# 课程编号：C0801207040

数据结构与算法

课程设计报告



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **姓名** | **李思宽** | **学号** | | **20175153** |
| **班级** | **软英1701** | **指导教师** | |  |
| **实验名称** | **数据结构与算法课程设计** | | | |
| **开设学期** | **2018-2019学年第2学期** | | | |
| **开设时间** | **第20周——第21周** | | | |
| **报告日期** | **2019.1.24** | | | |
| **评定成绩** |  | | **评定人** |  |
| **评定日期** | **2019-1-25** |

**东北大学软件学院**

第一章 系统分析

**（一）系统背景**

时下大多数人生活优越，信息业，交通业等行业的高速发展，带动了各地对旅游资源的开发。但景区管理仅靠人力管控较为困难，游客也难以得到较为舒适的服务。

为了使游客能够更加有效地掌握景区的相关信息，并方便管理者管控景区，开发、完成一个景区管理系统的实现，能够帮助景区管理者高效地对景区进行管理，同时该系统的实现及运用也能较好的满足游客们的具体游玩需求，让游客得到更加舒适智能的服务。

该系统将为游客及管理员提供全面准确的景区景点查询管理服务。

同时，该系统能还够提高学生对软件设计思路的理解以及进一步熟悉数据结构方面尤其是图论的算法。

1. **功能需求**

**管理员：**

* 对景点进行增删改查
* 对景点之间的道路进行增删改查
* 发布公告通知

**游客：**

* 查询景点的分布图
* 查询推荐的导游路线
* 查询两景点间的最短路径
* 根据景点名称或其描述进行关键字查询
* 景点根据受欢迎程度或景点岔路口数进行排序查询
* 景区停车场车辆进出信息

**（三）分析系统可能的解决方案**

此次数据结构课程设计的主要关键点及难点在于各种数据结构的实现及将其运用于景区管理系统当中。系统的主要难点在于数据结构的设计实现以及算法的选择、设计以及优化过程。经过深思熟虑地思考，查询资料及与指导教师和同学讨论后，针对各功能的实现给出以下的解决方案：

* **主界面**

系统运行后将进入主菜单界面，并在之后进行景点信息初始化工作，游客和管理员均可看到该界面并于此界面进行操作。该界面包括一个选项卡界面：

* 景区景点分布大地图选项卡
* 为游客推荐导游路线选项卡
* 查询两景点间的最短路径选项卡
* 景点查询及排序选项卡：
* 根据景点名称或其描述进行关键字查询
* 景点根据受欢迎程度或景点岔路口数进行排序查询
* 管理景区停车场车辆进出信息选项卡

还包括一个公告通知发布栏。

* **景点信息初始化**

景点信息需从景点信息文件中读入，

景点信息包括：

* 景点的名称
* 景点的描述
* 景点的欢迎度
* 有无休息区
* 有无公厕

将所有景点信息存入邻接链表之后，系统将链表信息加载到主界面。并可用于主界面中各选项卡界面。

* **管理员登录**

管理员登录可以通过主界面进行登录，登录时需要输入用户名和密码（为方便测试人员进行测试，在界面左上角有提示信息）。如果输入的账号或者密码错误，则会有错误信息的提示。如果验证通过，会进入相应的Administer界面。

* **管理员界面**

完成登录过程后可以进入管理员界面，并完成界面及景点信息初始化工作，只有管理员可看到该界面并于此界面进行操作。该界面同样包括一个选项卡界面：

* 景区景点插入选项卡
* 景区景点删除选项卡
* 两景点间的道路插入选项卡
* 两景点间的道路删除选项卡：
* 发布公告通知选项卡
* **发布通知**

设定一个专门的栏目，管理员有权限在该栏目上发布通告，同时， 在游客端可以看到管理员发布的消息，但是游客无法发布通告。

* **对景点进行增删改查**

当有新的景点需要加入景区的时候，管理员可以插入新的景点。管理员需要输入景点的名称，描述，欢迎度，有无休息区，有无公厕的信息。系统会对景点的名称进行查重，一旦有重名景点，系统将提示景点添加失败。

当一个景点不再对外界开放的时候，管理员可以删除该景点，同时也将删除与其相连的边。如果需要删除的景点不存在，系统会给出删除失败的提示。

当一个景点的信息需要被更新的时候，管理员可以更改景点的名称，描述等。如果未找到需要修改的景点，系统会提示修改失败。

当一个景点的信息需要被查询的时候，根据输入的关键字对景点信息进行匹配，找出相应的景点信息，如果景点不存在，返回景点不存在的提示。（位于主界面的景点查询及排序选项卡）

* **对景点之间的道路进行增删改查**

用一个邻接链表存储景点信息和道路信息，当对道路进行增删改查的时候，直接在邻接链表上进行操作即可更新道路信息。

* **管理景区停车场**

当有车辆到达的时候，管理员输入车牌号，将车的信息压入栈中。 当车位已满时，达到的车辆将进入等待区。

* **查询景点的分布图**

在主界面初始化后，读取邻接链表的信息，并进行解析，在用户界面处显示出相应的景点分布图。

* **查询推荐的导游路线**

使用汉密尔顿算法，欧拉算法，深度优先算法以及BFS、DFS相结合的BDFS创新算法等，用户可以对各个算法进行选择，然后可得到推荐的导游路线图，同时输出路径的长度，以便游客进行挑选及判断。

* **查询景点间的最短路径**

使用迪杰斯特拉算法,弗洛伊德算法，Bellmand-Ford 算法，SPFA 算法等，用户可以对各个算法进行选择，然后对两点间进行最短路径的计算并显示。

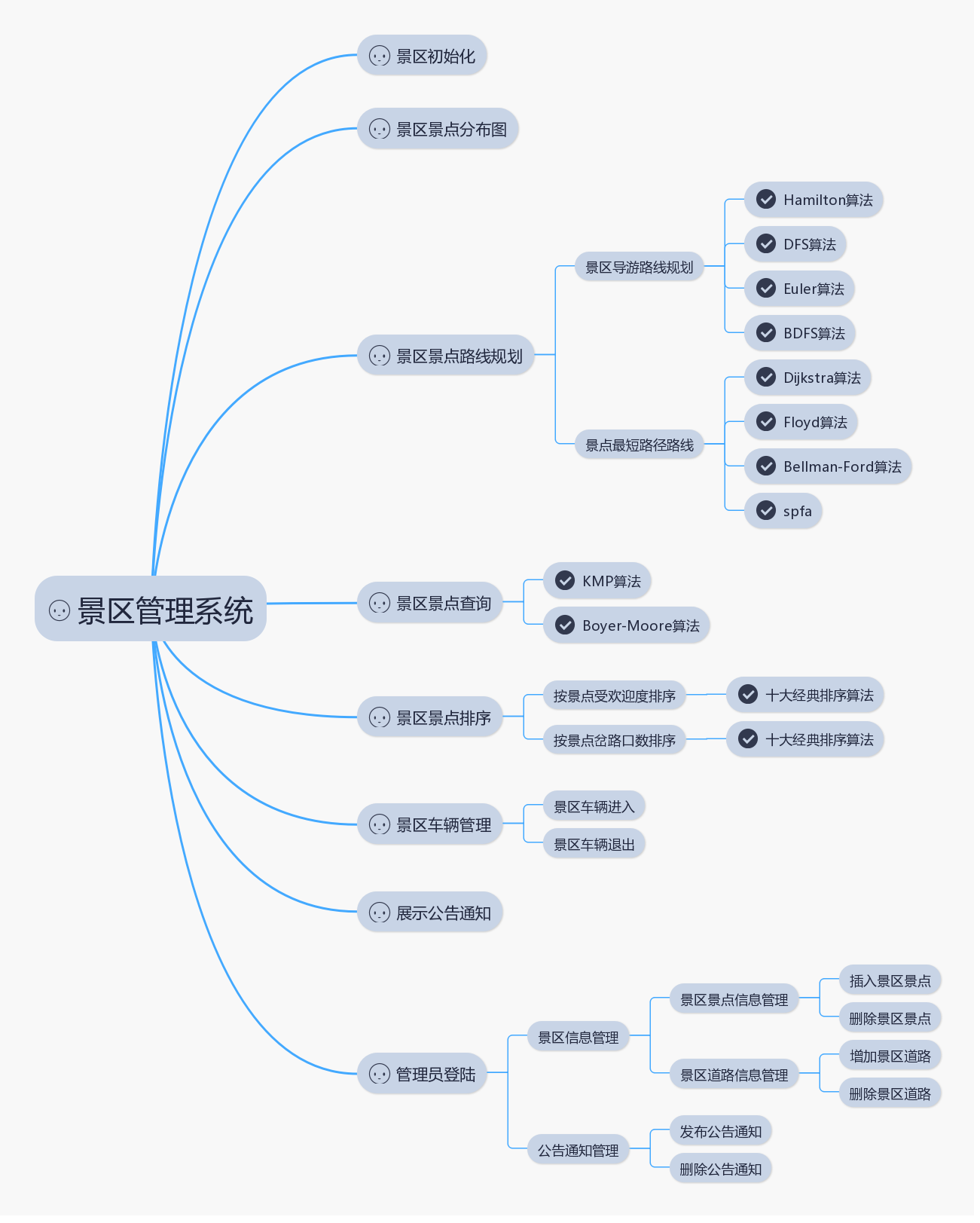
* **根据景点名称或其描述进行关键字查询**

使用 KMP, Boyer-Moore 等算法对用户输入的一段字符串进行子字符串的匹配，查找出与之相匹配对应的景点，在用户界面上显示各个匹配景点的按钮，用户可根据自己的意愿惊醒点击按钮，查看相应的景点信息。

* **景点排序查询**

使用希尔排序，桶排序等十种常用的排序方法对景点的相应属性（受欢迎程度，景点岔路口数）进行排序，用户可以对各个算法进行选择，然后将排序结果显示在用户界面上。

* **景区系统实现思维导图：**



**（四）个人工作和主要内容**

个人工作主要包含：

* **路线规划问题：**

实现使用常用的算法来进行路线规划，解决游客导游路线规划的困扰，使用了Hamilton算法，Euler算法， DFS 算法，此外，结合深度优先遍历和广度优先遍历的优点，进行了融合，来实现路线规划，能够达到较好的效果，简称为BDFS+, 关于BDFS+ 算法与以上算法的分析和比较将在后续部分详细阐述。

* **最短路径问题：**

实现使用Dijkstra算法, Floyd算法， Bellmand-Ford 算法，SPFA 算法四个方法进行求解，关于 四种算法的分析和比较会在后面的部分阐述。

* 用户可视化交互界面使用 JavaFX 框架进行设计。

第二章 系统设计

描述系统中自己的数据结构与算法的设计方案。

第三章 系统实现

针对自己所设计的方案，描述具体的实现开发内容。

第四章 系统测试

描述针对所开发的系统，如何展开测试的内容。列出在给定开发算法上的具体的实验分析内容，并分析实验结果。

第五章 结论

给出本次实验的结论。

参考文献

列出实验中所查阅的相关参考文献。

**附录：**

**《数据结构课程设计》实验成绩评定表**

****

|  |  |  |  |
| --- | --- | --- | --- |
| **评价内容** | **具 体 要 求** | **分值** | **得分** |
|
| **平时表现** | **课程设计过程中，无缺勤、迟到、早退现象，学习态度积极。能够主动查阅文献，积极分析系统中数据结构与算法的多种可能的设计方案，并认真地对所选择方案进行实现、测试、分析与总结。** | **20** |  |
| **分析与解决问题的能力** | **能够理解复杂数据结构及算法的设计思路和基本原理；能够应用所学数据结构与算法等相关知识和技能去解决实验系统中要求的各个题目；设计或实现思路有独特见解。** | **20** |  |
| **实验结果与工作量** | **能够按实验要求完成系统的开发与测试，并达到实验要求的预期结果；能够认真记录实验数据，并对实验结果分析准确，归纳总结充分；工作量饱满。** | **20** |  |
| **报告质量** | **实验报告文字通顺、格式规范，体例符合要求；报告内容充实、正确，实验目的归纳合理到位。** | **40** |  |
| **总 分** | | |  |