

****

本科毕业设计(论文)

GRADUATION DESIGN (THESIS)

题 目\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

学生姓名\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

指导教师\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

学 院\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

专业班级\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

本科生院制

2015年6月

通信专业离散数学动态演示系统的设计与实现

摘要

离散数学(Discrete mathematics)是研究离散量的结构及其相互关系的数学学科，是现代数学的一个重要分支。它在各个学科领域，特别是在计算机科学领域有着广泛的应用。离散数学作为许多计算机相关专业的专业课程，是程序设计语言、 数据结构、操作系统、编译原理、数字电路、人工智能、数据库、算法等必不可少的先修课程。

对于通信工程专业而言，离散数学也有着重要的地位。比如数字电路，数字信号处 理，信息论与编码等都会有所涉及。但是离散数学是一门相对抽象，对逻辑思维要求很高的学科，大部分学生在学习的过程中会感觉比较吃力。

目前，计算机编程技术相对成熟，各种编程语言和开发环境层出不穷。通过广泛查阅各种辅助教学软件的设计思想，构建一个离散数学课程的动态演示系统，帮助学生理解典型的离散数学知识点，直观的掌握离散数学的部分知识。

**关键词：** 通信工程 离散数学 动态演示系统

**Design and implementation of a dynamic demonstration system of discrete mathematics**

**ABSTRACT**

Discrete mathematics is a subject which discusses discrete structures and their relations, and it is an important branch of the modern mathematics. It is widely used in various fields, especially in the field of computer science. Discrete mathematics is a specialized courses of many computer related subjects, which is an Advanced Placement class for programming language, data structure, operating system, compiler theory, digital circuit, artificial intelligence, database, and algorithm, etc.

As for communication engineering, discrete mathematics also plays an important role. Such as digital circuits, digital signal processing, information theory and coding, will be involved. However, discrete mathematics is relatively abstract and needs high logical thinking, the majority of students in the learning process will feel much difficult.

Currently, computer programming technology is mature, a variety of programming languages and development environments are endless. With widely accessible through a series of auxiliary teaching software design, to build a dynamic discrete mathematics presentation system to help students understand the typical discrete mathematics knowledge, and intuitively grasp of part of the knowledge of discrete mathematics.

目录

1. 绪论
   1. 项目研究背景
   2. 国内外研究现状
   3. 任务概述
   4. 论文的结构
2. 系统概要设计
   1. 面向对象的编程思想
      1. 面向对象简介
      2. C++ 简介
      3. Qt 简介
      4. Git 简介
   2. 系统设计
      1. 系统的总体结构
      2. 系统的演示结构
   3. 系统界面设计
   4. 系统基本功能
   5. 系统主要模块
      1. 主窗口
      2. 声明窗口
      3. 帮助窗口
      4. 功能模块
3. 功能模块分析与实现
   1. 验证表达式是否为命题公式
   2. 求任意公式的真值表
   3. 求任意两个集合的运算
   4. 判断关系的性质
4. 系统演示
   1. 系统概述
   2. 声明和帮助
   3. 功能模块演示
5. 总结与展望
   1. 总结
   2. 展望

参考文献

致谢

附录

第1章 绪论

* 1. 项目研究背景

现今计算机技术已经发展的很成熟了，尤其是基于windows 平台的可视化软件更是层出不穷。将图形化编程技术运用到教学中，让学生从枯燥的黑板或者幻灯片中解放出来，通过图形形象的去观察，轻松掌握知识点。

离散数学是所有计算机相关专业学生必修的一门基础课。因为离散数学和计算机息息相关。无论是硬件的存储，编码还是操作系统，编译程序等等都离不开它。通信工程中的数字信号处理，信息论与编码，无线通信等等也都会或多或少用到离散数学的知识。所以其在大学学科中的重要地位不言而喻。

但是另一方面，众所周知，这是一门逻辑思维较强、抽象难懂的学科。定义诸多，推理复杂，学生学起来吃力，老师教起来也有难度。传统的教学中，老师播放幻灯片，学生一扫而过，之后什么印象都没有留下。这种授课方式效率不高。

把图形化编程技术和离散数学结合起来，实现一个动态的离散数学演示系统，让学生也可以自己动手操作，老师教授时也增强交互性。学生兴趣高，而且容易直观学习掌握。

* 1. 国内外研究现状

关于计算机图形化辅助教学工具，还是比较多的。对于计算机系，比较多的有数据结构演示系统，算法演示系统等等。但关于离散数学的演示系统还是很少的。国内有部分教师自己实现基于浏览器的一些演示功能，缺点是不完善，而且依赖浏览器。

在离散数学的一般教学过程中，都是老师通过板书和PPT实现。大部分课件都是繁琐的定义和冗长的推理过程。学生很容易昏昏欲睡。

本系统采用 C++语言开发，直接运行于windows平台，携带方便，而且提供了离散数学的动态演示。

* 1. 任务概述

项目的主要研究工作包括以下几个方面：

1. 根据离散数学课程和毕设任务书，对系统的总体进行设计，得到系统的总体框架。
2. 功能理论分析与实现。分析系统要实现的四个功能，每一部分用到的算法，并通过面向对象的方法，用图形化的方式展示出来。
3. 设计系统的演示界面。包括主界面，帮助和声明界面。以及具体的每一个功能的演示界面。用户可以随时查看随时使用。
   1. 论文的结构

本文的具体组织结构如下：

第1章：绪论。简单的介绍了本项目的一些相关情况。

第2章：系统概要设计。这部分主要是设计相关。介绍了开发环境和开发工具，然后进行需求分析，设计出系统的模块。

第3章：功能模块分析与实现。这部分主要是实现相关。介绍了四个核心功能模块使用到的算法，以及代码实现的关键。

第4章：系统演示。这部分主要是和系统的演示和使用相关。

第5章：总结与展望。对本系统的设计与实现进行总结，并提出改进意见。

第2章 系统概要设计

* 1. 面向对象的编程思想
     1. 面向对象简介

面向对象编程，英文简称OOP（object oriented programing）。与传统的强调算法的过程性编程不同，它强调的是数据，其理念是设计与问题的本质特性相对应的数据格式。OOP程序设计的方法首先是设计类，也就是现实事物的一个抽象，他们准确表示了程序要处理的东西。比如一个学生类，包含了数据成员（学号，姓名，电话，成绩等）和方法（求总成绩，平均成绩，排序等）。

面向对象，具体的来说设计抽象、继承、多态等设计方法。如果引申的话，可以理解为一种设计思想，这种设计思想已经渗透到软件设计的各个方面，如数据库系统，交互式界面，应用结构，分布式系统，人工智能等领域。面向对象的思想又可分为面向对象分析，面向对象设计以及我们经常说的面向对象编程。

采用面向对象的设计方法，优点有很多。易维护，因为设计结构清晰，可读性高，而且由于继承的存在，即使改变需求，维护也只发生在局部模块，所以维护起来是方便且低成本的。质量高，在设计时，可重用现有的，在以前的项目领域中已被测试过的类来使系统满足业务需求并具有较高的质量。效率高，在软件开发时，根据设计的需要对现实世界的事物进行抽象来生成类，使用这样的方法解决问题，接近于日常生活和自然界的思考方式，势必提高软件开发的效率和质量。易扩展，由于继承、封装、多态的特性，自然设计出高内聚、低耦合的系统结构，使得系统更灵活、更容易扩展，而且成本较低。

* + 1. C++ 简介

和C语言一样，C++也是在贝尔实验室诞生的，Bjarne Stroustrup于20世纪80年代在这里开发出了这种语言。C++是C语言的超集，这意味着任何有效的 C 程序都是有效的 C++ 程序。它们之间有些细微差异，但无足轻重。OOP 部分赋予了 C++ 语言将问题所涉及的概念联系起来的能力，C部分则赋予了C++语言紧密联系硬件的能力。C++正是在C的语言基础上添加面向对象的特性，使得该语言更加丰富，并获得了一定程度的成功。

了解过编程语言，如 C、C++、java、php等的同学可能都有体会，C++是最复杂的编程语言之一。事实也的确是这样，因为它要兼C，所以不能像java那样完全面向对象。C++ 有很多知识是让人很容易困惑的，比如引用、多重继承、模板等。但是这并不影响它的杰出特性，代码结构清晰，执行效率高，多种平台支持。C++用于桌面图形化编程是最好不过的了，因为操作系统本身就是使用C和C++编写的。也无需安装其他的运行环境，java 程序就必须安装jre ，程序才可以跑的起来。

* + 1. Qt 简介

Qt是奇趣科技公司于1991年开发的一个跨平台的C++图形用户界面应用程序框架。它拥有应用程序开发者建立美观的图形用户界面所需要的所有功能。基本上，Qt同Windows平台上的 MFC，ATL 是同类型的东西。Qt有开源版本，有完整的一套开发环境，编辑器，开发文档，GUI设计器以及丰富的API。

Qt是基于C++的，采用良好的封装机制使得其模块化程度非常高，可重用性好，对于用户开发来说是非常方便的。而且跨平台，windows下使用Qt写的程序，只需在Linux下重新编译，即可实现移植。Qt开发过程中有两个概念很重要，一个是布局，另外一个是信号和槽。布局就是界面设计，水平和垂直方向的组合和对齐等关系。信号和槽则是窗口程序上响应用户操作的关键。

* + 1. Git 简介

Git是一款开源的、免费的分布式版本控制系统，用于软件开发过程中的源码版本控制，适合各种类型的项目。简单的来说就是管理源代码的各个版本，在不用版本控制工具的时候，我们可能要拷贝整个目录来做个备份，这样一个项目下来可能有好多个目录，而且每一次拷贝备份也没有更多的说明信息。使用Git就可以避免这样的问题，而且Git支持远端操作，本地的版本仓库可以同步到远端。这样即使word崩溃了或是不小心删除了文件，都可以轻松恢复。

Git不仅仅可以管理源代码，理论上所有的文件都可以。比如论文，程序都是可以的。Git的具体使用可以参考《Pro Git》一书。本项目开发过程中，使用Git做版本控制，同时同步到github.com。

* 1. 系统设计