

# 武汉理工大学毕业设计（论文）

## 编译程序在线评测辅助教学系统

### 设计和实现

学院（系）：计算机科学与技术学院

专业班级：软件工程 1503

学生姓名：刘金科

指导教师：饶文碧

# 学位论文原创性声明

本人郑重声明：所呈交的论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包括任何其他个人或集体已经发表或撰写的成果作品。本人完全意识到本声明的法律后果由本人承担。

(宋体小四号)

作者签名：

年 月 日

# 学位论文版权使用授权书

本学位论文作者完全了解学校有关保障、使用学位论文的规定，同意学校保留并向有关学位论文管理部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权省级优秀学士论文评选机构将本学位论文的全部或部分内容编入有关数据进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于 1、保密口，在      年解密后适用本授权书

2、不保密口 。

(请在以上相应方框内打“√”)

作者签名：                      年    月    日

导师签名：                      年    月    日

# 摘 要

本文通过借助 Springboot 框架，以 Idea 为开发平台，Tomcat 作为服务器，Mysql 作为数据库，实现了一个 B/S 模式的基于功能测试的编译程序在线评测辅助教学系统，系统清晰地组织了各大功能模块，提高了应用程序的灵活性和可配置性。

论文主要详细讲述了本系统的整个体系结构，概述了系统的主要功能，分析了每个功能模块的主要任务。然后介绍从相关技术介绍，需求分析，系统设计到具体实现的过程。文章着重介绍了在线评测模块的原理设计，内核评测原理设计，黑盒和白盒测试用例设计，ActiveMQ 生产者消费者模型的设计以及对 6 种评判结果的设计。在此基础上，系统还针对于工程文件的评测，子模块之间的合并评测<sup>[1]</sup>做出了详尽的设计，更加丰富了系统的评测机制。

校内学生和喜欢编程的用户都可以通过本系统来进行题目训练，参与平台发布的比赛。另外，老师可以在平台发布实验作业与教学视频，极大方便了师生之间的教学互动。

**关键词：**在线评测；消息队列；合并编译；用例测试

# Abstract

This article uses the Springboot framework, Idea as the development platform, Tomcat as the server, Mysql as the database, realizes a B/S mode based on the functional test compiler online evaluation auxiliary teaching system, the system clearly organizes the major functional modules. Increased application flexibility and configurability.

The paper mainly describes the whole architecture of the system in detail, summarizes the main functions of the system, and analyzes the main tasks of each functional module. Then introduce the process from related technology introduction, requirements analysis, system design to concrete implementation. The article focuses on the principle design of the online evaluation module, the design of the kernel evaluation principle, the design of the black box and white box test cases, the design of the ActiveMQ producer consumer model and the design of the six evaluation results. On this basis, the system is also designed for the evaluation of engineering documents, and the combined evaluation between sub-modules has made a detailed design<sup>[1]</sup>, which enriches the evaluation mechanism of the system.

Both the students in the school and the users who like to program can use this system to train the questions and participate in the competitions released by the platform. In addition, the teacher can publish experimental homework and teaching videos on the platform, which greatly facilitates the teaching interaction between teachers and students.

**Key Words :** Online evaluation; message queue; merge compilation; use case test

## 目录

摘 要.....	I
Abstract.....	II
第 1 章 绪论.....	1
1.1 研究背景以及研究的目的和意义.....	1
1.2 国内外现状.....	1
1.3 本文主要研究内容.....	1
1.4 本文的结构组织.....	2
第 2 章 系统开发相关技术介绍.....	3
2.1 Springboot 框架.....	3
2.2 Mybatis 自动注入.....	3
2.3 B/S 架构.....	3
2.4 I/O 重定向.....	4
2.5 JAVA 并发之阻塞队列.....	4
2.6 GitHub 代码托管.....	4
第 3 章 系统需求分析.....	5
3.1 系统需求的作用.....	5
3.2 功能需求.....	5
3.2.1 软件系统的接口需求.....	5
3.2.2 软件系统的功能需求.....	5
3.3 非功能需求.....	6
3.3.1 软件系统的性能需求.....	6
3.3.2 软件系统的其他需求.....	7
3.4 本章内容总结.....	7
第 4 章 系统设计.....	8
4.1 系统设计的作用.....	8
4.2 系统总体设计.....	8
4.2.1 平台物理架构设计.....	8
4.2.2 系统软件架构设计.....	9
4.3 在线评测模块设计.....	9

4.3.1 内核评测原理设计 .....	9
4.3.2 黑盒白盒测试用例设计 .....	10
4.3.3 ActiveMQ 消息队列设计 .....	11
4.3.4 评判结果设计 .....	12
4.3.5 合并子模块设计 .....	13
<b>4.4 数据库设计 .....</b>	<b>15</b>
4.4.1 E-R 模型 .....	15
4.4.2 系统数据表设计 .....	18
<b>4.5 本章内容总结 .....</b>	<b>21</b>
<b>第 5 章 系统编码实现和测试 .....</b>	<b>22</b>
5.1 系统开发环境 .....	22
5.2 系统功能模块界面实现 .....	23
5.2.1 系统主页实现及其截图 .....	23
5.2.2 评测模块实现及其截图 .....	23
5.2.3 比赛模块实现及其截图 .....	25
5.2.4 排名和提交状态模块实现及其截图 .....	26
5.2.5 后台管理模块实现及其截图 .....	27
5.3 系统测试 .....	27
5.3.1 测试环境 .....	27
5.3.2 测试方法 .....	28
5.3 本章小结 .....	29
<b>第 6 章 总结与展望 .....</b>	<b>30</b>
6.1 总结 .....	30
6.2 展望 .....	31
<b>参考文献 .....</b>	<b>32</b>
<b>致谢 .....</b>	<b>33</b>

## 第 1 章 绪论

### 1.1 研究背景以及研究的目的和意义

“编译原理”是计算机专业的核心专业课程之一，其主要任务是使学生掌握高级语言的编译技术和高级语言编译程序的设计原理与构造技术，并能够依据编译的基本原理进行一些编译程序的设计和实现。因此，编译程序在线评测系统的开发具有重要的实际意义。一方面，可以将复杂的编译程序简单化、小型化、独立化，使编译程序的实现能够从小到大、从易到难、循序渐进；另一方面，它为学习和使用编译技术提供了一个完整的实践模式，有助于帮助学生树立编写复杂程序的信心，对课程教学起到极大的促进作用。编译程序在线评测系统力求在整体上展现整个编译过程的实现，它将编译过程拆分成一个个模块（阶段），各个模块之间又有着紧密联系，提供一套完整的编程接口和模板供用户进行调用与调试，以实现编译程序的编写和实现。

### 1.2 国内外现状

随着的在线测评系统不断的成熟和完善，越来越多的在线程序评测系统的方式能够充分锻炼学生的分析问题和解决问题的实际能力，更能激发大学生的团结合作精神，创造力和软件开发过程中的创新意识。例如北京大学的openjudge、浙江大学的acm平台、LeetCode在线评测平台。

但是，目前的在线评测系统系统还存在着如下几种缺陷。其一：多数ACM在线评测系统只能用于算法类的测评，通过黑盒测试中的测试用例集合来进行测评；其二：目前的测评系统仅限于支持控制台的输入输出，这对于有着大量输入的程序测试中，控制台的处理变得更加繁杂。其三、平台对用户编写的代码编写仅局限在一个类中，所有的类方法和成员变量仅在一个类中实现。而编译原理算法中每一个子模块（词法分析、语法分析、语义分析等等）都有不同形式的对应答案（多种输入和输出），每一个子模块要在不同类之间进行方法和成员变量的调用和处理，这就使得传统的在线评测系统难以处理。因此、现阶段开发一款编译原理在线评测辅助教学系统就显得很有必要。

### 1.3 本文主要研究内容

（1）编译程序在线评测系统的设计模块拆分：将编译过程各阶段中使用的算法进行了拆分归纳形成彼此独立的模块，由于模块之间的独立性，就可以单独实现其中任一模块的评测而无须实现其所依赖的模块，其依赖的模块将由系统提供的接口予以使用；系统架构：主要由 3 个子系统和其他一些辅助功能组成：合并子系统、评测子系统和展现



展现子系统，以及用户管理、题库管理和即时通信等子系统。

（2）编译程序在线评测系统的实现：重点是合并子系统、展示子系统以及用户管理、题库管理、在线比赛和即时通信等子系统的实现。

（3）解决传统模式下基于黑盒测试在评判系统中的不足，设计并实现满足现代教学目标，如针对面向对象编程思想的考核的细粒度的评判方案。

（4）研究分布式和虚拟化技术、针对大规模竞赛的特点、设计并实现支持大规模程序设计竞赛的在线评判系统。

## 1.4 本文的结构组织

本文的组织结构如下：

第一章为绪论。主要介绍本文的研究背景、研究目的和意义，当前的国内外现状，同时介绍本文的主要研究内容。

第二章为相关技术知识介绍。主要介绍了 Springboot 框架，Mybatis 自动注入，Json+Jquery 的前端知识构件，B/S 的服务架构，I/O 重定向技术，JAVA 并发之消息队列，GitHub 代码托管等相关知识。

第三章为系统需求分析。主要介绍了系统需求的系统的功能需求和非功能需求。系统的功能需求包括接口需求和功能需求；系统的非功能需求包括性能需求和其他需求。

第四章为系统设计。主要介绍了系统的总体设计，在线评测模块设计，数据库的设计。着重阐述了在线评测模块的设计包含内核评测原理设计，黑和白盒测试用例设计，ActiveMQ 消息队列的设计，评判结果的设计，合并子模块的设计。本章内容作为整个系统设计的核心，解决了评判过程中最为核心的问题。

第五章为系统编码和测试。主要介绍了系统的开发环境，各功能模块界面的设计，系统测试。这将为用户重构代码，系统的可扩展性提供一个良好的基础。

第六章为总结与展望。主要介绍了系统目前的状况已经存在的问题，以及到以后的发展前景。

## 第 2 章 系统开发相关技术介绍

### 2.1 Springboot 框架

SpringBoot是在2013年推出的新项目,主要用来简化 Spring 开发框架的开发、配置、调试、部署工作,同时在项目内 集成了大量易于使用且实用的基础框架。可以快速构建一个生产级别的Spring应用程序。<sup>[2]</sup>在Spring Boot中集成的基础框架,是在开发中经常需要使用的框架,如内嵌容器(Tomcat、Jetty、Undertow)、日志框 架、JMS 框架、持久化框架、流行NOSQL 数据库(Cassandra、 MongoDB)、缓存框架等。Spring Boot对目前的的主流构建工具Maven、Gradle都提供了良好的支持,对其他构建工具也提供了支持。

### 2.2 Mybatis 自动注入

MyBatis 是一款优秀的持久层框架,它支持定制化 SQL、存储过程以及高级映射。MyBatis 避免了几乎所有的 JDBC 代码和手动设置参数以及获取结果集<sup>[3]</sup>。MyBatis 可以使用简单的 XML 或注解来配置和映射原生类型、接口和 Java 的 POJO (Plain Old Java Objects, 普通老式Java 对象) 为数据库中的记录。

### 2.3 B/S 架构

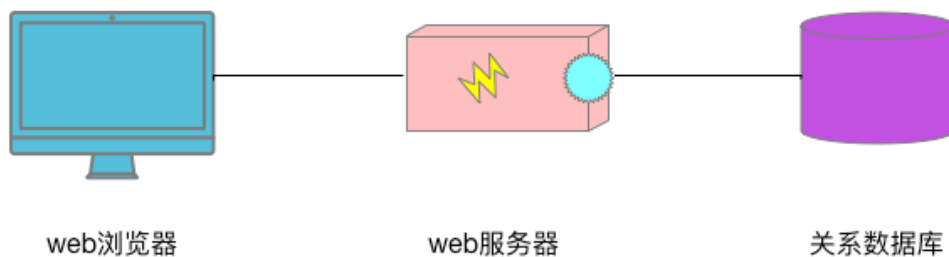


图 2.1 三层 B/S 体系流程图

B/S 结构 (Browser/Server, 浏览器/服务器模式), 是 WEB 兴起后的一种网络结构模式, WEB 浏览器是客户端最主要的应用软件<sup>[4]</sup>。这种模式统一了客户端, 将系统功能实现的核心部分集中到服务器上, 简化了系统的开发、维护和使用。客户机上只要安装一个浏览器, 如 Netscape Navigator 或 Internet Explorer, 服务器安装 SQL Server、Oracle、MYSQL 等数据库。浏览器通过 Web Server 同数据库进行数据交互。

## 2.4 I/O 重定向

实现用户与远程软件工具的交互主要是使用重定向技术实现。操作系统 Shell 环境中支持输入输出重定向,用符号“<”和“>”来表示。0、1 和 2 分别表示标准输入 STDIN、标准输出 STDOUT 和标准错误 STDERR 信息输出,可以用来指定需要重定向的标准输入或输出。可以把输入重定向到一个文件里,重定向操作符可以用来将命令输入和输出数据流从默认位置重定向到其他位置,其输入或输出数据流的位置称为句柄;常见的句柄有三种,即:标准输入、标准输出和标准错误<sup>[5]</sup>。

## 2.5 JAVA 并发之阻塞队列

阻塞队列（BlockingQueue）是一个先进先出的队列（Queue）因为阻塞队列支持当获取队列元素但是队列为空时，会阻塞等待队列中有元素再返回；也支持添加元素时，如果队列已满，那么等到队列可以放入新元素时再放入。阻塞队列是一个接口，继承自 Queue，所以其实现类也可以作为 Queue 的实现来使用，而 Queue 又继承自 Collection 接口。

## 2.6 GitHub 代码托管

Github 是一个代码托管平台和开发者社区，开发者可以在 Github 上创建自己的开源项目并与其他开发者协作编码。创业公司可以用它来托管软件项目，开源项目可以免费托管，私有项目需付费。GitHub 可以托管各种 git 库，并提供一个 web 界面，但与其它像 SourceForge 或 Google Code 这样的服务不同，GitHub 的独特卖点在于从另外一个项目进行分支的简易性。为一个项目贡献代码非常简单：首先点击项目站点的“fork”的按钮，然后将代码检出并将修改加入到刚才分出的代码库中，最后通过内建的“pull request”机制向项目负责人申请代码合并。已经有人将 GitHub 称为代码玩家的 MySpace。

## 第 3 章 系统需求分析

### 3.1 系统需求的作用

软件需求是(1)用户解决问题或达到目标所需条件或权能(Capability)。(2)系统或系统部件要满足合同、标准、规范或其它正式规定文档所需具有的条件或权能。(3)一种反映上面(1)或(2)所述条件或权能的文档说明。它包括功能性需求及非功能性需求,非功能性需求对设计和实现提出了限制,比如性能要求,质量标准,或者设计限制。

### 3.2 功能需求

#### 3.2.1 软件系统的接口需求

(1) 人机接口: 系统分为三种身份登录用户端、老师端、管理员端; 用户端账号注册完成、老师端注册管理员审核完成、管理员由开发人员内部分配。

(2) 硬件接口: 服务器端口号 8081。

(3) 软件接口: 在网络状况良好的情况下, 建议使用谷歌 6.0 版本以上的浏览器, 手机端只可查看信息, 仅仅 PC 端支持在线编程、在线比赛。

(4) 数据接口: 对用户每次提交数据时, 入口函数都有规定要求; 提交文件压缩包时, 压缩包的文件格式有要求; 对于每个子模块(词法分析、语法分析、语义分析、中间代码生成)之间的输入输出都有数据格式的规定。

#### 3.2.2 软件系统的功能需求

系统中各模块要实现的功能:

(1) 题目中心模块: 实现用户查看题目、搜索题目(包括精确查找、模糊查找、多条件查找)、查看题目要求; 用户在线编程编写代码、提交代码(包括文本框提交、文件打包提交)、评判结果输出、错误输出。

(2) 在线教学模块: 实现用户使用校园账号进行登录、课件下载、作业发布查看、作业提交、教学视频观看。

(3) 在线比赛模块: 用户实现比赛报名、比赛登录、比赛信息查看、进入比赛(是否过时判断)、比赛答题、比赛计时(倒计时提醒)、提交题目、退出比赛; 用户可以查看每道题目的通过率、提交记录。

(4) 提交状态模块: 实现用户可以查看近期系统的代码提交状况, 一条状态包括用户 ID、问题名称、结果、内存、耗时、编程语言、题目分类、评判的服务器, 用户提交用户

的主页、源代码；用户实现对提交记录的的查询（包括精确查找、模糊查找、多条件查找）。

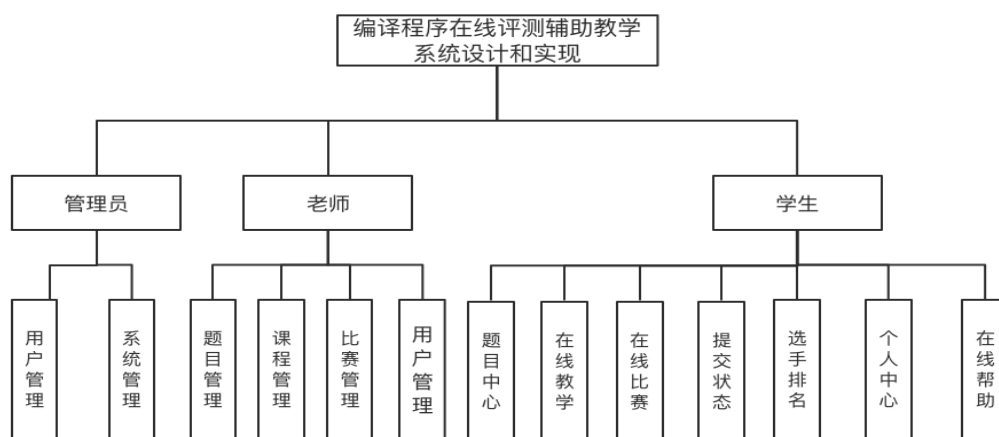


图 3.1 系统功能需求树形图

（5）选手排名模块：实现用户可以查看用户的排名状况，每条记录包括用户的名次、用户 ID、用户名、通过题目数、提交的总次数、通过率、人气指数；用户可以查看用户的主页、给用户增加人气指数、实现模糊查询选手的信息。

（6）帮助中心模块：实现用户可以查看系统的使用手册。手册包括：文本框程序的输入输出接口、文件打包上传的格式要求、编程语言的选择使用、程序提交的时空限制、参加比赛的注册说明。

（7）老师端模块：实现四大核心模块：题库管理、课程管理、比赛管理、用户管理。题库管理包括：题目的查看、查询（模糊查询和多条件查询）、添加试题题目、题目测试用例的导入、编辑试题信息、删除试题；课程管理包括：课程的查看和查询（模糊查询和多条件查询）、课程的录入、作业的发布、实验的发布、资料的查询和上传、讲义的下载；比赛管理包括：比赛的查看、查询、比赛的创建发布、比赛题目的录入、比赛信息的编辑、比赛的删除。用户管理包括：用户的题库授权、用户分析。

（8）管理员中心模块：包括用户管理、用户权限的升级、系统升级发布。

### 3.3 非功能需求

#### 3.3.1 软件系统的性能需求

（1）时间需求：用户登录延迟不能超过 5s；代码提交排队时长不能超过 10s, 超过十秒要有错误处理。

（2）安全需求：对于用户提交的死循环具有破坏系统功能的函数要及时处

理、比赛结束时要对用户的代码进行及时保存，以防用户数据丢失，对于用户多次提交的代码要及时更新。

（3）处理容错需求：对于用户编写的不规范代码要进行容错处理、对于用户的提交和结果输出要体现一致性原则。

### 3.3.2 软件系统的其他需求

系统的可扩展性反映了系统适应软件变化的能力。随着用户需求的不断变化，软件的更新迭代也会随之而来，如果一个系统可扩展性不强的话，就会陷入升级灾难，容易造成系统瘫痪。这就要求开发者在开始之处就要遵循软件开发的设计原则。接口分离原则，里氏替换原则，单一性原则，开闭原则，迪米特原则，接口抽象原则等设计原则以及低耦合，高内聚将是我们必须要着重考虑的地方。

系统界面的需求会对用户评判这个系统起着很重要的影响。在界面设计中，要遵循可用性，易用性，易学性，易急性，低出错率，主观满意度等用户操作事项。我们也好遵循界面的设计原则，简洁设计，一致性设计，低出错率设计，易记性设计等设计原则。

## 3.4 本章内容总结

本章总共主要描述了软件系统的需求分析，从功能需求和非功能需求两个方面进行阐述。功能需求方面主要介绍了软件系统的接口需求和软件系统的功能需求；非功能需求方面主要介绍了软件系统的性能需求和软件系统的其他需求。

## 第4章 系统设计

### 4.1 系统设计的作用

软件设计是从软件需求规格说明书出发，根据需求分析阶段确定的功能设计软件系统的整体结构、划分功能模块、确定每个模块的实现算法以及编写具体的代码，形成软件的具体设计方案。软件设计是把许多事物和问题抽象起来，并且抽象它们不同的层次和角度。整个设计阶段是一个综合性反馈过程。系统设计内容，包括确定系统功能、设计方针和方法，产生理想系统并作出草案，通过收集信息对草案作出修正产生可选设计方案，将系统分解为若干子系统，进行子系统和总系统的详细设计并进行评价，对系统方案进行论证并作出性能效果预测。

### 4.2 系统总体设计

#### 4.2.1 平台物理架构设计

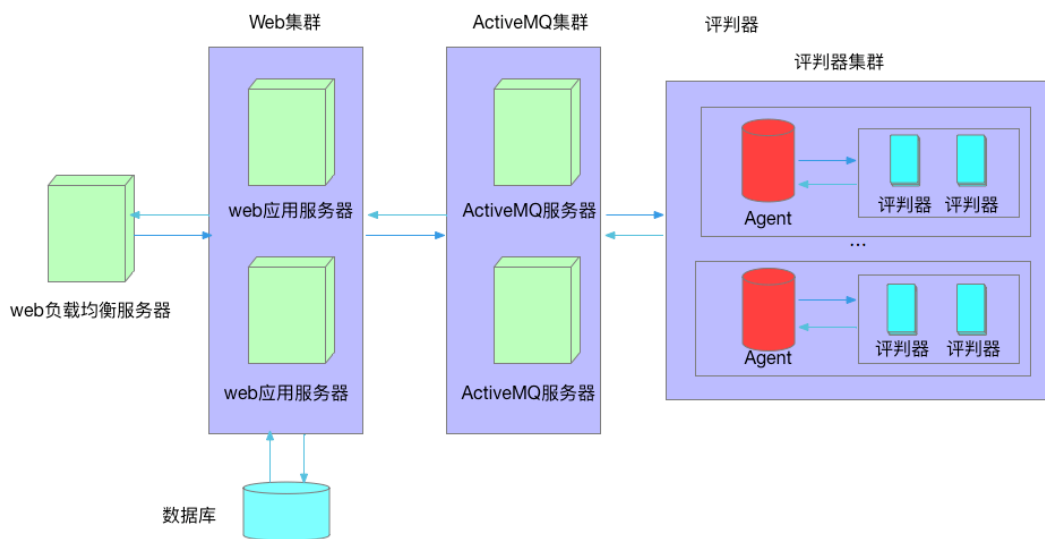


图 4.1 平台物理图

在本文提出的大规模实时编程在线评判系统的设计方案中，需要强调大规模、实时和通用编程题判断评判这三个特性。为了支持规模较大的竞赛，系统整体应具备较大的吞吐能力；为了保证评判的实时性，系统应能自适应各个节点的负载，以达到系统整体最优；为了实现通用编程评判，本文提出了一种机遇解耦和分层设计的插件技术，并针对各类语言程序的编译和运行作了抽象，实现了一种通用的程序评判方案。

## 4.2.2 系统软件架构设计

本系统的架构是建立在基于 Springboot 的一个 Maven 项目。充分利用了面向对象的设计原则，通过该架构可以提高系统的可维护性，可伸缩性和系统性能。该开发平台是基于三层架构的 B/S 架构，由分散在多个层的松散耦合的中间件组成。软件架构图如图 4-1 所示。

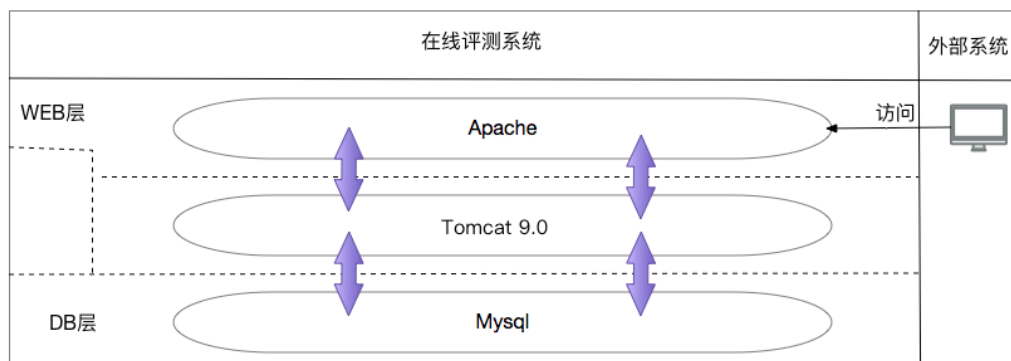


图 4.2 软件架构图

## 4.3 在线评测模块设计

### 4.3.1 内核评测原理设计

用户通过页面表单将题号、语言、解题代码提交后，由业务层做一些持久化工作，并新建一个唯一编号的评判任务，编号是为了让评判内核能够唯一确定某用户在某题上的一次提交。将每一个评判任务都设计为一个独立的线程，[2]根据用户选择的语言生成对应类型的线程。如 C 线程、GPP 线程，Java 线程等，视 OJ 系统支持的编译系统而定，然后立即将其加入评判队列。评判队列已为了监听评判队列的变化，设计一个评判内核启动器 Starter，自动地从评判队列中获取评判任务并启动评判过程<sup>[6]</sup>。按照编译系统的不同编译生成的文件分为可执行文件或字节码文件，前者运行在操作系统之上，后者运行在虚拟机之上。评判内核自身无法提供这 2 种不同的运行平台。由于评判任务必须在题目规定的时间内完成评判，当时限到达时即使没有获得运行结果也必须结束子进程。我们通过调用系统所提供的编译器的接口，让用户提交的源代码进行去编译，通过重定向技术将最终的结果返回给用户。因此设计了一个计时器 TimerTask 来控制子进程的执行时间。先将标准输入数据通过子进程的输入流传递给编译好的程序。然后将 TimerTask 启动，若在题目规定的时限内仍未从子进程的输出流获得任何结果，则 TimerTask 强制销毁子进程，评判程序



运行超时；若在题目规定时限内从子进程输出流获得了运行结果则立即停止 TimerTask 的计时，记算时间消耗和内存消耗，判断子进程的出口值，确定解题程序在执行过程中是否运行错误。若出口值不等于 0 则表示子进程运行异常，再通过分析子进程的错误流来确定错误的种类。例如数组越界、除数为 0、数据溢出、内存溢出、栈溢出等，否则程序运行正常。

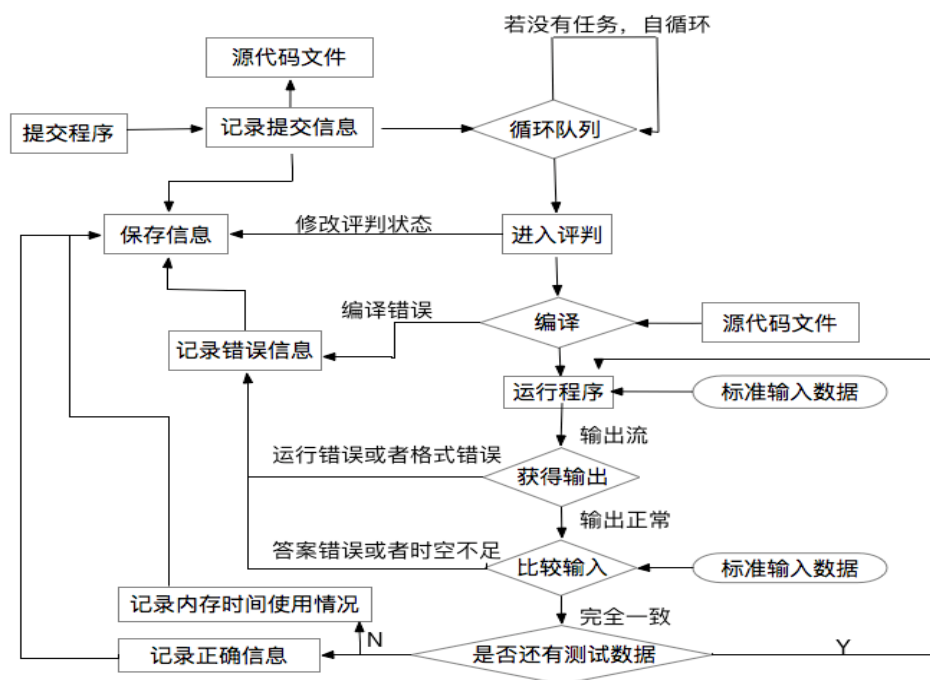


图 4.3 评判流程图

此时应将子进程的输出结果与标准测试数据对比，若去掉前后空白和换行时完全匹配则返回精度错误，若去掉前后空白和换行时也不匹配则返回答案错误，若完全匹配则返回答案正确。程序经过编译、运行、返回结果之后，业务层将结果、时间消耗、空间消耗等发送到持久层进行保存，并将结果和相关数据(如耗时、内存使用、代码长度、使用语言等)发送给表现层呈现给用户。完成本次评判任务<sup>[7]</sup>。

#### 4.3.2 黑盒白盒测试用例设计

在线评测系统在线评测，采用的白盒、黑盒相结合的用例测试方法<sup>[8]</sup>。对于用户提交的编写代码和编写文件第一步先进行白盒测试。白盒测试称为结构测试或逻辑驱动测试，是针对被测单元内部是如何进行工作的测试。它根据程序的控制结构设计测试用例，主要用于软件或程序验证。白盒测试法检查程序内部逻辑结构，对所有的逻辑路径进行测试，是一种穷举路径的测试方法。如果用户提交的是文本框代码，首先通过文件指针读入到到

main.java 或者 (main.cpp) 文件中，通过调用 JavaCompiler 中函数去执行编译过程（Runtime.getRuntime().exec()）。通过 I/O 重定向技术将标准输出流输出到 error.txt 文件中，完成编译的测试工作。第二步进行黑盒测试，黑盒测试又称为功能测试、数据驱动测试或基于规格说明书的测试，是一种从用户观点出发的测试。因为在线评测系统的题库分为算法 1000 道题库和编译原理子模块的题库，不同的题库对应着不同类型的测试用例，因为我们针将输入输出判断量很大的题型的测试用例通过自动生成测试用例来完成导入；对于输入输出答案比较短的采用手动输入多组测试用例，提交给系统完成评判功能。

表 4.1 评判测试方法表

	白盒测试	黑盒测试
评判粒度	程序内部	程序结果
评判标准	内部逻辑结构	输出结果
测试来源	调用编译器接口去判断	通过输入输出测试集去比对
评测方法	模块接口测试	边界值测试
对出题者的要求	高	低
测试次序	第一步	第二步

#### 4.3.3 ActiveMQ 消息队列设计

消息队列是为了解决系统组件通讯问题而诞生的。需要使用消息中间件通讯的组件通常有一定的规模并大部分位于不同的逻辑机器上，简单的进程通讯手段无法满足要求，如共享内存、共享文件和管道等。若将各个组件用网络通信接口 Socket 进行两两相连，则形成网状接口。这样的结构对网络开销的成本都很大。首先，每个组件都需要知道其他组件的信息，如数据交换的网络地址，需要预先配置或者额外的配置管理服务，不利于系统规模在线扩展。其次，在网状结构中，各组件的地位是平等的，任务的调度和容灾都需要很精细地控制。最后，直接点对点的服务调用难以充分根据目标机器当前吞吐量进行调度。使用了消息队列<sup>[9]</sup>之后，网状的复杂结构优化成总线型结构，所有组件直接连接到总线上，各自从总线竞争消费资源。系统组件根据自身负载情况从总线上拉去消息，实现按吞吐量调度任务。消息队列的队列模式是一个生产者消费者模型，天然支持组件的在线调整。此

外，队列模式是一种异步编程模式，可以更加充分地利用处理器资源，减少不必要的等待时间。

本文选用 ActiveMQ 作为消息中间件。ActiveMQ 是 Apache 基金会的顶级项目，在支持 JMS (Java Message Service, Java 消息服务) 标准的同时，兼容工业界主流消费交换协议，如 OpenWire, Stomp, AMQP 和 MQTT 等，方便与其他非 Java 语言编写程序进行交互。ActiveMQ 支持多热设备，能够根据消费者对消息处理的速度动态调整消费者权重，从而优先将消息投递给响应时间的消费者，从而实现负载均衡，并具有良好的生产性能<sup>[10]</sup>。

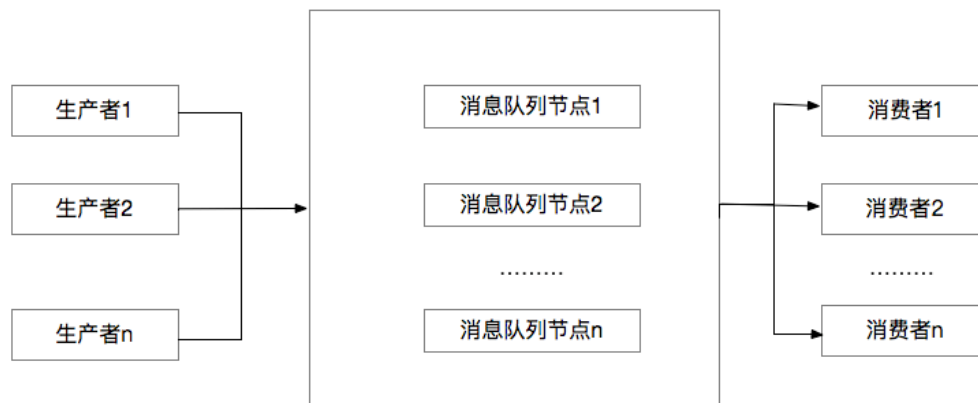


图 4.4 生产者-消费者集群模型图

#### 4.3.4 评判结果设计

测试模块采用先白盒测试后黑盒测试的方法。首先评判结果是根据提交的题目来进行评测的。通常的代码测试提交通常是以文本框的形式提交，而这样的提交方式常常有局限性，这仅仅对一些小的算法通常在只包含一个类文件的算法进行的提交。但是我们平时对一些大中型的算法项目比如向词法分析、语法分析、语义分析等常常涉及几个类文件的处理，显然这样的单文本框的处理显然不能满足大中型算法的处理。因此，针对以上情况，我们设计了文本框提交和文件打包提交评判的两种方式，这样就极大地满足了用户的代码评判需求。

接下来当用户提交代码之后就是评判的过程了。第一步，白盒测试评判，针对用户提交的代码进行评测。将用户的处理方式分为两种：

(1) 如果用户提交的是文本框代码，将会根据所选择的编码类型保存为 main.java 文件（或者 main.cpp 文件）

(2) 如果用户提交的.zip 格式的打包文件, 先将代码解压到指定工程目录下, 等待系统调用编译器进行编译。

首先对于不同的编码文件调用不同的编译器, 比如 Java 文件则调用 `ToolProvider.getSystemJavaCompiler()` 函数生成.class 文件, Cpp 文件则 Java 通过 `Exec()` 函数执行编译命令 “`g++ -w -g -o main your_cpp_name`”

生成可执行文件。如果编译过程中出现编译错误, 则通过 I/O 输出重定向将错误输出到 `error.txt` 中, 更新数据库的提交记录。若第一步的编译成功编译, 第二步则通过测试用例进行黑盒测试, 第一步白盒测试中生成了.class 类文件和 unix 可执行文件, 因此需调用不同的函数去执行程序。在判定的过程中会用 `System.currentTimeMillis()` 记录下执行的时间, 用 `Run.totalMemory()` 记录下执行所消耗的内存<sup>[11]</sup>。评判结果分为以下五种:

(1) Accepted: 表示程序完成了所有的测试用例和编译过程, 成功地输出了结果。

(2) AnswerError: 表示程序完成了编译过程, 但是输入输出答案错误或者没有通过所有的测试用例。

(3) CompilerError: 表示程序在编译的过程中出现错误。

(4) TimeOut: 表示程序编译成功, 但是在执行过程中出现了运行超时。

(5) MemoryOut: 表示编译成功, 但是在执行过程中出现了内存溢出。

(6) FunctionError: 表示在编译前, 用户在编写函数时, 调用了一些不安全的函数。

#### 4.3.5 合并子模块设计

编译程序在线评测辅助教学系统作为在编译原理课程中的一门重要的实验应用。除过在线比赛之外, 将在线评测系统应用于编译原理的算法合并。编译原理作为一门比较抽象的学科, 为了让学生们能够更好地理解整个编译过程, 我们将编译原理从编译过程中的各个阶段, 词法分析、语法分析、语义分析、中间代码生成、代码优化各个过程进一步的细化。因为编译原理的中间各个算法不像数据结构中算法设计, 有着标准的输入和输出, 而编译原理某种算法的过程可能具有多种的分析方法。

譬如: 语法分析有自底向上的语法分析, 有自顶向下的语法分析。NFA 到 DFA 之间的转化, 因为 DFA 可能能最小化成最小的 DFA, 这就使得可能结果存在着多种情况。这时就不能单单地用一种算法的标准输入输出来评判整个程序的结果, 因此我们想了一种办法。由于中间路径的多样性, 中间结果的不唯一性, 于是采用合并编译, 因为最终得到的答案是唯一的, 只需要将整个过程合并起来, 一起编译, 进而来判定中间程序的正确性。

合并子模块的设计原理如上图 4.3 所示, 我们将整个编译原理的中间过程划分成各个子模块, 例如词法分析可以分为文法输入转化 NFA, NFA 转化 DFA, DFA 转化最小 DFA 这三个过程。我们作出如下设定:

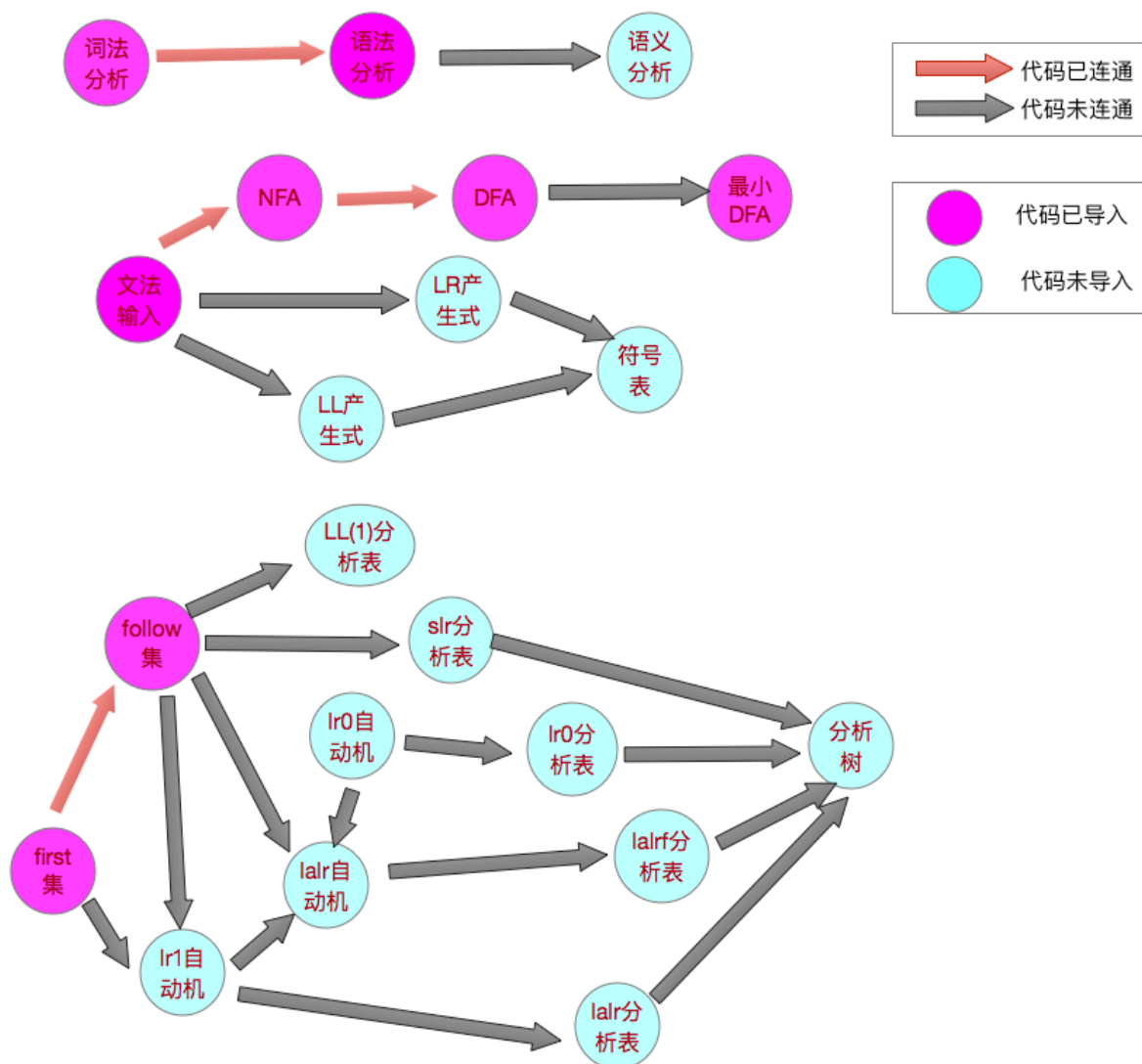


图 4.5 合并子模块设计图

- (1) 红色的圆圈表示：该算法的代码已经导入；
- (2) 蓝色的圆圈表示：该算法的代码还未导入；
- (3) 红色的箭头表示：两端的路径已经完成连通；
- (4) 灰色的箭头表示：两端的路径还未连通；
- (5) 两点之间没有箭头表示：两端之间不能连通。

合并的过程是：第一步首先学生到处该点的算法，然后提交给系统编译，如果能编译成功，则将错误返回给用户。第二步合并子模块，选择要合并的路线，一定要确保选择的路径是连通的，然后点击提交按钮，让系统去评判，如果评判成功，则告知用户答案正确，则将之间的路径变成红色。反之，答案错误，用户重新编译代码。

## 4.4 数据库设计

### 4.4.1 E-R 模型

E-R 模型（Entity-Relationship）就是实体-关系模型，用来描述数据库中各种数据之间的关系的图形表示工具，构成 E-R 图的基本主要实体，属性和关系。E-R 的实体(entity)即数据模型中的数据对象，例如人、学生、音乐都可以作为一个数据对象，用长方体来表示，每个实体都有自己的实体成员(entity member)或者说实体对象(entity instance)，例如学生实体里包括张三、李四等，实体成员/实体实例（不需要出现在 ER 图中。ER 图的属性(attribute)即数据对象所具有的属性，例如学生具有姓名、学号、年级等属性，用椭圆形表示，属性分为唯一属性(unique attribute)和非唯一属性，唯一属性指的是唯一可用来标识该实体实例或者成员的属性，用下划线表示，一般来讲实体都至少有一个唯一属性。ER 图的关系(relationship)用来表现数据对象与数据对象之间的联系，例如学生的实体和成绩表的实体之间有一定的联系，每个学生都有自己的成绩表，这就是一种关系，关系用菱形来表示。

(1) 用户实体。用户的实体属性包括用户的 ID、用户的名称、密码、学校、邮箱、性别、等级、身份、注册时间等。

(2) 题目实体。题目的实体属性包括题目的 ID、题目的名称、分类、作者、题目内容、样本输入、样本输出、题目的等级、状态、是否完成用例导入、创建时间。

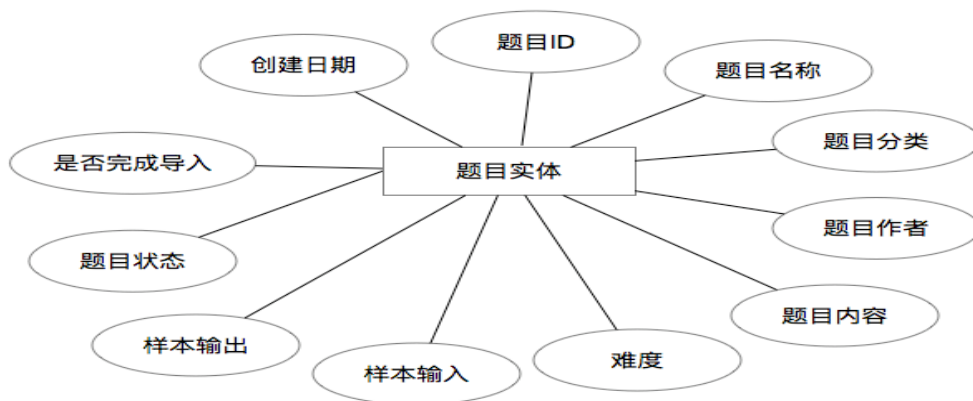


图 4.7 题目实体属性图

(3) 比赛信息实体。比赛信息的实体包括比赛的 ID，比赛的题目、主办方、开始时间、结束时间、比赛形式、比赛内容、比赛语言、比赛状态、是否导入题目、常见日期。

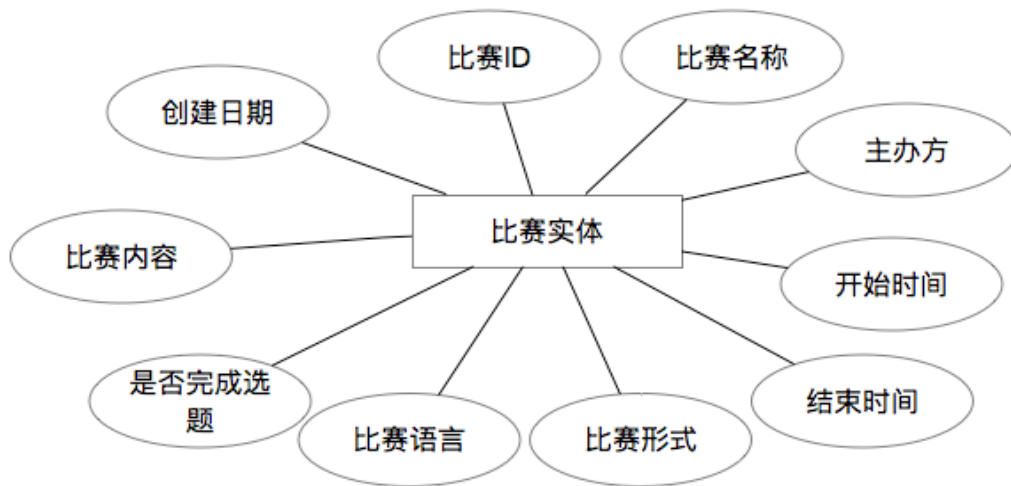


图 4.8 比赛状态实体属性图

（4）提交记录实体。提交记录的实体包括提交记录的 ID、题目名称、题目的分类、提交的用户 ID、提交的内容、提交的次数、是否提交成功、错误的原因、运行时间、运行内存、评测的语言、提交的日期。

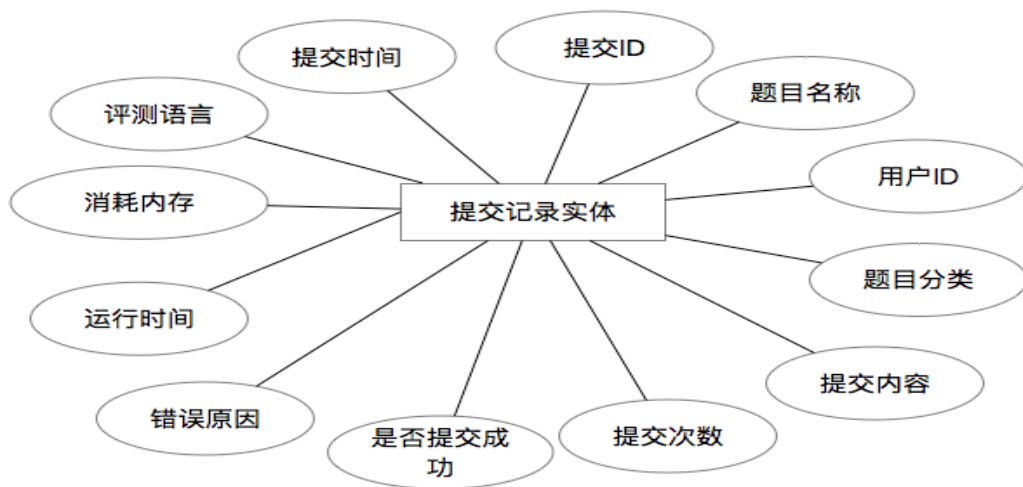


图 4.9 提交记录实体属性图

（5）参赛用户实体。参赛用户的实体包括参赛人员编号的 ID、比赛的 ID、用户的 ID、用户的排名、提交次数、通过题目数量、注册成功日期。

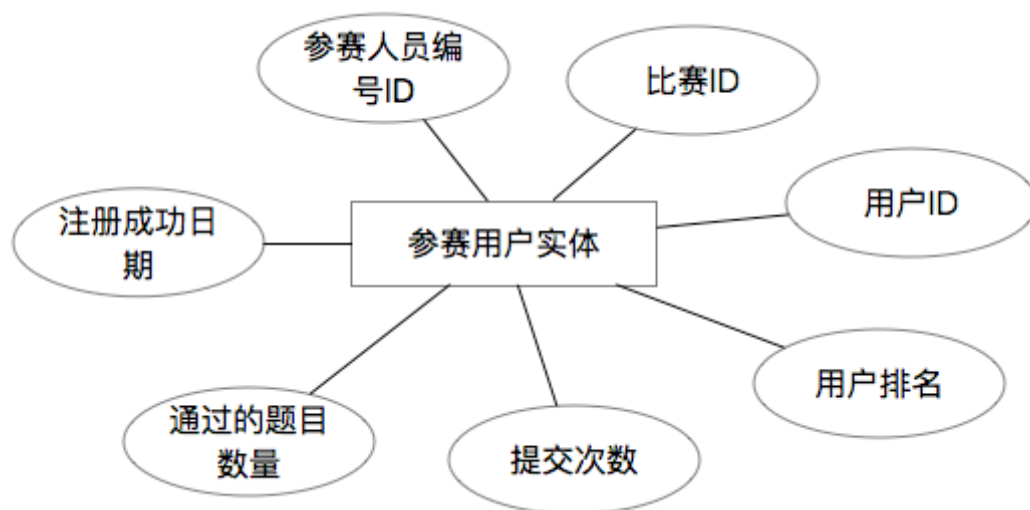


图 4.10 参赛用户实体属性图

（6）系统全局的 E-R 设计图。全局 ER 模式的确定,除能准确、全面地反映用户功能需求外,还应满足:实体类型的个数尽可能少;实体类型所含属性个数尽可能少;实体类型之间无冗余关系。全局 ER 在优化时一般遵循 3 个原则。

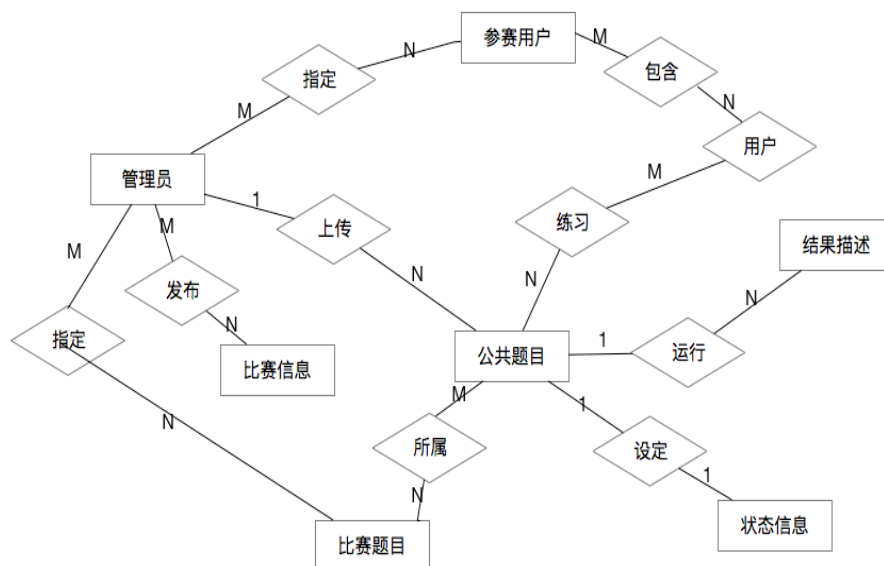


图 4.11 系统全局 E-R 图

①实体类型的合并。即相关实体类型的合并。在全局模型中,实体类型最终转换成关系模式涉及多个实体类型的信息要通过链接操作获得。因而减少实体类型个数,可减少连接的开销,提高处理效率。



②冗余属性的消除。在各个局部结构中一般不允许存在冗余属性。但在综合成全局 ER 模式后,可能产生全局范围内的冗余属性。通常同一非键的属性出现在几个实体类型中,或一个属性值可以从其他属性导出,此时,应将这些冗余的属性从全局模式中去除。

③冗余联系的消除。局部结构综合成全局模式时,有时会出现冗余的联系,此时可以借助规范化理论中函数依赖的概念消除冗余联系。

但在实际的设计过程中,这些条件不是绝对的,要视具体情况而定。在本系统中设计了 6 个实体,在全局 ER 设计时,消除了一些冗余的联系:但兼顾了实际需求和访问效率,根据本系统硬件可以接受的损耗。最终全局的 ER 模式如图 4.14 所示。

#### 4.4.2 系统数据表设计

(1) 用户信息表。表 4.2 用来存储用户的个人信息,分为静态数据字段和动态数据字段。其中 UserID 是主键。

表 4.2 用户信息数据表

变量名称	说明	数据类型
UserId	用户 ID	Int (5)
username	用户名	Varchar (20)
password	密码	Varchar (20)
userEmail	邮箱	Varchar (20)
userSchool	学校	Varchar (20)
userIdentity	身份	Varchar (20)
userLevel	等级	Int (5)
userGender	性别	Tinyint (1)
registerTime	注册日期	Varchar (20)

(2) 题目信息表。表 4.3 用来存储题目的信息表。分为静态数据字段和动态数据字段。其中 ProblemID 是主键。

表 4.3 题目信息数据表

变量名称	说明	数据类型
problemId	题目 ID	Int (5)
problemName	题目名称	Varchar (20)
problemAuthor	题目作者	Varchar (20)
problemState	题目状态	Tinyint (1)
problemInput	样本输入	Varchar (1000)
problemOutput	样本输出	Varchar (1000)
ProblemContent	题目内容	Varchar (5000)
problemLevel	难度等级	Int (2)
problemIstest	是否导入测试用例	Tinyint (1)
uploadTime	上传时间	Varchar (20)

（3）提交状态信息表。表 4.4 用来存储题目提交状态的信息表。分为静态数据字段和动态数据字段。其中 submitId 是主键。

表 4.4 提交状态信息数据表

变量名称	说明	数据类型
submitId	提交 ID	Int (5)
submitName	题目名称	Varchar (20)
submitnameClassfy	题目分类	Varchar (20)
submitUserId	用户 ID	Int (5)
submitContent	提交内容	Varchar (5000)
timeOut	运行时间	Varchar (20)
memoryOut	运行内存	Varchar (20)

submitNums	提交次数	Int (2)
submitSuccess	是否提交成功	Tinyint (1)
errorCause	错误原因	Varchar (5000)
codingLanguage	编码语言	Varchar (10)
uploadTime	提交时间	Varchar (20)

（4）比赛信息表。表 4.5 用来存储比赛的信息表。分为静态数据字段和动态数据字段。其中 contestId 是主键。

表 4.5 比赛信息数据表

变量名称	说明	数据类型
contestId	比赛 ID	Int (5)
contestName	比赛名称	Varchar (20)
contestHost	比赛主办方	Varchar (20)
contestStartTime	开始时间	Varchar (20)
contestEndTime	结束时间	Varchar (20)
contestForm	比赛形式	Varchar (20)
contestContent	比赛内容	Varchar (500)
contestLanguage	比赛评判语言	Varchar (20)
contestState	比赛状态	Tinyint (1)
contestCreateTime	创建时间	Varchar (20)

（5）参赛用户信息表。表 4.6 用来参赛用户的信息表。分为静态数据字段和动态数据字段。其中 matchId 是主键。

表 4.6 参赛用户数据表

变量名称	说明	数据类型
mathchId	参赛 Id	Int (5)
contestId	比赛 Id	Int (5)
userId	用户 Id	Int (5)
userRank	用户排名	Int (5)
submitNums	提交次数	Int (5)
AcceptNums	通过题目数量	Int (5)
registertime	注册日期	Varchar (20)

## 4.5 本章内容总结

本章主要从系统设计的作用、在线评测模块设计、系统的总体设计、在线评测模块设计、数据库设计四个方面展开阐述。从系统总体设计、系统模块设计、系统数据库设计等安全设计等方面对编译原理在线评测系统的设计思想进行了描述。通过系统的架构架构图、软件架构图、内核评测原理设计、黑盒白盒测试用例对比图、消息队列的消息处理以及数据库的 E-R 图模型，数据库表的设计，建立了系统的组织模型，业务模型及信息模型，为进一步的系统实现提供了有力的支撑。

## 第 5 章 系统编码实现和测试

### 5.1 系统开发环境

本项目的整个开发工作是在 MacOS 的开发平台上完成的，开发工具使用的是 Idea；数据模型采用的关系型数据库模式，后台数据库采用的是 MySQL 数据库，应用服务器采用的是 Tomcat9.0 版本，后台采用的框架是 SpringBoot 框架。开发的环境是先通过在本本地搭建模拟的服务器环境，等到整个项目 1.0 版本完成以后，然后再部署到服务器，进行压力测试。服务器是租用腾讯云的学生版服务器，因为服务器的容量有限，因此在压力测试时，会出现多用户访问时，出现卡顿的现象。

表 5.1 系统软件及运行环境概述表

名称	描述
客户端操作系统	Windows All/Mac OS
应用服务器操作系统	Windows All/Mac OS
数据服务器端操作系统	Windows All/Mac OS
开发环境操作系统	Windows All/Mac OS
应用服务器	Apache Tomcat 1.9
数据库	Mysql 8.0.15
开发工具	Idea 1.8.0
调试浏览器	谷歌浏览器 74.0.3729.131

IDEA 全称 IntelliJ IDEA，是 java 编程语言开发的集成环境。IntelliJ 在业界被公认为最好的 java 开发工具之一，尤其在智能代码助手、代码自动提示、重构、J2EE 支持、各类版本工具(git、svn 等)、JUnit、CVS 整合、代码分析、创新的 GUI 设计等方面的功能可以说是超常的。IDEA 是 JetBrains 公司的产品，这家公司总部位于捷克共和国的首都布拉格，开发人员以严谨著称的东欧程序员为主。它的旗舰版本还支持 HTML，CSS，PHP，MySQL，Python 等。用 Idea 来进行前后端分离的开发工作，会使整个系统层次变得更加清

楚。同时，Idea 还内置了数据库的借口连接，这样就会是的不在下载第三的数据库管理工具。

## 5.2 系统功能模块界面实现

### 5.2.1 系统主页实现及其截图

该页面主要包括题目中心、在线教学、在线比赛、提交状态、选手排名、帮助中心几大模块，还用用户的登录注册功能。初次使用系统的用户可以在次页完成登录注册，在帮助中心可以查看系统的使用说明。如下图 5.1-5.4 所示



图 5.1 系统主页部分截图

### 5.2.2 评测模块实现及其截图

评测模块主要包括题目的查看搜索、源代码的在线编写、文件打包的提交，评测的结果和状态。



图 5.2 题目查看搜索截图

评判核心代码：

```
for(Testinfo testinfo:testinfos){
```

```

/** 重定向输出到 111 文件中 */
try {
    PrintStream input = new PrintStream(new
FileOutputStream("/Users/tp5admin/Desktop/CodingOnline/test/src/main/java/com/example/login/code/t
estset/inputset/input.txt"));
    System.setOut(input);
} catch (IOException e) {
    e.printStackTrace();
}
System.out.println(testinfo.getTestinput()); /** 得到输入*/
/** 从文件中读取输入 */
FileInputStream fis = null;
try {
    fis = new
FileInputStream("/Users/tp5admin/Desktop/CodingOnline/test/src/main/java/com/example/login/code/tes
tset/inputset/input.txt");
} catch (FileNotFoundException e) {
    e.printStackTrace();
}
System.setIn(fis);
Scanner scanner = new Scanner(System.in);
process = run.exec(cmd);//当前平台执行对应命令
/** 从控制台输入 */
OutputStream out = process.getOutputStream();
PrintWriter writer = new PrintWriter(out);
while (scanner.hasNextLine()) {
    writer.println(scanner.nextLine());
    writer.flush();
}

/** 输出执行结果 重定向输出 */
PrintStream result = new PrintStream(new
FileOutputStream("/Users/tp5admin/Desktop/CodingOnline/test/src/main/java/com/example/login/code/t
estset/buffoutput/result.txt",true));

```

```
System.setOut(result);

BufferedReader br = new BufferedReader(new InputStreamReader(process.getInputStream()));//获取执行进程的输入流

String runInfo = null;

while (null != (runInfo = br.readLine())) { //读取执行结果并写入到 reslut.txt 中

    System.out.println(runInfo);

}

}
```

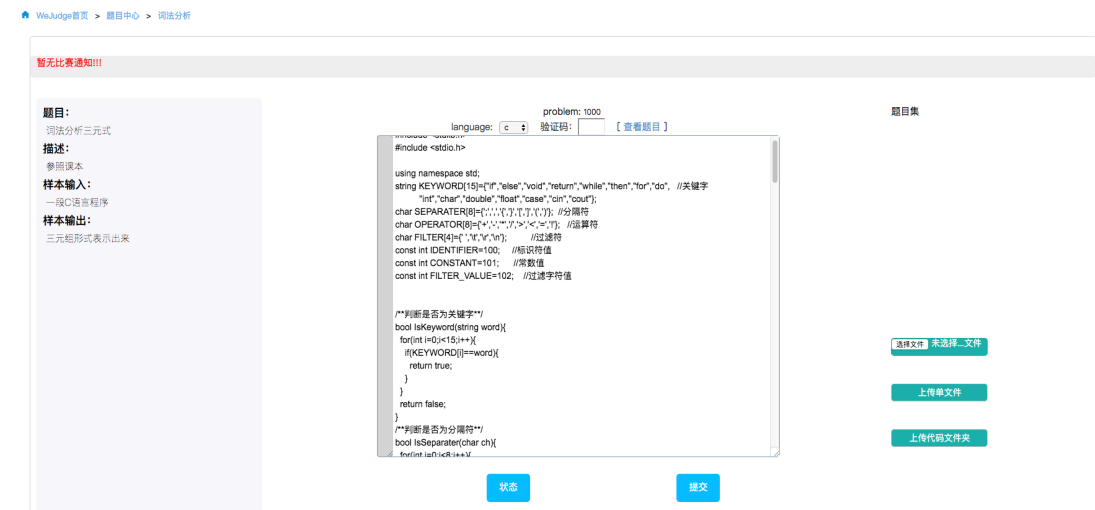


图 5.3 代码编写截图

暂无比赛通知!!!									
题目编号: <input type="text"/> 用户: <input type="text"/> 语言: <input type="text"/> 结果: <input type="text"/> <input type="button" value="提交"/>									
提交编号	用户	问题	结果	内存	耗时	语言	分类	操作	提交时间
1	james	A+B Problem	Answer Error	1543ms	2150kb	java/编辑	算法1000道	合并子模块	2019-05-20 11:00:54
2	james	A+B Problem	Compile Error	nullms	nullkb	c/编辑	算法1000道	合并子模块	2019-05-20 11:00:38
3	james	词法分析三元式	ACCEPT	445ms	4123kb	cpp/编辑	词法分析	合并子模块	2019-05-17 14:52:30
4	james	词法分析三元式	Answer Error	508ms	3248kb	c/编辑	词法分析	合并子模块	2019-05-17 14:50:36
5	james	词法分析三元式	Answer Error	3273ms	4895kb	cpp/编辑	词法分析	合并子模块	2019-05-17 14:47:55
6	james	词法分析三元式	Answer Error	512ms	1119kb	cpp/编辑	词法分析	合并子模块	2019-05-17 14:44:32
7	james	Exponentiation	ACCEPT	915ms	6735kb	c/编辑	算法1000道	合并子模块	2019-05-17 14:15:26
8	james	Exponentiation	Answer Error	332ms	1187kb	c/编辑	算法1000道	合并子模块	2019-05-17 14:15:16
9	james	Exponentiation	TimeOut	超时ms	nullkb	cpp/编辑	算法1000道	合并子模块	2019-05-17 14:13:05
10	james	A+B Problem	ACCEPT	472ms	3645kb	cpp/编辑	算法1000道	合并子模块	2019-05-17 14:11:41

图 5.4 提交状态截图

5.2.3 比赛模块实现及其截图

比赛模块主要包括了比赛信息浏览、进入比赛、作答提交几个部分。用户可以在里



面进行比赛注册、参与比赛等。

比赛通知!!!						
<div>刷新列表</div> <div>筛选</div>						
编号	名称	主办方	比赛开始时间	比赛时长	参赛形式	状态
1	武汉理工大学新生杯	武汉理工大学	2019-05-24 10:00	2小时	个人	在线
2	武汉理工大学蓝桥杯	武汉ACM协会	2019-06-10 10:00	2小时	团队	在线

图 5.5 比赛信息截图

题目:

487-3279

描述:

Businesses like to have memorable telephone numbers. One way to make a telephone number memorable is to have it spell a memorable word or phrase. For example, you can call the University of Waterloo by dialing the memorable TUT-GLOP. Sometimes only part of the number is used to spell a word. When you get back to your hotel tonight you can order a pizza from Gino's by dialing 310-GINO. Another way to make a telephone number memorable is to group the digits in a memorable way. You could order your pizza from Pizza Hut by calling their "three tens" number 3-10-10-10. The standard form of a telephone number is seven decimal digits with a hyphen between the third and fourth digits (e.g. 888-1200). The keypad of a phone supplies the mapping of letters to numbers, as follows: A, B, and C map to 2; D, E, and F map to 3; G, H, and I map to 4; J, K, and L map to 5; M, N, and O map to 6; P, R, and S map to 7; T, U, and V map to 8; W, X, and Y map to 9.

样本输入:

12 4873279 ITS-EASY 888-4567 3-10-10-10 888-GLCP TUT-GLOP 967-11-11 310-GINO F101010 888-1200 -4-8-7-3-2-7-9- 487-3279

样本输出:

310-1010 2 487-3279 4 888-4567 3

提交

题目区

57

58

59

60

图 5.6 比赛作答区截图

5.2.4 排名和提交状态模块实现及其截图

排名和提交状态模用户可以进行提交的记录查看以及选手排名的浏览。

WU Judge

题目中心

在线教学

在线比赛

提交状态

选手排名

帮助中心

James

WU Judge 首页 > 提交状态

暂无比赛通知!!!

题目名称:

题目类别:

难度:

出题者:

提交编号	用户ID	问题	结果	内存	耗时	语言	分类	提交时间	评判服务器
619	14	词法分析三元式	ACCEPT	568ms	3711kb	cpp	词法分析	2019-05-20 11:08:41	127.0.0.1
618	14	A+B Problem	Answer Error	1543ms	2150kb	java	算法1000道	2019-05-20 11:00:54	127.0.0.1
617	14	A+B Problem	Compile Error	nullms	nullkb	c	算法1000道	2019-05-20 11:00:38	127.0.0.1
616	15	A+B Problem	ACCEPT	437ms	1520kb	c	算法1000道	2019-05-18 16:20:24	127.0.0.1
615	14	词法分析三元式	ACCEPT	445ms	4123kb	cpp	词法分析	2019-05-17 14:52:30	127.0.0.1
614	14	词法分析三元式	Answer Error	508ms	3248kb	c	词法分析	2019-05-17 14:50:36	127.0.0.1
613	14	词法分析三元式	Answer Error	3273ms	4885kb	cpp	词法分析	2019-05-17 14:47:55	127.0.0.1
612	14	词法分析三元式	Answer Error	512ms	1119kb	cpp	词法分析	2019-05-17 14:44:32	127.0.0.1

图 5.7 提交记录截图



名次	用户ID	用户名	通过	提交	通过率
1	14	james	6	55	10.91%
2	15	ljk	1	1	100.00%

图 5.8 选手排名截图

## 5.2.5 后台管理模块实现及其截图

后台管理模块主要分为题库管理、课程管理、比赛管理、用户管理四个模块。重要提供给老师和管理员使用。如下图 5.9 所示



ID	题目ID	题目	出题者	题目状态	是否载入用例	创建日期	操作
1	57	A+B Problem	北京大学OJ题库	在线	已载入	2019-05-17 12:13:31	<a href="#">编辑</a> <a href="#">删除</a>
2	58	Exponentiation	北京大学OJ题库	在线	已载入	2019-05-17 12:24:19	<a href="#">编辑</a> <a href="#">删除</a>
3	59	487-3279	北京大学OJ题库	在线	已载入	2019-05-17 13:42:43	<a href="#">编辑</a> <a href="#">删除</a>
4	60	词法分析三元式	饶文碧老师	在线	已载入	2019-05-17 14:43:47	<a href="#">编辑</a> <a href="#">删除</a>
5	61	io	ujl	在线	已载入	2019-05-18 16:27:10	<a href="#">编辑</a> <a href="#">删除</a>

图 5.9 后台管理截图

## 5.3 系统测试

系统测试，英文是 System Testing。是对整个系统的测试，将硬件、软件、操作人员看作一个整体，检验它是否有不符合系统说明书的地方。这种测试可以发现系统分析和设计中的错误。如安全测试是测试安全措施是否完善，能不能保证系统不受非法侵入。再例如，压力测试是测试系统在正常数据量以及超负荷量(如多个用户同时存取) 等情况下是否还能正常地工作。

### 5.3.1 测试环境

系统的测试环境我们通过提供处理器，内存，磁盘空间，网络配置，操作系统，测试软件等相关参数，在开发的过程中因为先是搭建的本地服务器，因此是在 100MB 的局域网下进行测试的，操作系统因为用的是苹果电脑，所以操作系统系统是 MacOS。测试的配置环境如下图 5.2 所示

表 5.2 系统测试环境

部件	WEB 服务器端	数据库服务器	测试客户端
处理器	1.7GHz Inter	1.7GHz Inter	1.7GHz Inter
内存	4GB 1333 MHz	4GB 1333 MHz	4GB 1333 MHz
硬盘空间	128G	128G	128G
网络配置	100MB 局域网络		
操作系统	MacOS	MacOS	MacOS
测试软件	N/A	N/A	Google chrome

### 5.3.2 测试方法

根据各功能模块特点，对部分主要功能场景进行的测试。主要对本系统主要功能的运行状态，运行结果进行监控和分析，可以有效地分析软件的运行性能。可以采用手动编写的测试用例以及用户提交的源代码，执行测试用例及跟踪管理的方式进行。

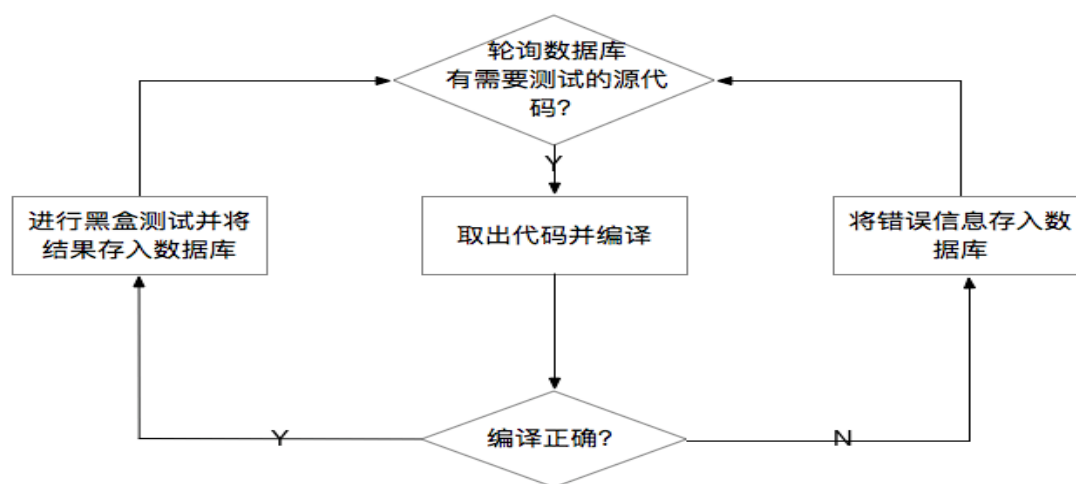


图 5.10 黑盒测试流程图

此外，本系统使用的是先白盒测试后黑盒测试对用户提交的代码进行评测，并给出评测结果。用这种方法进行测试时，可以将被测程序当作看不见的黑盒子。在不考虑内部结构和内部特性的情况下，测试者仅依据程序功能的需求规范，考虑测试用例和推断测结

果的正确性。

特意地，在程序评测的过程中，可能出现用户和测试用例给出的输出结果不一致的现象，但其实用户的答案并没有错，这是因为测试输出用例和运行程序的输出结果之间存在着换行符和空格的差异。因此，在此黑盒测试的基础上，要先进行空格和换行符的去除，在进行结果的比对。

### 5.3 本章小结

本章总共从系统编码实现和测试两个模块进行了阐述。系统编码实现主要对各个功能模块的代码实现以及主要的核心代码进行了说明。同时，对系统的测试方法进行了说明。通过编码实现和系统测试将会对后面代码的重构和迭代起到很大的作用。

## 第 6 章 总结与展望

本文设计了基于功能测试的 B/S 的编译原理的辅助在线测评系统，充分利用了 Java 语言良好的跨平台特性，可移植性等特点以及 B/S 结构的优势，建立了具有可扩展性、安全性和可重用性高、便于维护和升级等优点的在线评测系统。

### 6.1 总结

“编译程序在线评测辅助教学系统”从开始选题、开题报告、到系统开发总共耗时半个多学期的时间。通过自己对已有的各大网站的评测系统进行分析和总结，结合我们学校的上课的实际需求，设计和实现了编译程序在线评测辅助教学系统。与传统的手工评测相比，大大地提高了效率和准确程度。主要研究的内容和成果包括如下内容：

（1）题库获得极大的丰富，不仅仅局限于传统的 ACM 算法类的解答，而且增加了词法分析、语法分析、语义分析等算法的题目。

（2）用户提交代码方式的改变。没有拘泥于传统的只是对于一个类基于文本框代码的评测，支持多个类的文件打包评测。

（3）合并编译。本系统不仅仅支持单个算法的评测，而且可以将一个大的算法划分成很小的模块，让用户能够有针对性的练习，用户可以通过提交一部分的代码可以得到整个系统运行的总结果。

（4）在线比赛。作为本系统的另外一个重要模块，在线比赛可以定期的让老师去考察学生的知识掌握情况，而且还可以发布一些比赛，提高学生的编程的积极性和能力。

（5）题库管理。系统还支持题库和比赛的管理功能，管理员通过授权给老师可以进行方便的编辑题目和比赛。

（6）在线教学。管理员通过录入学生的课程学号，可以针对于特定课程为学生提供教学视频、课件讲义、作业发布与提交、实验的发布，极大的方便了老师的日常教学和减轻老师的日常教学任务。

本系统能够完成编译程序在线评测的基本功能之外，由于时间仓促加之一切其他的客观因素，导致本系统还存在一些待优化的地方。用户界面还可以做的更加丰富交互性更强，管理员的功能还应该更加完善，支持编译的语言库还应该更加丰富（现在仅支持 C、C++ 和 Java 三种编译的语言环境），多并发的处理应该得到解决，采用线程池对任务进行处理，用户的交流版块还未开发，还不能满足用户之间的线上交流，在线教学因为还没有录入教务处的学生信息数据，导致在线教学的模块学生难以实现登录功能。

本在线评测系统是一个在线程序与算法设计练习、在线比赛和在线教学的一个平台

主要功能包括用户模块，题库模块，评测模块，比赛模块，在线教学模块和管理员模块。该系统提供了大量供学生练习的竞赛题目，学生在线或者本地环境开发打包上传程序代码，系统可以自行编译成可执行文件或者类文件，根据题目所对应的测试用例，进行循环测试，从程序的正确性，程序运行总时间、消耗内存、单个用例所执行的时间程序返回结果等各方面去评测代码，并正确返回所提交代码所得到的结果。

特别需要指出的是，此在线编译辅助教学系统，和以往的在线教学系统不一样的是，此次的评测模式不像以往的那么单一，只能对小型的算法进行评测，而且能对含有几个类文件的一个项目工程进行编译运行，这样就极大地丰富了题目的类型和全面性。同时，题目的题型可以进行模块化，这样对学生的练习更有针对性<sup>[12]</sup>。

## 6.2 展望

“编译程序在线评测辅助教学系统”的开发一方面利用信息系统自动化的特性减轻教学者的工作量，利用信息系统丰富的表现方式吸引学生的注意力，提高他们的学习兴趣；另一方面利用信息系统的挖掘信息本身的内在特性和联系，更好的帮助教学者在错综复杂的教学资源和千变万化的教学对象之间抓住和发现教学活动中的关键时段和关键因素，创造出相对于以往教学方式更加具有交互性，更因材施教的教学方式。

本系统的平台叫做 WHUT Online Judge, 不仅仅局限于传统的 ACM 评测系统，而且能和当下的教学课程紧密地结合起来，可以针对不同的实验课程，提供不同的测试接口，更加丰富了系统在日常教学的作用，可扩展性较强。

先已经完成 1.0 版本的整个系统开发，已经部署上线到服务器，学校学生可以免费注册体验系统的交互环境，进行竞赛题目的练习。亦为我们学校举办编程比赛提供比赛的平台，可以极大提高学生编程的积极性和热情。

## 参考文献

- [1] 尤枫, 史晟辉, 赵瑞莲. 编译程序在线评测系统的实现[J]. 实验室研究与探索, 2010, 29(12): 69-72.
- [2] 陈瑞. 基于 Springboot 高并发 Java Web 开发模式[J]. 电脑编程技巧与维护, 2019(04): 27-30.
- [3] 黄艳秀. 基于 mybatis 的面向数据库自动生成技术[J]. 河南科技, 2014(04): 21-22.
- [4] 乔少杰, 杨燕, 葛永明, 张翠芳, 戴齐. 基于 B/S 架构的多用户在线程序评判系统设计与实现[J]. 计算机工程与科学, 2011, 33(S1): 58-61.
- [5] 梅刚, 林龄, 潘峰. 基于 Web 的在线编译环境设计[J]. 计算机时代, 2015(05): 29-31.
- [6] 何迎生, 罗强. Online Judge 评判内核的设计与实现[J]. 吉首大学学报(自然科学版), 2010, 31(06): 37-39.
- [7] 庄奇东, 王键闻, 张楠, 张爽, 任娜. Online Judge 系统的优化[J]. 计算机系统应用, 2011, 20(08): 115-121.
- [8] 吕宣姣. 软件测试之黑白盒策略的应用[J]. 现代计算机(专业版), 2016(09): 48-50.
- [9] 王鹏, 从波, 李国杰, 匡海燕, 刘静静. 基于 ActiveMQ 消息总线的性能测试方法[J]. 测试技术学报, 2019, 33(02): 147-152.
- [10] 黄洪波. 大规模编程题在线评判技术研究[D]. 华南农业大学, 2016.
- [11] 赵春风. 基于 J2EE 技术 ACM 竞赛程序在线评测系统的设计与实现[D]. 厦门大学, 2013.
- [12] 何爱华, 戚晓明. 一种改进的源代码在线评测系统设计及实现[J]. 长江大学学报(自科版), 2014, 11(01): 36-39.

## 致 谢

流年似水，岁月如歌。大学四年的生活匆匆而过，回首走过的岁月，心中倍感充实。在即将走向社会之际，我想对我的这四年来一直帮助过我的人表达一下我的敬意。

感谢我的毕设指导老师饶老师。从国创项目就开始带我们一直到毕设作业。饶老师渊博的知识、严谨的治学作风、高尚的学术品质、热情的待人态度给我留下了深刻的影响。在本次毕业设计的过程中以及论文编写过程中，能够得到饶老师的指导是我的荣幸。每周一次的例会，您不仅仅会给我们解决毕设过程中所遇到的困难，而且您会根据自身的经验会为我们每一个提供人生的道路选择建议，会关心我们每一个人的就业升学状况。感谢饶老师在我遇到困难时对我细心的指导，感谢饶老师每周总会抽出一两个小时和我们谈心，感谢饶老师能够在百忙之中查阅我论文并给我提出宝贵的意见。

感谢我的母校武汉理工大学在大学四年传授给我专业的计算机知识和做人做事的道理，把我培养成计算机方面的人才。感谢我们和蔼热心的辅导员陶导，为我们的日常事务保驾护航。感谢我的母校带给我大学四年快乐的日子，他将是我最珍贵的回忆。

感谢我的父母和家人。没有你们的支持，没有你们为我做家庭坚强的后盾，我根本没有机会来到这里学习。当我想到父亲骑着摩托车，想到母亲在外打工的情形，我的心中就充满了动力。感谢我的哥哥姐姐们，为我在人生道路的选择上出谋划策，建言献策。

感谢我的大学挚友们，感谢小龙和冷阳，我们在一起度过了最快乐最充实的六个月的考研时光。感谢钰瑾、李杰、玉全我们一起在数次大作业留下的美好时光。感谢春光，前礼给我毕设困惑时的慷慨解惑。感谢小松、叶涛还有余家头的好友们，我们在大一一起度过的美好时光。

在完成毕业之际，对帮助过我的老师和同学表示重新的感谢。限于本人的学识水平，文中难免有不当之处，望各位老师不吝赐教。