

浙江大学

Object-Oriented Programming 中期 Project 报告



学生姓名：	<u>应承峻</u>	学号：	<u>3170103456</u>
学生姓名：	<u>夏林轩</u>	学号：	<u>3170103290</u>
学生姓名：	<u>黄思泳</u>	学号：	<u>3170104180</u>
学生姓名：	<u>陈 欢</u>	学号：	<u>3170102503</u>
学生姓名：	<u>张 淼</u>	学号：	<u>3170105674</u>

2018~2019 秋冬学期 2018 年 11 月

1 项目实现的 IDE、Compiler 和库

1.1 IDE 介绍

本组经过讨论后，决定选择 Visual Studio 2017 作为本次 Project 的 IDE。

Visual Studio 集成开发环境可用于编辑、调试生成代码以及发布应用。集成开发环境（IDE）是一个功能丰富的程序，可用于软件开发的许多方面。除了大多数 IDE 提供的标准编辑器和调试器之外，Visual Studio 还包括编译器、代码完成工具、图形设计器和许多其他功能，以简化软件开发过程。同时 Visual Studio 还支持许多工具，例如 .NET Core 和 ASP.NET Core、Node.js 工具、Data Tools 和 SQL Server 等。

首先，我们考虑到的是 Visual C++ 2017 版本的更佳的 C++14 一致性级别和 CMake 支持，这对于我们编写 C++ 的程序和使用一些开源的第三方库有着极大的帮助。次之，我们知道在多人协作的情况下，对于代码的测试是非常重要的，VS2017 的调试和诊断功能比较强大。另外除了将代码整合之后的调试，整合之前各自代码的单元测试也非常重要，VS2017 可以在编辑器中实现单元测试结果和代码覆盖率的实时可视化，对于我们也是有一定的帮助。再次，Visual Studio 对于多人协作也有着很好的支持，虽然我们不一定会用得上，但是它团队资源管理器中的新增 Git 功能并增强了连接体验，在 Visual Studio 2017 中可使用 Git 完成更多工作，并可轻松找到要连接到的项目和存储库。最后，我们考虑到一些高级功能的需求可能会涉及 Microsoft Office 等软件，所以使用同一家公司出品的 IDE 可能会得到一些相关的库支持，更有利于我们进行开发。

以下是网上对该 IDE 的一些摘要性质的介绍：

- **开发：快速导航、编写并修复代码**
 - 新的安装体验：降低了最小内存需求量以实现更快、更定制化的安装，并且支持脱机安装。
 - VisualStudioIDE：大幅改进了 VisualStudio2017，包括减少启动和解决方案加载时间、改进登录和标识、改进代码导航以及增添打开文件视图和链接的服务，实现了应用和任意本地服务或云端服务之间的连接。
 - VisualC++：版本 2015 和 2017 之间没有重大更改。更佳的 C++14 一致性级别、CMake 支持、生成吞吐量和编辑器效率。

- **C#和 VisualBasic:** 扩展对新的 C#7.0 和 VisualBasic15 语言功能的支持, 引入新的重构以便能够通过一个操作来整理源代码。
- **F#:** 支持 F#4.1 语言功能、对 F#编译器和 FSharp.Core 的改进, 以及基于 Roslyn 工作区的新编辑器。
- **JavaScript 和 TypeScript:** TypeScript2.1 可用于 VisualStudio 中的所有 TypeScript 项目。提供新的 JavaScript 语言服务, 并默认启用。
- **调试: 轻松调试、探查和诊断**
- **调试和诊断:** 对异常帮助器进行了大改造, 且可以通过“运行时单击”更快地导航代码。此外,“诊断工具”窗口中增添了应用程序事件摘要, 并对 CPU 工具实施了多项改进。
- **实时单元测试:** 在编辑器中实现单元测试结果和代码覆盖率的实时可视化。
- **测试工具:** 使用测试资源管理器将自动化与测试用例工作项关联。
- **团队资源管理器中的新增 Git 功能并增强了连接体验:** 在 VisualStudio2017 中可使用 Git 完成更多工作, 并可轻松找到要连接到的项目和存储库。
- **项目和项模板的可发现性:** 突破性地改进了 VisualStudio 使用项目和项模板的方式, 提升了模板发现的性能。

1.2 Compiler 介绍

本项目使用的 C/C++编译器为 Visual Studio 自带的编译器 cl.exe。

cl.exe 是一个控制 Microsoft C 和 C++编译器和链接器的工具。

编译器生成通用对象文件格式 (COFF) 对象 (.obj) 文件。链接器生成可执行 (.exe) 文件或动态链接库 (DLL)。

1.3 库介绍

1.3.1 C++标准库所需部分类与子库

<new>: 支持动态内存分配。

<iostream>: 支持标准流 cin、cout、cerr 和 clog 的输入和输出, 还支持多字

节字符标准流 `wcin`、`wcout`、`wcerr` 和 `wclog`。

`<string>`: 为字符串类型提供支持和定义, 包括单字节字符串 (由 `char` 组成) 的 `string` 和多字节字符串 (由 `wchar_t` 组成)。

`<sstream>`: 支持字符串的流输入输出。

`<fstream>`: 支持文件的流输入输出。

`<ctime>`: 支持系统时钟函数。

`<locale>`: 提供的本地化包括字符类别、排序序列以及货币和日期表示。

`<cstdlib>`: 声明 C 标准库函数 `bsearch()` 和 `qsort()`, 进行搜索和排序。

1.3.2 QT 所需函数

`QPainter::Antialiasing` 在可能的情况下, 反锯齿。

`QPainter::TextAntialiasing` 在可能的情况下, 文字反锯齿。

`QPainter::SmoothPixmapTransform` 采用平滑的 Pixmap 变换算法。

`drawLine()`: 提供起始坐标绘制直线。

`drawRect()`: 提供左上顶点及长和宽绘制矩形。

`drawRoundRect()`: 提供右上顶点及宽和高绘制圆角矩形。

`drawEllipse()`: 提供椭圆外接矩形的左上顶点及长和宽绘制椭圆。

`drawText()`: 提供起始点写入文本。

`setPen(QPen(Qt::blue,4))`: 设置画笔形式。

`setPalette(QPalette(Qt::white))`: 设置背景颜色。

`mousePressEvent(QMouseEvent *event)`: 鼠标单击

`mouseReleaseEvent(QMouseEvent *event)`: 鼠标释放

`mouseMoveEvent(QMouseEvent *event)`: 鼠标移动

2 项目需要完成的功能目标

2.1 基本功能

2.1.1 试题的分类、录入和维护

在基本功能中, 所有的试题将被分为两类, 一类是选择题 (包括单项选择题和多项选择题), 另外一类是判断题。其中, 选择题给出 ABCD 共四个选项, 判断题给出 T 和 F 两个选项。试题在出题页面录入时, 首先需要选择题型, 然

后根据题型对试题的相应信息进行录入：

- 选择题：选择题的相应信息应当包括试题的题目描述、A 选项描述、B 选项描述、C 选项描述、D 选项描述、选择题类型（单选、多选）、题目答案、试题分值、试题作者等。
- 判断题：判断题的相应信息应当包括试题的题目描述、题目答案、试题分值、试题作者等

试题录入完毕后，点击“提交”按钮进行提交，在基础功能中，试题的数据将以文件形式保存在本地。保存后试题将被归入题库。

在题库界面中，试题以一定的顺序呈现，也可以通过筛选方式（按作者、试题类型、试题分值、题目/选项描述关键字等），每题显示一行，点击某题目中“修改”按钮后进入试题录入界面，该界面会渲染中当前待修改试题的信息，对试题修改后提交即可完成修改。同时试题也可以通过“删除”按钮进行删除。

2.1.2 手动选择生成试卷

手动选择生成试卷功能通过试卷生成界面实现，生成试卷时，需要录入试卷名称、试卷编号（系统自动生成）、可答题起始时间、可答题终止时间、最长答题时间、以及试卷说明。之后可以从题库中选择试题添加到该试卷（筛选方式同试题录入），也可以删除已经选择的试题。试题选择完毕后点击“提交”按钮进行提交。

2.1.3 题目呈现

考生对某试题点击进入考试时，会进入考试界面，按照事先设定好的顺序呈现出题目。考生需在此界面中进行作答，作答完毕后进行提交。

2.1.4 考试环境的提供

考试界面中提供计时器，可以让考生知道考试剩余时间。同时离考试截止不足 5 分钟时会弹出小窗口提示考试即将结束。考试时间截止或考生点击提交时考试所有试题的答案将被提交到后台。在基础功能中，数据仍以文件形式保存。

2.1.5 试题评分

答题后，选择题和判断题部分由系统根据题库中事先设定好的答案进行比对，并计算最终得分，得分将会以得分/满分的形式显示在用户的成绩查询界面中。其中不定项选择题的比对方式为只有完全正确才得分（即多选、少选、错选

均不得分)

2.1.6 成绩查询

在成绩查询页面中，考生可以看到本次考试的得分，排名以及自己每一道题的得分，同时会显示自己的作答情况，即：你的答案/标准答案/本题得分。

2.2 扩展功能

2.2.1 用户交互界面

启动程序后，系统将进入登录界面，用户需输入用户名和密码，同时选定登录身份（管理员、考生）后，系统将综合用户名密码和身份进行数据比对（此时数据仍然保存在本地文件中）。登录成功后，进入用户主界面，第一次进入主界面时用户需要完善自己的个人信息（例如姓名、学号、手机号、邮箱等等，其中学号、手机号、邮箱等数据需要进行正则表达式判断数据的合法性），完善成功后，主界面将显示用户的个人信息，所有考试（列表展现：考试名称，进行状态，进入考试按钮，成绩查询按钮）。

2.2.2 填空题实现

扩展功能提供了管理员可以在题库中添加填空题的功能，相应地，在添加填空题时，管理员需要录入题目描述，题目答案，题目分值以及作者。填空题的批改方式为全文比对，只有与答案完全相符才可以得分。

2.2.3 出题和答题状态保存

为了防止意外所导致的数据丢失，扩展功能为管理员和考生提供了创建试卷和考试时的状态保存功能。用户可点击相应页面的“保存结果”按钮对当前组卷或做题状态进行保存。同时系统将在距离考试结束不足 5 分钟时自动对当前答题状态进行保存。

2.2.4 设定显示批改结果的时间

考虑到同一场考试的公平性，扩展功能为管理员提供了设定显示批改结果时间的功能，管理员可在创建试题界面选择批改结果显示给考生的时间：

- ①考生提交时立即显示提交结果
- ②考试时间（有效答题的终止时间）截止后显示提交结果
- ③自由设定显示结果的时间，精确到分（即年-月-日-时-分）

2.2.5 管理员查看考试数据

为了便于管理员对考生作答的答题情况进行分析，系统为管理员提供了查看某一场考试数据的功能，在该界面中，参加本场考试的考生将以排名为关键字进行升序排序，同时将显示考生排名/考试姓名/考试得分/单项选择题得分/不定项选择题得分/判断题得分/填空题得分。当有考试得分相同的考生时，相同分数的考生排名将被记为同一名次。除此之外，管理员还可以通过点击考生姓名来查看考生每一小题的得分（通过模态框或新界面显示）

2.3 高级功能

2.3.1 数据存储于数据库

高级功能中，为了严格模拟考试系统的功能，所有数据将不再存储于文件中，而是存储在数据库中。在本项目中，我们约定采用 SQL 语言进行数据库操作。

2.3.2 主观题实现

高级功能提供了管理员可以在题库中添加主观题的功能，在添加主观题时，类似于填空题，管理员需要录入题目描述，题目分值以及作者。主观题的批改则由管理员在主观题批改页面中进行人工打分。当主观题没有批改时，考生在查看成绩时将显示“未批改”

2.3.3 试卷分权限查看

为了保证考试的公平性。每名学生不能在任意的时间打开任意的试卷，只能在规定的考试时间打开自己参加的考试试卷。因此，在考生的试题列表中，只显示满足以下条件的试题①自己有权参加的考试②时间已到达管理员事先设定的考试起始时间。

2.3.4 管理员以 Excel 方式导出考生数据

高级功能中，为了方便后期的数据保存和使用，管理员可以将所有考生的数据导出到 Excel 表格中。在本项目中，我们将会使用命名空间 Excel 中的相关对象来实现 Excel 的导出。

3 项目实现的基本思路

3.1 开发流程

3.1.1 前期规划

经过小组的讨论，我们决定选择“试卷生成和考试评分系统”这一项目，并进行细致分工。

3.1.2 需求分析

经讨论，我们需要实现试题的分类、录入和维护等 5 项基本功能，用户交互界面等 5 项扩展功能，数据存储于数据库等 4 项高级功能。

3.1.3 项目设计

根据项目需求建立项目总体结构和模块间的关系，确定各模块功能，定义各功能模块的接口，设计全局数据库和数据结构；然后进行细节的编程。

以“考试”为核心模块，“试题”“试卷”“考生”“管理员”等为子模块。

3.1.4 功能测试

逐一测试每项功能的实现情况，确保项目功能的完善。

3.2 页面架构

- **登录界面：**用于实现用户的登录验证功能
- **信息完善界面：**引导考生或管理员进信息的完善
- **考生主界面：**主界面用于显示考生个人信息、可参加的考试等基本信息
- **考试界面：**为考生提供考试环境
- **成绩查询界面：**考试可在此进行考试成绩以及答题情况等信息的查询
- **管理员主界面：**管理员主界面用于引导管理员进入题库界面、添加试题界面、试卷生成界面、查看考试数据界面，
- **题库界面：**管理员可在该页面查看题库中的试题，同时可进行添加试题、删除试题以及修改试题的操作
- **添加/修改试题界面：**管理员可在此实现添加/修改试题功能
- **试卷生成界面：**管理员在此界面可进行组卷操作
- **查看考试数据界面：**显示所有考生的答题情况

3.3 类的架构

3.3.1 试题模块

```
ObjectQuestion 选择题类 {  
    QuestionCode    //试题编码  
    Type            //选择题类型  
    Description      //试题描述  
    ChoiceA         //A 选项描述  
    ChoiceB         //B 选项描述  
    ChoiceC         //C 选项描述  
    ChoiceD         //D 选项描述  
    Answer          //答案  
    Value           //分值  
    Author          //作者  
}
```

```
Blank 填空题类 {  
    QuestionCode    //试题编码  
    Description      //试题描述  
    Answer          //答案  
    Value           //分值  
    Author          //作者  
}
```

```
Judge 判断题类 {  
    QuestionCode    //试题编码  
    Description      //试题描述  
    Answer          //答案  
    Value           //分值  
    Author          //作者  
}
```

```
SubjectQuestion 主观题类 {
```

```

    QuestionCode    //试题编码
    Description      //试题描述
    Value           //分值
    Author          //作者
}

```

3.3.2 考试模块

```

Exam 试卷类 {
    ExamCode    //试卷编码
    ExamName    //试卷名称
    StartTime   //起始时间
    EndTime     //终止时间
    Duration    //限定时间
    Information  //试卷信息
    Question    //包含试题
    Status      //考试答题情况
}

Status 考生答题情况类 {
    Username    //考生用户名
    ExamCode    //试卷编码
    IsSubmit    //是否已经完成答题
    ObjectChoice //单选题答案
    MultiChoice  //不定项选择题答案
    JudgeAnswer  //判断题答案
    BlankAnswer  //填空题答案
    SubjectAnswer //主观题答案
    ObjectScore  //单选题得分
    MultiObjectScore //多选题得分
    JudgeScore   //判断题得分
    BlankScore   //填空题得分
}

```

```

        SubjectScore    //主观题得分
        TotalScore      //试卷得分
    }

```

3.3.3 用户模块

```

User 考生类: {
    Username    //用户名
    Password    //密码
    StudentId   //学号
    PhoneNumber //电话号码
    Email       //邮箱
    Exam        //考试
}

Admin 管理员类: {
    Username    //用户名
    Password    //密码
    Exam        //创建的考试
}

```

4 人员职责

角色	姓名（学号）	职责
组长	应承峻 3170105674	✧ 总体策划，负责协调活动 ✧ 负责召集组内成员学习讨论 ✧ 考生部分代码实现
组员	夏林轩 3170103290	✧ IDE 下载、安装、调试； ✧ 库的下载、安装、使用； ✧ 模块代码实现
组员	黄思泳 3170104280	✧ Project 总体架构设计； ✧ 模块代码实现

		✧ 程序单元测试
组员	陈 欢 3170102503	✧ 管理员部分代码实现； ✧ 图形界面美化
组员	张 淼 3170105674	✧ 模块代码实现 ✧ Project 总体测试； ✧ Project 中期报告撰写。

5 附录

5.1 线下讨论照片



5.2 会议记录

OOP 中期有关选题分工确定商讨会议

会议时间：2018 年 11 月 10 日

会议地点：月牙楼

会议出席人员：应承峻、夏林轩、黄思泳、陈欢、张淼

会议内容：

一、讨论确定 project 的 IDE、编译器、库

经过大家的商讨，我们一致同意：

1.IDE 选择 Visual Studio 2017

2.compiler 选择 Visual Studio 2017 自带的 C/C++编译器--cl.exe

3.除了 C++的常用库外，我们选择了 QT 作为拓展库以用于实践。

二、讨论确定选题

基于备选题以及发动大家的想象力，我们将试卷生成和考试评分系统、图形编辑器、图像处理工具软件作为备选项，对其基本功能、技术要素、高级拓展功能进行了探讨，最后一致决定选择试卷生成和考试评分系统作为我们 project 的选题。

我们认为试卷生成和考试评分系统有更大的发挥空间，在基本功能之上，有较多可以作为目标的高级功能，可以拓宽我们的技术栈，丰富我们的知识以及拓展我们的技术。

三、确定组长以及组员的任务分工 规划时间线

讨论决定由应承峻担任此次项目的组长，根据组员的个性以及技术特长经验，基本初步确定了人员分工（详情见上表）。

四、选题重点细节的讨论确定

1.基本框架的确定

2.基本功能的确定

3.高级拓展功能的提出

4.主要页面的确定

5.有关数据库的大致框架的确定

6.类的架构的确定