刘 键

责任校对：时翠兰

责任印制：王静怡

出版发行：清华大学出版社

网 址：<http://www.tup.com.cn>, <http://www.wqbook.com>

地 址：北京清华大学学研大厦 A 座 邮 编：100084

社总机：010-62770175 邮 购：010-62786544

投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈：010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载：<http://www.tup.com.cn>, 010-62795954

印 装 者：北京鑫海金澳胶印有限公司

经 销：全国新华书店

开 本：185mm×260mm 印 张：26.75 字 数：648 千字

版 次：2017 年 7 月第 1 版 印 次：2017 年 7 月第 1 次印刷

印 数：1~2000

定 价：59.00 元

产品编号：070658-01

目 录

第 1 章 Python 概述	1
1.1 Python 语言概述	1
1.1.1 Python 语言简介	1
1.1.2 Python 语言的特点	1
1.1.3 Python 语言的应用范围	2
1.2 Python 语言版本和开发环境	2
1.2.1 Python 语言的版本	2
1.2.2 Python 语言的实现	2
1.2.3 Python 语言的集成开发环境	3
1.3 下载和安装 Python	3
1.3.1 下载 Python	3
1.3.2 安装 Python	4
1.3.3 安装和管理 Python 扩展包	4
1.4 使用 Python 解释器解释执行 Python 程序	6
1.4.1 运行 Python 解释器	6
1.4.2 运行 Python 集成开发环境	7
1.5 使用文本编辑器和命令行编写和执行 Python 源文件程序	8
1.5.1 编写 Hello World 程序	9
1.5.2 Hello World 程序 (hello.py) 源代码分析	10
1.5.3 运行 Python 源代码程序	10
1.5.4 命令行参数	11
1.6 使用集成开发环境 IDLE 编写和执行 Python 源文件程序	12
1.6.1 使用 IDLE 编写程序	12
1.6.2 使用 IDLE 编辑程序	13
1.7 在线帮助和相关资源	13
1.7.1 Python 交互式帮助系统	13
1.7.2 Python 文档	16
1.7.3 Python 官网	17
1.7.4 Python 扩展库索引 (PyPI)	17
复习题	18
上机实践	19

第 2 章 Python 语言基础	20
2.1 Python 程序概述	20
2.1.1 引例	20
2.1.2 Python 程序构成	20
2.2 Python 对象和引用	21
2.2.1 Python 对象概述	21
2.2.2 使用字面量创建实例对象	21
2.2.3 使用类对象创建实例对象	22
2.2.4 数据类型	22
2.2.5 变量和对象的引用	22
2.2.6 Python 是动态类型语言	23
2.2.7 Python 是强类型语言	24
2.2.8 对象内存示意图	24
2.2.9 对象的值比较 (==) 和引用判别 (is)	25
2.2.10 不可变对象 (immutable) 和可变对象 (mutable)	25
2.3 标识符及其命名规则	26
2.3.1 标识符	26
2.3.2 保留关键字	27
2.3.3 Python 预定义标识符	27
2.3.4 命名规则	27
2.4 变量和赋值语句	28
2.4.1 变量的声明和赋值	28
2.4.2 链式赋值语句	28
2.4.3 复合赋值语句	29
2.4.4 删除变量	29
2.4.5 系列解包赋值	29
2.4.6 常量	30
2.5 表达式和运算符	30
2.5.1 表达式的组成	30
2.5.2 表达式的书写规则	31
2.5.3 运算符概述	31
2.5.4 Python 运算符及其优先级	31
2.6 语句	32
2.6.1 Python 语句	32
2.6.2 Python 语句的书写规则	33
2.6.3 复合语句及其缩进书写规则	33
2.6.4 注释语句	34
2.6.5 空语句 pass	34

2.7	函数和模块	34
2.7.1	函数的创建和调用	34
2.7.2	内置函数	35
2.7.3	模块函数	35
2.7.4	函数 API	36
2.8	类和对象	36
2.8.1	创建类对象	36
2.8.2	实例对象的创建和调用	37
2.9	模块和包	37
	复习题	37
	上机实践	39
第 3 章	程序流程控制	41
3.1	顺序结构	41
3.2	选择结构	41
3.2.1	分支结构的形式	42
3.2.2	单分支结构	42
3.2.3	双分支结构	43
3.2.4	多分支结构	44
3.2.5	if 语句的嵌套	45
3.2.6	if 语句典型示例代码	46
3.2.7	选择结构综合举例	47
3.3	循环结构	48
3.3.1	可迭代对象	48
3.3.2	range 对象	49
3.3.3	for 循环	49
3.3.4	while 循环	50
3.3.5	循环的嵌套	51
3.3.6	break 语句	52
3.3.7	continue 语句	53
3.3.8	死循环(无限循环)	54
3.3.9	else 子句	55
3.3.10	循环语句典型示例代码	55
3.3.11	循环结构综合举例	56
	复习题	57
	上机实践	60
第 4 章	常用内置数据类型	64
4.1	Python 内置数据类型概述	64

4.1.1	数值数据类型	64
4.1.2	序列数据类型	64
4.1.3	集合数据类型	65
4.1.4	字典数据类型	65
4.1.5	NoneType、NotImplementedType 和 EllipsisType	65
4.1.6	其他数据类型	65
4.2	int 数据类型（任意精度整数）	65
4.2.1	整型字面量	66
4.2.2	int 对象	66
4.2.3	int 对象的方法	66
4.2.4	整数的运算	67
4.3	float 类型（有限精度浮点数）	68
4.3.1	浮点类型字面量	68
4.3.2	float 对象	68
4.3.3	float 对象的方法	68
4.3.4	浮点数的运算	69
4.4	complex 类型（复数）	70
4.4.1	复数类型字面量	70
4.4.2	complex 对象	70
4.4.3	complex 对象属性和方法	70
4.4.4	复数的运算	70
4.5	bool 数据类型（布尔逻辑值）	71
4.5.1	布尔值字面量	71
4.5.2	bool 对象	71
4.5.3	逻辑运算符	72
4.6	str 数据类型（字符串）	73
4.6.1	字符串字面量	73
4.6.2	字符串编码	73
4.6.3	转义字符	73
4.6.4	str 对象	74
4.6.5	str 对象属性和方法	74
4.6.6	字符串的运算	75
4.6.7	对象转换为字符串	75
4.6.8	字符串的格式化	75
4.7	比较关系运算和条件表达式	76
4.7.1	条件表达式	76
4.7.2	关系和测试运算符	77
4.8	算术运算符和位运算符	78
4.8.1	算术运算符	78

4.8.2	位运算符	78
4.9	混合运算和数值类型转换	79
4.9.1	隐式转换	79
4.9.2	显式转换(强制转换)	79
4.10	内置标准数学函数	80
4.10.1	内置数学运算函数	80
4.10.2	数制转换函数	80
	复习题	81
	上机实践	84
第5章	系列数据类型	89
5.1	Python 系列数据概述	89
5.1.1	数组	89
5.1.2	系列数据类型	89
5.2	系列数据的基本操作	90
5.2.1	系列的长度、最大值、最小值、求和	90
5.2.2	系列的索引访问操作	90
5.2.3	系列的切片操作	91
5.2.4	系列的连接和重复操作	92
5.2.5	系列的成员关系操作	92
5.2.6	系列的比较运算操作	93
5.2.7	系列的排序操作	94
5.2.8	内置函数 all()和 any()	94
5.2.9	系列拆封	94
5.3	元组	95
5.3.1	使用元组字面量创建元组实例对象	95
5.3.2	使用 tuple 对象创建元组实例对象	96
5.3.3	元组的系列操作	96
5.4	列表	96
5.4.1	使用列表字面量创建列表实例对象	96
5.4.2	使用 list 对象创建元组实例对象	97
5.4.3	列表的系列操作	97
5.4.4	list 对象的方法	97
5.4.5	列表解析表达式	98
5.5	字符串	98
5.5.1	字符串的系列操作	98
5.5.2	字符串编码	99
5.5.3	字符串格式化	99
5.6	字节系列	102

5.6.1	bytes 常量	102
5.6.2	创建 bytes 对象	102
5.6.3	创建 bytearray 对象	103
5.6.4	bytes 和 bytearray 的系列操作	103
5.6.5	字节编码和解码	104
	复习题	104
	上机实践	106
第 6 章	输入和输出	108
6.1	输入和输出概述	108
6.2	命令行参数	108
6.2.1	sys.argv 与命令行参数	108
6.2.2	argparse 模块和命令行参数解析	109
6.3	标准输入和标准输出函数	110
6.3.1	输入和输出函数	110
6.3.2	交互式用户输入	111
6.3.3	运行时提示输入密码	112
6.4	文件和文件对象	112
6.4.1	文件对象和 open 函数	112
6.4.2	文件的打开、写入、读取和关闭	113
6.4.3	with 语句和上下文管理协议	113
6.5	标准输入、输出和错误流	114
6.5.1	标准输入、输出和错误流文件对象	114
6.5.2	读取任意长度的输入流	115
6.5.3	标准输入、输出和错误流重定向	115
6.6	重定向和管道	116
6.6.1	重定向标准输出到一个文件	117
6.6.2	重定向文件到标准输入	117
6.6.3	管道	118
6.6.4	过滤器	119
	复习题	121
	上机实践	121
第 7 章	错误和异常处理	123
7.1	程序的错误	123
7.1.1	语法错误	123
7.1.2	运行时错误	123
7.1.3	逻辑错误	124
7.2	异常处理	125

7.2.1	异常处理概述	125
7.2.2	内置的异常类	125
7.2.3	引发异常	127
7.2.4	捕获处理异常机制概述	128
7.2.5	Python 虚拟机捕获处理异常	128
7.2.6	使用 try...except...else...finally 语句捕获处理异常	128
7.2.7	捕获异常的顺序	129
7.2.8	finally 块和发生异常后的处理	130
7.2.9	自定义异常类	130
7.3	断言处理	131
7.3.1	断言处理概述	131
7.3.2	assert 语句和 AssertionError 类	131
7.3.3	启用/禁用断言	132
7.4	程序的基本调试方法	132
7.4.1	语法错误的调试	133
7.4.2	运行时错误的调试	133
7.4.3	逻辑错误的调试	134
	复习题	134
	上机实践	135
第 8 章	函数	137
8.1	函数概述	137
8.1.1	函数的基本概念	137
8.1.2	函数的功能	137
8.1.3	Python 函数分类	137
8.2	函数的声明和调用	138
8.2.1	函数对象的创建	138
8.2.2	函数的调用	139
8.2.3	作为对象的函数	140
8.2.4	Lambda 表达式和匿名函数	140
8.2.5	函数的副作用	141
8.3	参数的传递	141
8.3.1	形式参数和实际参数	141
8.3.2	形式参数变量和对象引用传递	142
8.3.3	传递不可变对象的引用	142
8.3.4	传递可变对象的引用	143
8.3.5	可选参数	143
8.3.6	位置参数和命名参数	144
8.3.7	可变参数	145

8.3.8	强制命名参数	145
8.3.9	参数类型检查	146
8.4	函数的返回值	146
8.4.1	return 语句和函数返回值	146
8.4.2	多条 return 语句	147
8.4.3	返回多个值	148
8.5	变量的作用域	148
8.5.1	全局变量	148
8.5.2	局部变量	149
8.5.3	全局语句 global	150
8.5.4	非局部语句 nonlocal	151
8.5.5	类成员变量	151
8.5.6	输出局部变量和全局变量	151
8.6	递归函数	152
8.6.1	递归函数的定义	152
8.6.2	递归函数的原理	153
8.6.3	递归函数需要注意的问题	154
8.6.4	递归函数的应用: 最大公约数	154
8.6.5	递归函数的应用: 汉诺塔	155
8.7	内置函数的使用	156
8.7.1	内置函数一览	156
8.7.2	eval 函数	156
8.7.3	exec 函数	157
8.7.4	compile 函数	157
	复习题	157
	上机实践	159
第 9 章	类和对象	161
9.1	面向对象概念	161
9.1.1	对象的定义	161
9.1.2	封装	161
9.1.3	继承	161
9.1.4	多态性	161
9.2	类对象和实例对象	162
9.2.1	类对象	162
9.2.2	实例对象	162
9.3	属性	163
9.3.1	实例属性	163
9.3.2	类属性	164

9.3.3	私有属性和公有属性	164
9.3.4	@property 装饰器	165
9.3.5	特殊属性	167
9.3.6	自定义属性	167
9.4	方法	168
9.4.1	实例方法	168
9.4.2	静态方法	169
9.4.3	类方法	170
9.4.4	__init__ 方法 (构造函数) 和 __new__ 方法	171
9.4.5	__del__ 方法 (析构函数)	172
9.4.6	私有方法与公有方法	173
9.4.7	方法重载	173
9.5	继承	174
9.5.1	派生类	174
9.5.2	查看继承的层次关系	175
9.5.3	类成员的继承和重写	176
9.6	对象的特殊方法	176
9.6.1	对象的特殊方法概述	176
9.6.2	运算符重载与对象的特殊方法	177
9.6.3	@functools.total_ordering 装饰器	179
9.6.4	__call__ 方法和可调用对象	179
9.7	对象的引用、浅拷贝和深拷贝	180
9.7.1	对象的引用	180
9.7.2	对象的浅拷贝	180
9.7.3	对象的深拷贝	181
	复习题	181
	上机实践	183
第 10 章	模块和客户端	185
10.1	模块化程序设计的概念	185
10.1.1	模块化程序设计	185
10.1.2	模块的 API	185
10.1.3	模块的实现	186
10.1.4	模块的客户端	187
10.1.5	模块化程序设计的优越性	187
10.2	模块的设计和实现	188
10.2.1	模块设计的一般原则	188
10.2.2	API 设计	188
10.2.3	创建模块	188

10.2.4	模块的私有函数	189
10.2.5	模块的测试代码	190
10.2.6	编写模块文档字符串	191
10.2.7	按字节编译的.pyc 文件	192
10.3	模块的导入和使用	192
10.3.1	导入模块和使用模块	192
10.3.2	导入模块中的成员	192
10.3.3	重新加载模块	193
10.3.4	动态代入模块	193
10.4	包	194
10.4.1	包的概念	194
10.4.2	创建包	194
10.4.3	包的导入和使用	195
10.5	模块的导入顺序	195
10.5.1	导入模块时的搜索顺序	195
10.5.2	模块搜索路径 sys.path	196
10.5.3	dir() 内置函数	197
10.6	命名空间与名称查找顺序	198
	复习题	198
	上机实践	199
第 11 章	算法与数据结构基础	200
11.1	算法及其性能分析	200
11.1.1	算法概述	200
11.1.2	算法的时间复杂度分析	200
11.1.3	增长量级	201
11.1.4	算法的空间复杂度分析	202
11.2	查找算法	202
11.2.1	顺序查找法	202
11.2.2	二分查找法	204
11.2.3	Python 语言提供的查找算法	205
11.3	排序算法	205
11.3.1	冒泡排序法	205
11.3.2	选择排序法	206
11.3.3	插入排序法	207
11.3.4	归并排序法	208
11.3.5	快速排序法	210
11.3.6	Python 语言提供的排序算法	211
11.4	常用数据结构	211

11.4.1	数据结构概述	211
11.4.2	常用的数据结构概述	212
11.4.3	Python 的 collections 模块	212
11.5	数组	213
11.5.1	列表和数组	213
11.5.2	array.array 对象和数组	213
11.6	栈和队列	214
11.6.1	栈的实现：使用列表	214
11.6.2	deque 对象	215
11.6.3	deque 作为栈	216
11.6.4	deque 作为队列	216
11.7	集合	217
11.7.1	集合的定义	217
11.7.2	判断集合元素是否存在	217
11.7.3	集合的运算：并集、交集、差集和对称差集	218
11.7.4	集合的比较运算：相等、子集和超集	218
11.7.5	集合的长度、最大值、最小值、元素和	219
11.7.6	可变集合的方法	219
11.8	字典（映射）	220
11.8.1	对象的哈希值	220
11.8.2	字典的定义	220
11.8.3	字典的访问操作	221
11.8.4	字典的视图对象	221
11.8.5	判断字典键是否存在	221
11.8.6	字典对象的长度和比较	221
11.8.7	字典对象的方法	222
11.8.8	defaultdict 对象	222
11.8.9	OrderedDict 对象	223
11.8.10	ChainMap 对象	223
11.8.11	Counter 对象	224
11.9	collections 模块的其他数据结构	226
11.9.1	namedtuple 对象	226
11.9.2	UserDict、UserList 和 UserString 对象	227
	复习题	227
	上机实践	230
第 12 章	图形用户界面	233
12.1	图形用户界面概述	233
12.1.1	tkinter	233

12.1.2	其他 GUI 库简介	233
12.2	tkinter 概述	234
12.2.1	tkinter 模块	234
12.2.2	图形用户界面构成	234
12.2.3	框架和 GUI 应用程序类	235
12.2.4	tkinter 主窗口	236
12.3	几何布局管理器	236
12.3.1	pack 几何布局管理器	237
12.3.2	grid 几何布局管理器	237
12.3.3	place 几何布局管理器	239
12.4	事件处理	240
12.4.1	事件类型	240
12.4.2	事件绑定	240
12.4.3	事件处理函数	241
12.5	常用组件	242
12.5.1	Label	242
12.5.2	LabelFrame	242
12.5.3	Button	243
12.5.4	Message	244
12.5.5	Entry	245
12.5.6	Text	245
12.5.7	Radiobutton	247
12.5.8	Checkbutton	247
12.5.9	Listbox	250
12.5.10	OptionMenu	251
12.5.11	Scale	253
12.5.12	Toplevel	254
12.5.13	ttk 子模块控件	254
12.6	对话框	255
12.6.1	通用消息对话框	255
12.6.2	文件对话框	256
12.6.3	颜色选择对话框	257
12.6.4	通用对话框应用举例	258
12.6.5	简单对话框	259
12.7	菜单和工具栏	260
12.7.1	创建主菜单	260
12.7.2	创建上下文菜单	262
12.7.3	菜单应用举例	263
	复习题	265

上机实践	266
第 13 章 图形绘制	268
13.1 Python 绘图模块概述	268
13.2 基于 tkinter 的图形绘制	269
13.2.1 基于 tkinter 画布绘图概述	269
13.2.2 创建画布对象	269
13.2.3 绘制矩形	270
13.2.4 绘制椭圆	270
13.2.5 绘制圆弧	271
13.2.6 绘制线条	271
13.2.7 绘制多边形	272
13.2.8 绘制字符串	273
13.2.9 应用举例：函数图形	273
13.3 基于 turtle 模块的海龟绘图	274
13.3.1 海龟绘图概述	274
13.3.2 turtle 模块概述	274
13.3.3 绘制正方形	275
13.3.4 绘制多边形	275
13.3.5 绘制圆形螺旋	276
13.3.6 递归图形	277
13.3.7 海龟绘图的应用实例	278
13.4 基于 Matplotlib 模块的绘图	279
13.4.1 Matplotlib 模块概述	279
13.4.2 安装 Matplotlib 模块	279
13.4.3 使用 Matplotlib 模块绘图概述	280
13.4.4 绘制函数曲线	280
13.4.5 绘制多个图形	281
13.4.6 绘制直方图	282
复习题	283
上机实践	284
第 14 章 数值日期和时间处理	286
14.1 相关模块概述	286
14.1.1 数值处理的相关模块	286
14.1.2 日期和时间处理的相关模块	286
14.2 math 模块和数学函数	286
14.2.1 math 模块的 API	286
14.2.2 math 模块应用举例	289

14.3	cmath 模块和复数数学函数	290
14.4	random 模块和随机函数	291
14.4.1	种子和随机状态	291
14.4.2	随机整数	292
14.4.3	随机系列	293
14.5	数值运算模块 NumPy	293
14.5.1	数值运算模块的基本使用	293
14.5.2	创建数组	294
14.5.3	处理数组	295
14.5.4	数组应用举例	295
14.6	日期和时间处理	295
14.6.1	相关术语	295
14.6.2	时间对象	296
14.6.3	测量程序运行时间	296
14.6.4	日期对象	297
14.6.5	获取当前日期时间	297
14.6.6	日期时间格式化为字符串	298
14.6.7	日期时间字符串解析为日期时间对象	298
	复习题	299
	上机实践	300
第 15 章	字符串和文本处理	303
15.1	相关模块概述	303
15.1.1	字符串和文本处理的相关模块	303
15.1.2	字符串处理概述的常用方法	303
15.2	字符串处理的常用操作	303
15.2.1	字符串的类型判断	303
15.2.2	字符串大小写转换	304
15.2.3	字符串的填充、空白和对齐	304
15.2.4	字符串的测试、查找和替换	305
15.2.5	字符串的拆分和组合	305
15.2.6	字符串的翻译和转换	306
15.2.7	字符串应用举例	306
15.3	正则表达式	307
15.3.1	正则表达式语言概述	307
15.3.2	正则表达式引擎	308
15.3.3	普通字符和转义字符	308
15.3.4	字符类和预定义字符类	309
15.3.5	边界匹配符	309