

学习Python——做个有编程能力的设计师



[学习Python——做个有编程能力的设计师 下载链接1](#)

著者:包瑞清

出版者:江苏凤凰科学技术出版社

出版时间:2015-6

装帧:平装

isbn:9787553745404

《学习PYTHON—做个有编程能力的设计师》不是纯粹的Python编程学习，而是在学习

基本Python编程语言时阐述辅助设计的方法，从Python Shell到RH_Python再到GH_Python，将Python真正作为设计者从事设计辅助的编程语言；同时，这又不是纯粹几何形式编程的说明，而是包括Python编程数据结构、基本语句、函数与类的学习过程。

在掌握Python语言的数据结构、基本语句、函数、类与异常，并通过具体案例阐述深入了解实践应用的方法之后，在“项目”部分探索与详细阐述了十个案例，包括：

“折叠的过程-圆柱体V形与Mesh顶点排序”，结合Kangaroo动力学模块模拟折叠的过程，使用Python编写符合建立Mesh格网输入条件的点组织模式；

“盒体的展开”，讲述了如何把一个盒体连续展开在一个平面上，并获取多个解的方法；

“解读蚁群算法与TSP问题以及在GH_Python中的实现”，蚁群算法是一种用来在图中寻找优化路径的机率型算法，灵感来源于蚂蚁在寻找食物过程中发现路径的行为。本部分详细阐述蚁群算法并探索将其在Grasshopper平台中的Python脚本中实现和解决TSP旅行商问题；

“最短路径与Dijkstra算法”，最短路径问题是图论研究中经典算法问题，旨在寻找图中两节点之间最短的路径，本部分使用Dijkstra算法在GH Python中实现；

“探索性研究适宜分析方法GH_Python下的实现”，适宜性分析的方法一般是在地理信息系统平台中实现，例如ArcGIS。适宜性分析在城乡规划和风景园林中经常被用到，根据应用的范畴又可以细分为城市建设用地的评价、环境影响评价、自然保护区及旅游区用地评价、区域规划和景观规划等。本次项目主要的研究目的是如何将适宜性分析在GH_Python中实现；

“使用递归解决八皇后问题与满足特定间距要求的建筑布局”，八皇后问题是一个以国际象棋为背景的问题：如何能够在 8×8 的国际象棋棋盘上放置八个皇后，使得任何一个皇后都无法直接吃掉其他的皇后。为了达到此目的，任两个皇后都不能处于同一条横行、纵行或斜线上；

“解读生命游戏_元胞自动机与建立生长模型”，元胞自动机(Cellular Automaton)，是一个时间和空间都离散的动力系统。散布在规则格网(Lattice Grid)中的每一元胞(Cell)取有限的离散状态，遵循同样的作用规则，依据确定的局部规则做同步更新。大量元胞通过简单的相互作用而构成动态系统的演化；

“GoogleEarth.KML文件的调入程序”，规划设计过程中经常借助于Google Earth来寻找和定位地标和路径，那么如何将地标和路径直接加载到Gasshopper平台中，是本部分需要解决的问题；

“解读粒子群(PSO)算法与建立点运动程序”，粒子群算法，也称粒子群优化算法 (Particle Swarm Optimization)，缩写为PSO，模拟鸟群的捕食行为，通过追随当前搜索到的最优值来寻找全局最优解

“城市土地利用结构信息熵”，在信息论中，熵是接收的每条消息中包含信息的平均量，又被称为信息熵、信源熵等。消息往往代表来自分布或数据流中的事件、样本或特征。在规划设计领域引入信息熵，研究土地利用结构，分析功能的融合程度，信息熵的高低可以反映城市土地利用的均衡程度，熵值越高，表明不同职能的土地利用类型数越多

。本部分案例主要通过Python 程序，阐述信息熵和均衡度公式计算过程。

作者介绍:

城市规划与设计专业博士，任教于西安建筑科技大学建筑学院，主攻计算机辅助设计应用研究，从作为设计者的角度寻找和融合编程与建筑、风景园林、城乡规划等设计规划学科之间的关联，建立面向设计师的编程设计知识系统，力图使编程设计成为设计领域的基础环节，探索未来设计的方法。

2012 年，作为caDesign 设计创始人之一，以“编程让设计更具创造力！”的信念为指引，分享研究成果，发展面向设计师的编程设计知识系统网络服务体系。

目录: 9 Python+PythonScript+GhPython

17 数据结构

18 1 列表(List)

19 1.1 索引

25 1.2 列表的基本操作

32 1.3 列表的方法

43 2 元组(Tuple)

43 3 字典(Dictionary)

44 3.1 Python 的字典与Grasshopper 的树型数据结构

48 3.2 Python 字典的方法

50 3.3 GhPython 与树型数据结构

66 4 字符串(String)

66 4.1 用Python 替代Grasshopper 处理字符串的方法

72 4.2 字符串格式化

84 4.3 re(regular expression) 正则表达式

95 基本语句

96 1 print() 与import

96 1.1 print()

97 1.2 使用import 导入模块或者函数

97 2 赋值的方法

98 3 循环语句

105 4 条件语句

117 函数

118 1 创建函数

136 2 随机模块

141 3 递归

155 4 时间模块

163 类

165 1 创建类

169 2 迭代器

172 3 生成器

177 异常

181 项目

183 Case_1: 折叠的过程- 圆柱体V 形与Mesh 顶点排序

191 Case_2: 盒体的展开

199 Case_3: 解读蚁群算法与TSP 问题以及在GH_Python 中的实现

211 Case_4: 最短路径与Dijkstra 算法

221 Case_5: 探索性研究适宜分析方法GH_Python 下的实现

251 Case_6: 使用递归解决八皇后问题与满足特定间距要求的建筑布局

257 Case_7: 解读生命游戏_元胞自动机与建立生长模型
265 Case_8: GoogleEarth.KML 文件的调入程序
276 Case_9: 解读粒子群(PSO) 算法与建立点运动程序
283 Case_10: 城市土地利用结构信息熵
· · · · · (收起)

[学习Python——做个有编程能力的设计师](#) [下载链接1](#)

标签

编程

python

数字化建筑

设计

grasshopper

Rhino

Python

数字化

评论

不知道本书的目标人群如何。非常不友好的框架和行文方法，根本无法读。写书并不是贴出大段代码就可以的。

一本数字化的工具书，书中有些错别字

越往后看基础要求越高，提高知识水平再来一遍

[学习Python——做个有编程能力的设计师](#) [下载链接1](#)

书评

[学习Python——做个有编程能力的设计师](#) [下载链接1](#)