**实验报告**

**实验名称：** 口算习题出题程序项目

**院（系）：** 软件学院

**专业班级：** 软件工程20-01

**小组成员：** 张赛威 张艺龙 张志毅

赵浩勃 赵艳威 周昊

**指导教师：**  刘永文 王文冰

**学年学期：** 2021-2022学年第1学期

**成绩：**

目录

[1 引言 1](#_Toc101460350)

[1.1 背景 1](#_Toc101460351)

[1.2 定义 1](#_Toc101460352)

[1.3 参考资料 1](#_Toc101460353)

[2 项目概括 2](#_Toc101460354)

[2.1 项目背景 2](#_Toc101460355)

[2.2 项目任务 2](#_Toc101460356)

[2.3 项目目标 2](#_Toc101460357)

[3 系统用例图 3](#_Toc101460358)

[4 系统功能需求 4](#_Toc101460359)

[4.1 生成数字 4](#_Toc101460360)

[4.2 生成符号 4](#_Toc101460361)

[4.3 需要50道 5](#_Toc101460362)

[5 非功能性需求 6](#_Toc101460363)

[5.1 用户界面 6](#_Toc101460364)

[5.2 测试结果导出 8](#_Toc101460365)

[5.3 草稿纸 9](#_Toc101460366)

[5.4 饼图分析 10](#_Toc101460367)

[5.5 硬件环境 11](#_Toc101460368)

[5.6 软件环境 11](#_Toc101460369)

[5.7 性能需求 12](#_Toc101460370)

[5.8 系统构架 12](#_Toc101460371)

[6 随机数生成预循环（F1）模块 14](#_Toc101460372)

[6.1 程序描述 14](#_Toc101460373)

[6.2 功能 14](#_Toc101460374)

[6.3 基本功能图 14](#_Toc101460375)

[6.4 具体实现 15](#_Toc101460376)

[7 界面（F2）模块 16](#_Toc101460377)

[7.1 程序描述 16](#_Toc101460378)

[7.2 功能 17](#_Toc101460379)

[7.3 流程图 17](#_Toc101460380)

[7.4 具体实现 18](#_Toc101460381)

[8 目标与范围 21](#_Toc101460382)

[8.1 测试目标 21](#_Toc101460383)

[8.2 测试范围 21](#_Toc101460384)

[8.3 性能需求 21](#_Toc101460385)

[8.4 测试输出 22](#_Toc101460386)

[9 测试资源 23](#_Toc101460387)

[9.1 测试分类 23](#_Toc101460388)

[9.2 测试工具 24](#_Toc101460389)

[9.3 测试环境 24](#_Toc101460390)

# 1 引言

## 1.1 背景

项目名称 : 50道100以内的加、减混合运算口算习题随机生成系统

用 户 : 小学生及其家长和老师

开 发 者 ：项目组第二组

## 1.2 定义

本程序可以从根据用户的需求随机生成一定数量的口算练习题。改程序不仅能生成口算习题，而且还能由系统进行自我批改，之后用户能够查看自己的完成情况，并由饼图的方式呈现给用户，最后用户可以根据需要进行练习题的导入和导出。

## 1.3 参考资料

参考资料表如表1-1所示。

**表1-1 参考资料**

|  |  |
| --- | --- |
| 参考资料 | 备注 |
| 《Java 8编程入门官方教程》 | 清华大学出版社 |
| 《深入浅出面向对象分析与设计》 | 电子工业出版社 |
| 《深入浅出设计模式》 | 东南大学出版社 |

# 2 项目概括

使用人员可以通过此程序来随机生成一定数量的加减口算练习题，并且能够实现试题的导入和导出，错题统计，以及查看成绩统计。

## 2.1 项目背景

为了减轻小学幼儿园老师及儿童家长的负担，使老师、家长更加快速准确地给儿童出作业题，并且软件简单易学习能让老年人快速学习及使用。该软件还能让儿童自己平时练习与巩固。

## 2.2 项目任务

编写一个可以生成一定数量的加减口算习题出题程序项目。

## 2.3 项目目标

1.需要一个登陆界面,方便不同用户使用该习题自测系统。

2.该软件有草稿纸功能，方便用户做简单记录。

3.有一个可以显示题目的界面。

4.每道题生成的两个数字必须在0~100之间。

5.所有的题目可以是加、减或者混合运算。

6.要达到50道题。

7.要实现自动批改，避免增加使用者工作负担。

8.可以实现习题的导出，方便用户线下多次查看。

9.用饼图显示一个人练习多套习题的结果，便于分析。

# 3 系统用例图

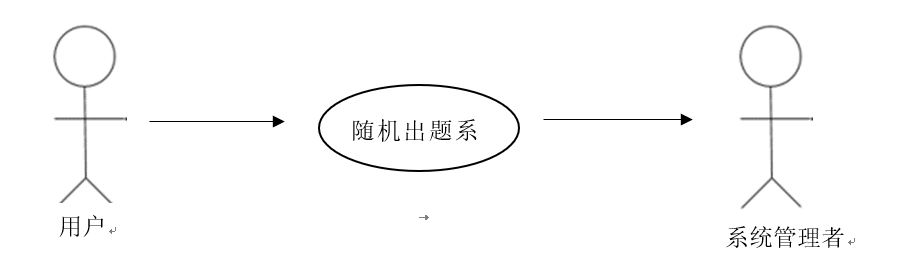
用例图(Use Case Diagram)是由软件需求分析到最终实现的第一步，它描述人们如何使用一个系统。用例视图显示谁是相关的用户、用户希望系统提供什么样的服务，以及用户需要为系统提供的服务，以便使系统的用户更容易理解这些元素的用途，也便于软件开发人员最终实现这些元素。用例图在各种开发活动中被广泛的应用，但是它最常用来描述系统及子系统。

当用例视图在外部用户出现以前出现时，它捕获到系统、子系统或类的行为。它将系统功能划分成对参与者（即系统的理想用户）有用的需求。而交互部分被称作用例。用例使用系统与一个或者多个参与者之间的一系列消息来描述系统中的交互。

用例图包含六个元素，分别是：参与者(Actor)、用例(Use Case)、关联关系(Association)、包含关系(Include)、扩展关系(Extend)以及泛化关系(Generalization)。

使用者利用口算习题随机生成系统参与系统的交互，由用户启动交互，从而随机出题系统被启动，使系统管理者也被启动。利用系统用例图可以：（1）获取需求；（2）指导测试；（3）还可在整个过程中的其它工作流起到指导作用。

系统用例图如图3-1所示：



**图3-1 系统用例图**

# 4 系统功能需求

为了减轻小学幼儿园老师及儿童家长的负担，使老师、家长更加快速准确地给儿童出作业题，针对以上问题，本系统做出了设计与改进。

在程序设计过程中,该程序是要实现简单加减乘除 运算等功能。还要实现数据输入,输出,计算,显示及程 序退出等功能。用户提交查询时,得到返回结果延时不得超过 5秒 提交数据录入是,得到结果延时不得超过 5秒。

## 4.1 生成数字

功能描述：可以从0~100中随机选取两个数字。

生成数字环节的用例图如图4-1所示：

用户

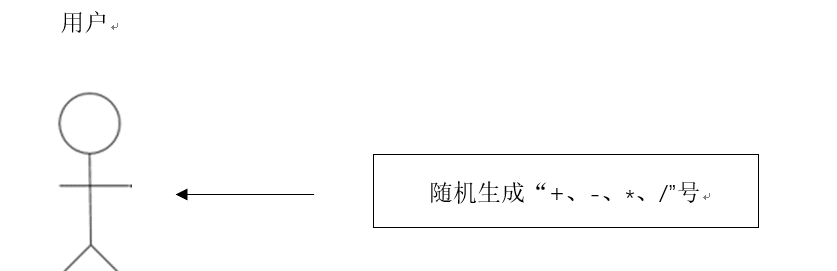
从0-100中随机生成两个数字

**图**4-1 **生成数字环节的用例图**

## 4.2 生成符号

功能描述：随机地从四个运算符“+”、“-”、“\*”、“/”中挑选一个。

生成符号环节的用例图如图4-2所示：



**图4-2 生成符号环节的用例图**

## 4.3 需要50道

功能描述：前两个步骤可循环50次来获得50道口算题。

生成50道题的用例图如图4-3所示：

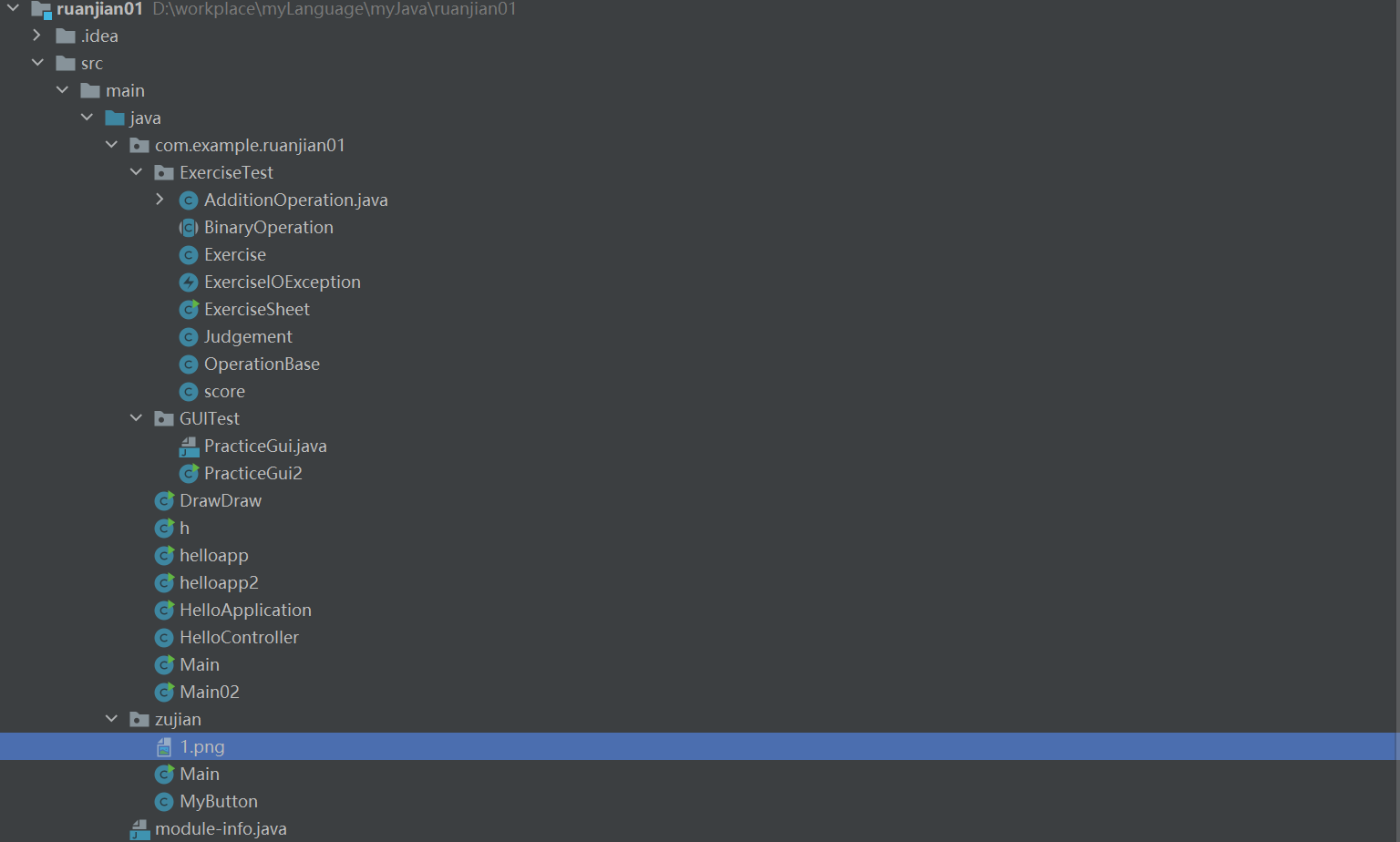


**图4-3 生成50道题的用例图**

**4.5 项目结构**

功能描述：根据设计好的类去实现相关功能。

项目结构如图4-4所示：



**图4-4 项目结构图**

# 5 非功能性需求

非功能性需求，是指软件产品为满足用户业务需求而必须具有且除功能需求以外的特性，包括安全性、可靠性、互操作性、健壮性等。在需求分析时，功能性需求是人们普遍关注的，但也不能忽视非功能性需求的分析，因为它所涉及的方面比较广泛。

本套练习题主要要以界面的形式输出，而且服务对象主要是小学生，所以界面要求美观、大方，并且界面要求通俗易懂易于操作。

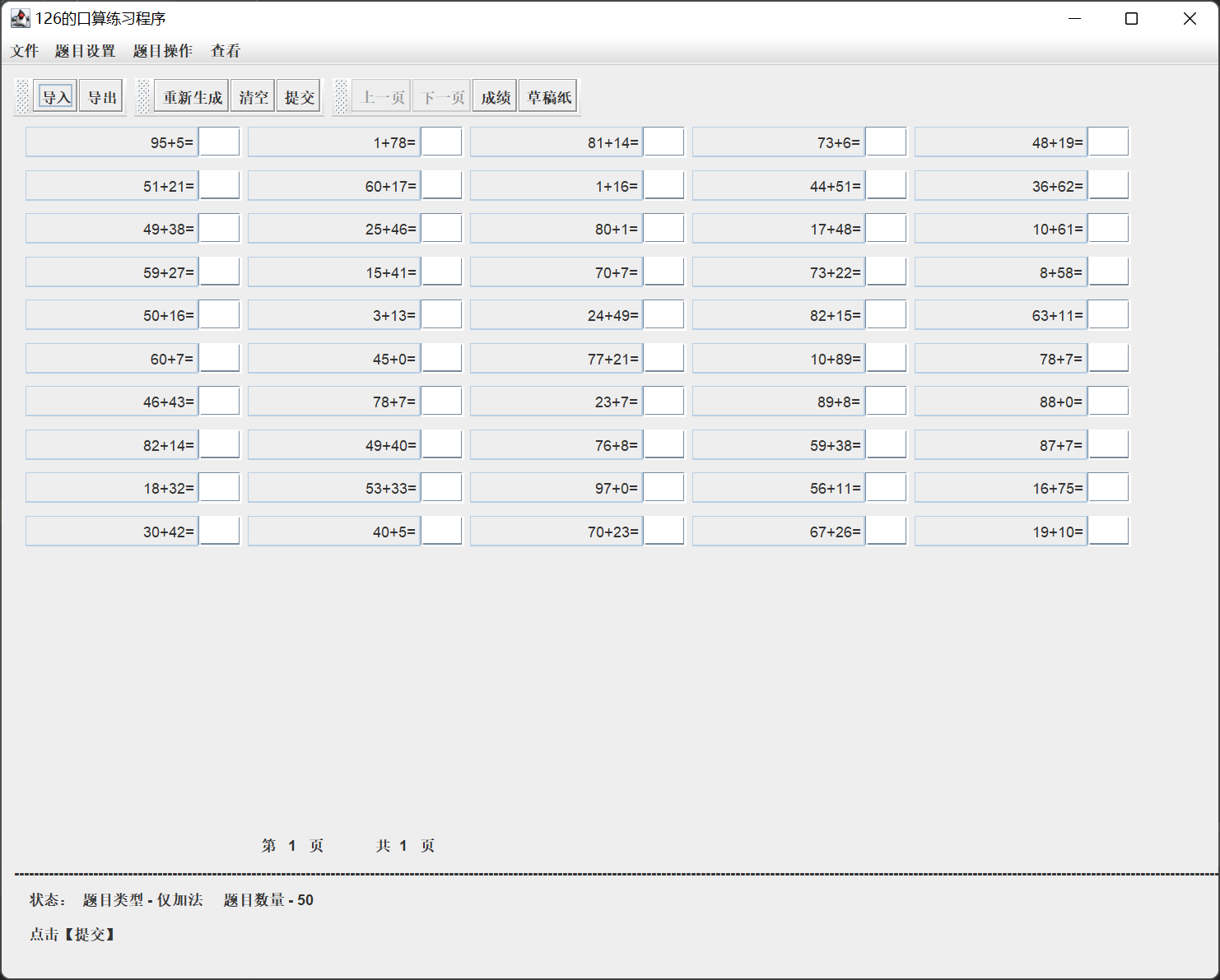
## 5.1 用户界面

[用户](https://baike.baidu.com/item/%E7%94%A8%E6%88%B7)界面(User Interface)是指对[软件](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6/12053)的人机交互、操作逻辑、界面美观的整体设计。好的UI设计不仅是让[软件](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6/12053)变得有个性有品味，还要让软件的操作变得舒适、简单、自由、充分体现软件的定位和特点。

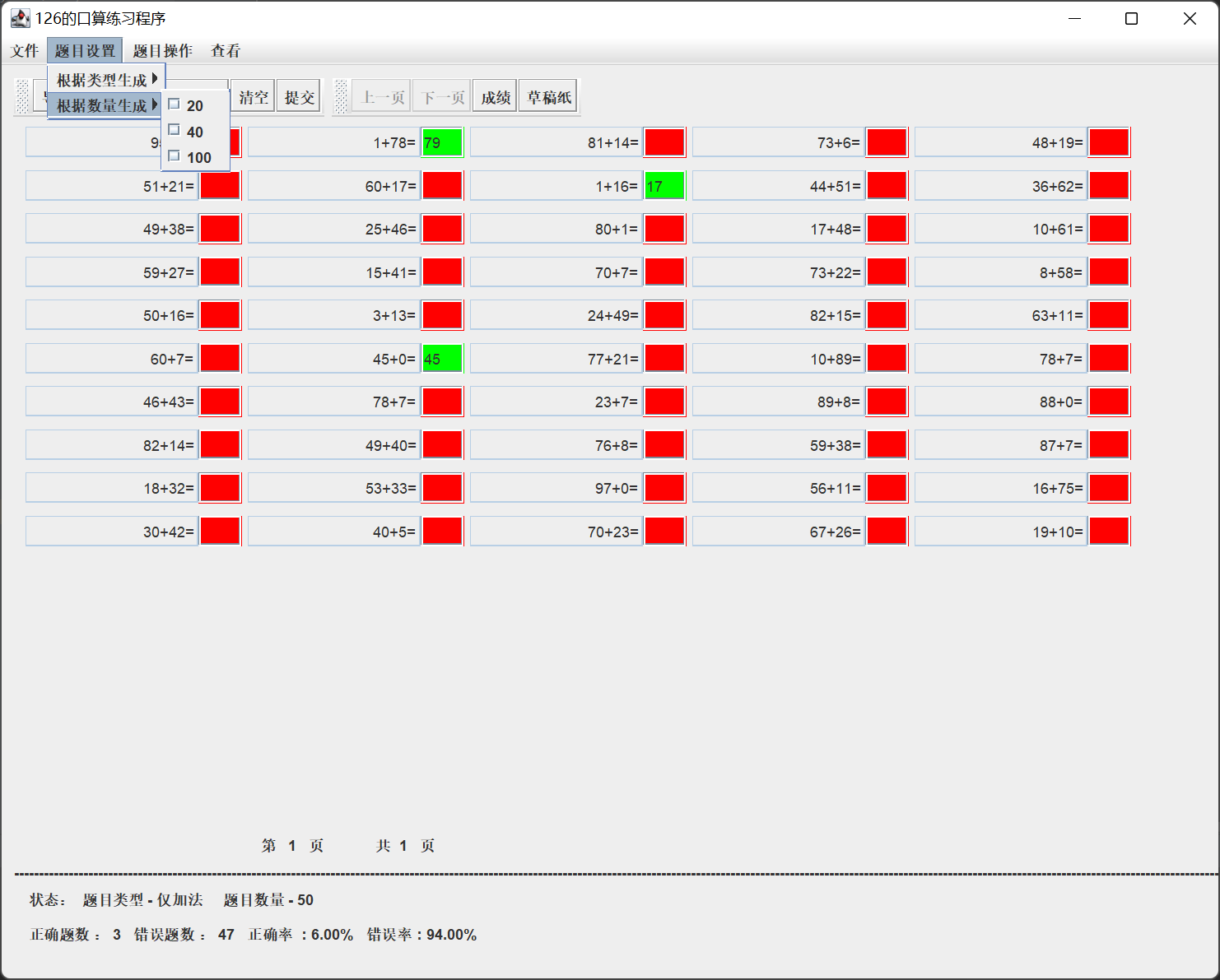
用户界面（User Interface，简称 UI，亦称使用者界面）是系统和用户之间进行交互和信息交换的媒介，它实现信息的内部形式与人类可以接受形式之间的转换。用户界面是介于用户与[硬件](https://baike.baidu.com/item/%E7%A1%AC%E4%BB%B6/479446)而设计彼此之间交互沟通相关[软件](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6/12053)，目的在使得用户能够方便有效率地去操作硬件以达成双向之交互，完成所希望借助硬件完成之工作，用户界面定义广泛，包含了[人机交互](https://baike.baidu.com/item/%E4%BA%BA%E6%9C%BA%E4%BA%A4%E4%BA%92/61313)与图形用户接口，凡参与人类与机械的信息交流的领域都存在着用户界面。

在系统用户界面如图5-1可以自由选择运算类型（+、-）以及选择运算题目的数量（20、40、100），当完成选择后系统自动生成相应数量的习题。

生成习题后用户即可开始做题，做完后点击“提交”按钮系统即可自动批改并及时出分，能够让用户清晰的看到做题的对错情况（在其提交后，做对的题会以绿色背景标出，而做错的题以红色背景标出），及时改错或者是选择重做来提升正确率。选择加法运算后的习题生成界面及批改情况如图5-2所示：



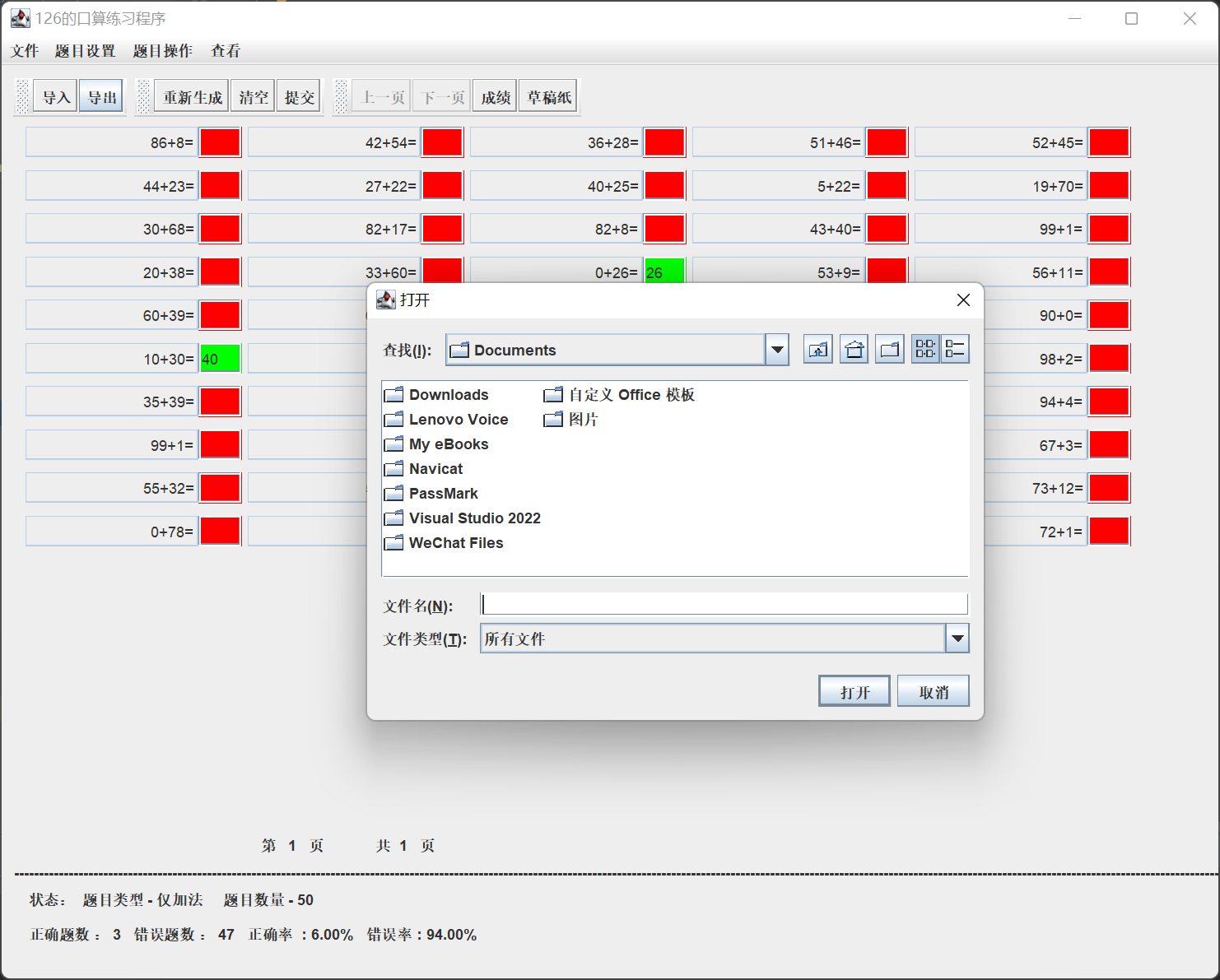
**图5-1 系统主界面图**



**图5-2 系统用户界面**

## 5.2 测试结果导出

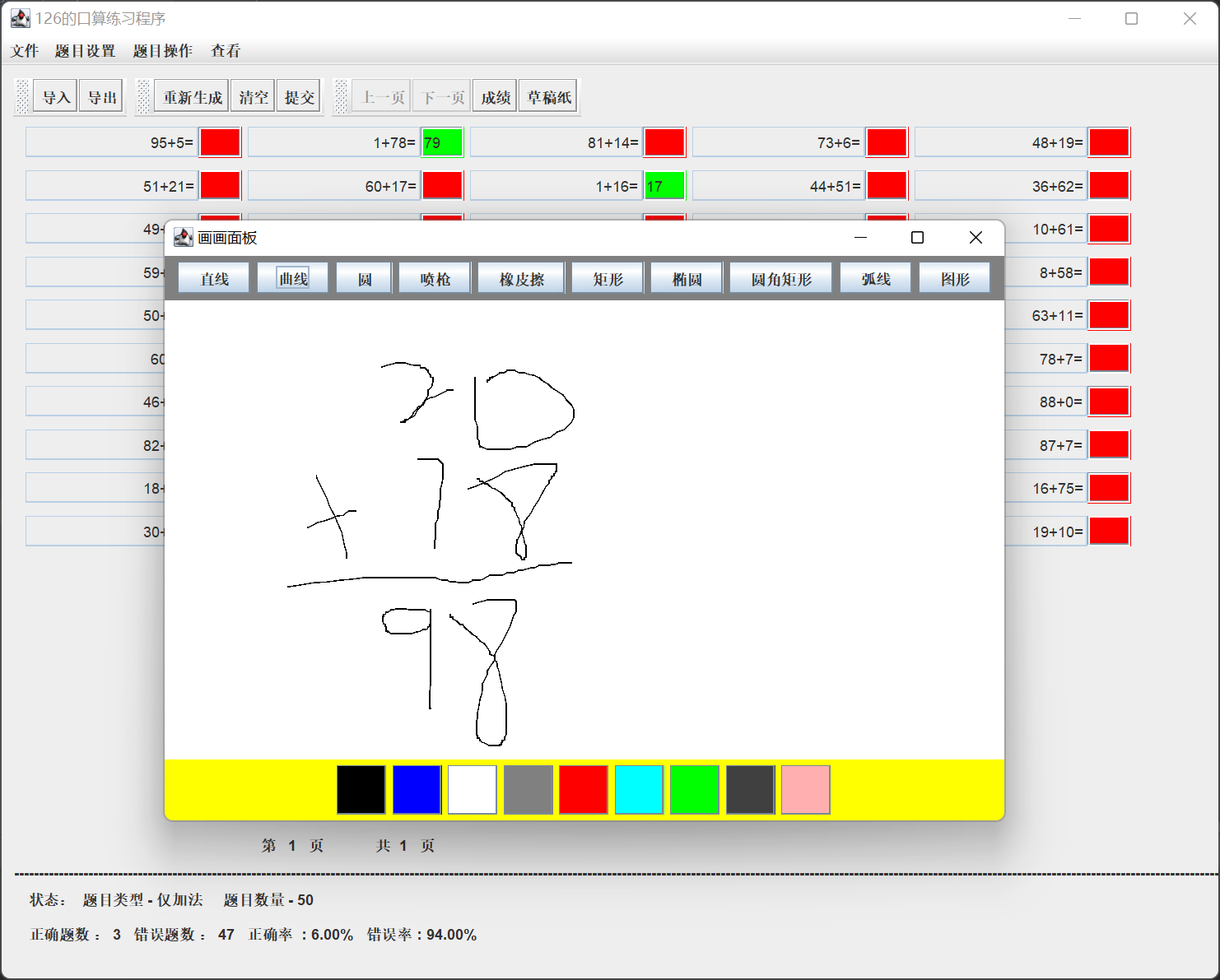
用户还可以选择将测试结果导出，以便以后线下查看以及以后导入回顾。导出习题功能如图5-3所示：



**图5-3 导出习题界面**

## 5.3 草稿纸

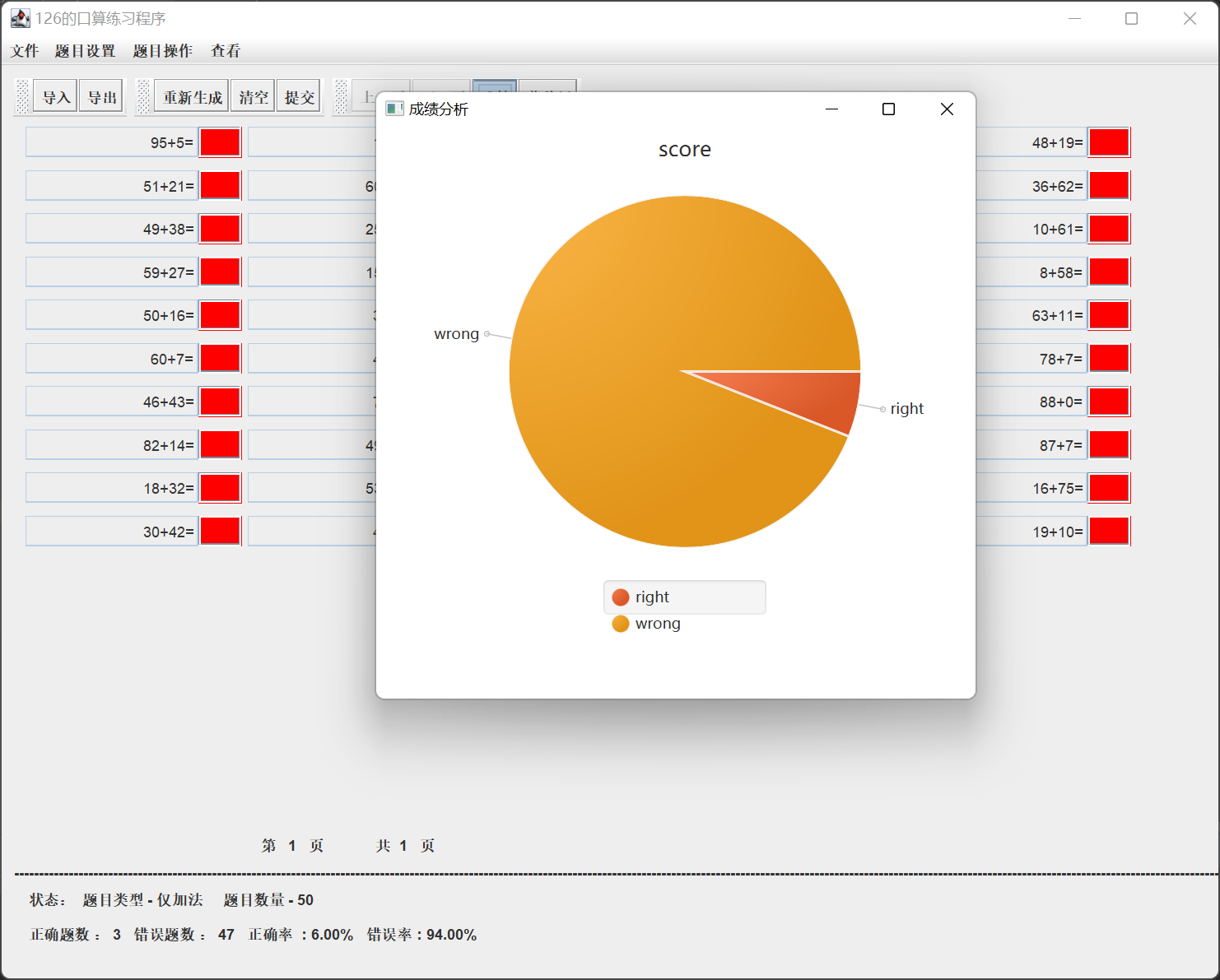
本软件主要是帮助用户（主要是小学生群体）来提高算数能力，所以，在进行两位数的加减时，可以使用草稿纸功能进行列竖式来进行辅助计算。草稿纸功能如图5-4所示：



**图5-4 草稿纸功能**

## 5.4 饼图分析

为帮助用户更加直观的看到自己做题的结果情况，用户可以选择菜单中的成绩选项，从而生成本次做题结果的饼状图，更直观的看到自己的做题结果。如图5-5所示。



**图5-5 饼图分析功能**

## 5.5 硬件环境

服务器硬件：

要求：

内存：不少与2G

CPU：Intel 2.8以上

硬盘：RAID，160G

## 5.6 软件环境

服务器软件：

IDEA

Window 11

## 5.7 性能需求

（1）快速反应

简单的操作可以在极短的时间内得到响应。

（2）通俗易懂

系统界面简洁易懂，更加方便用户的使用。进入系统，就会默认生成20道随机加法，在题目设置中可进行题目数量以及运算法则的设置。

（3）稳定

系统的稳定性非常重要，它将直接影响到各类用户的使用质量，所以系统必须保证的稳定运行。

## 5.8 系统构架

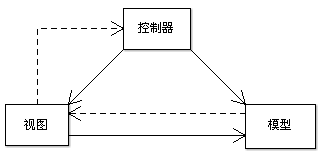
随机出题系统采用了MVC(Model-View-Controller，模型-视图-控制器模式)架构模式。MVC模式的目的是实现一种动态的程式设计，对程序的修改和扩展简化，并且使程序某一部分的重复利用成为可能。除此之外MVC模式通过对复杂度的简化使程序结构更加直观。软件系统通过对自身基本部份分离的同时也赋予了各个基本部分应有的功能。专业人员可以通过自身的专长分组：

控制器--负责转发请求，对请求进行处理。

视图--界面设计人员进行图形界面设计。

模型--程序员编写程序应有的功能（实现算法等等）、数据库专家进行数据管理和数据库设计(可以实现具体的功能)。

MVC三层结构图如图5-5所示：



**图5-5 MVC架构图**

软件基于IDEA开发工具，采用JAVA语言。IDEA可以很快速开发基于JAVA技术的各种类型的项目，是当今使用最广泛的集成开发环境(IDE)。另外，IDEA可以无缝地集成各种插件，保证了开发项目的良好扩展性能。

Java不同于一般的编译运行计算机语言和解释执行计算机语言。它首先将源代码编译成字节码，然后依赖各种不同平台上的虚拟机来解释执行字节码，从而实现了“一次编译、到处执行”的跨平台特性。Java编程语言是个简单、面向对象、分布式、解释性、健壮、安全与系统无关、可移植、高性能、多线程和动态的语言。本应用系统以方便用户为原则，在统一的用户界面下提供各种实用帮助，尽可能降低使用后的维护投入；不仅应适用于当前实际的运行环境，而且还具有应变能力，以适应未来变化的环境和需求。

# 6 随机数生成预循环（F1）模块

## 6.1 程序描述

用户在该使用该系统时，要实现算数的功能首先要有随机数生成。换句话说就是本功能就是生成题的功能。在功能中通过Random去产生一系列的随机数。产生一系列随机数后，用户就可以去答题了。当用户答完指定的题目后，在调用Random类就会产生一组新的随机数供用户继续答题。

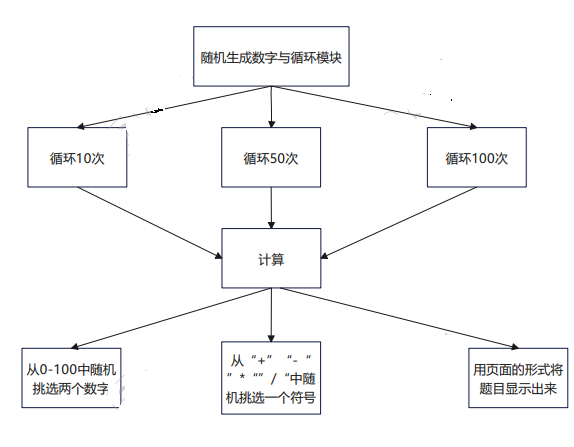
## 6.2 功能

本部分功能主要分成三个部分：生成两个随机数，生成随机符号，题目循环10次、20次、100次。

## 6.3 基本功能图

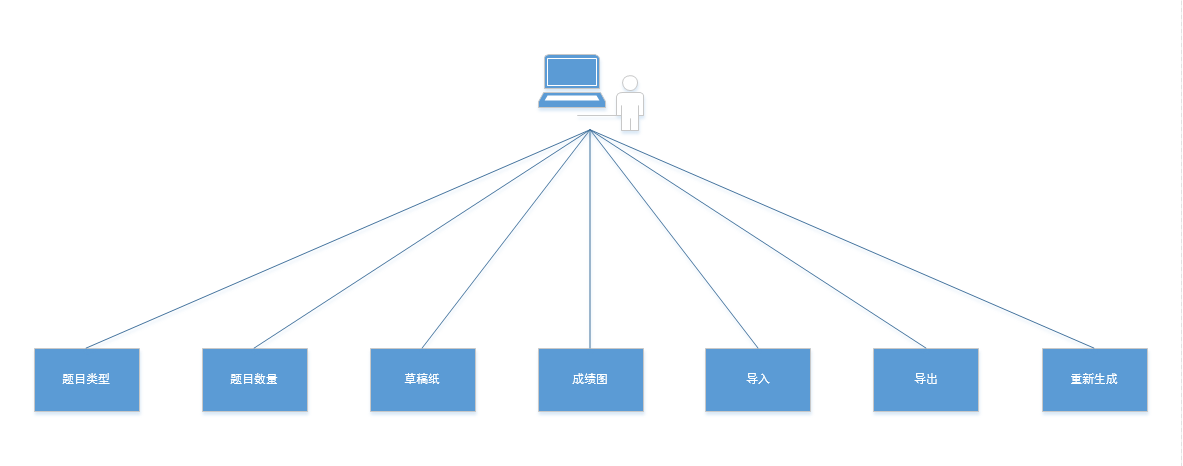
一般情况下产品或系统的总功能可分解为若干分功能，各分功能又可进一步分解为若干二级分功能，如此继续，直至各分功能被分解为功能单元为止。这种由分功能或功能单元按照其逻辑关系连成的结构称为功能结构。分功能或功能单元的相互关系可以用图来描述，表达分功能或功能单元相互关系或从属关系的图称为功能结构图。

功能结构图的定义：以功能模块为类别，介绍模块下其各功能组成的图表。



**图6-1 基本功能**

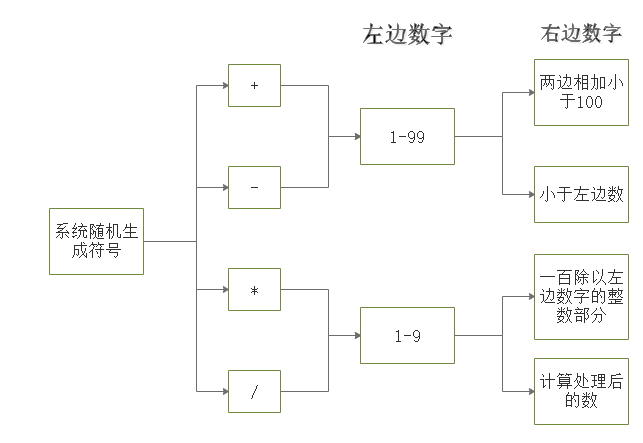
用户设计用例图，明了的显示了程序的开发功能。图6-2所示。



**图6-2系统主要功能图**

## 6.4 具体实现

首先生成一个随机符号，如果随机符号为正号则随机符号左边的数为1~99，然后随机符号右边的数，与左边的数相加小于100。如果随机的符号是负号则随机符号左边的数为1~99，然后随机符号右边的数小于左边的数。如果随机的符号是乘号则符号左边的数为1~9，为保证结果小于100，符号右边随机产生数的参数为100除以符号左边的数。如果随机的符号是除号则符号右边的数为1~9，为保证结果为整数，经过数据计算处理取得符号左边的数。具体如图6-2所示。

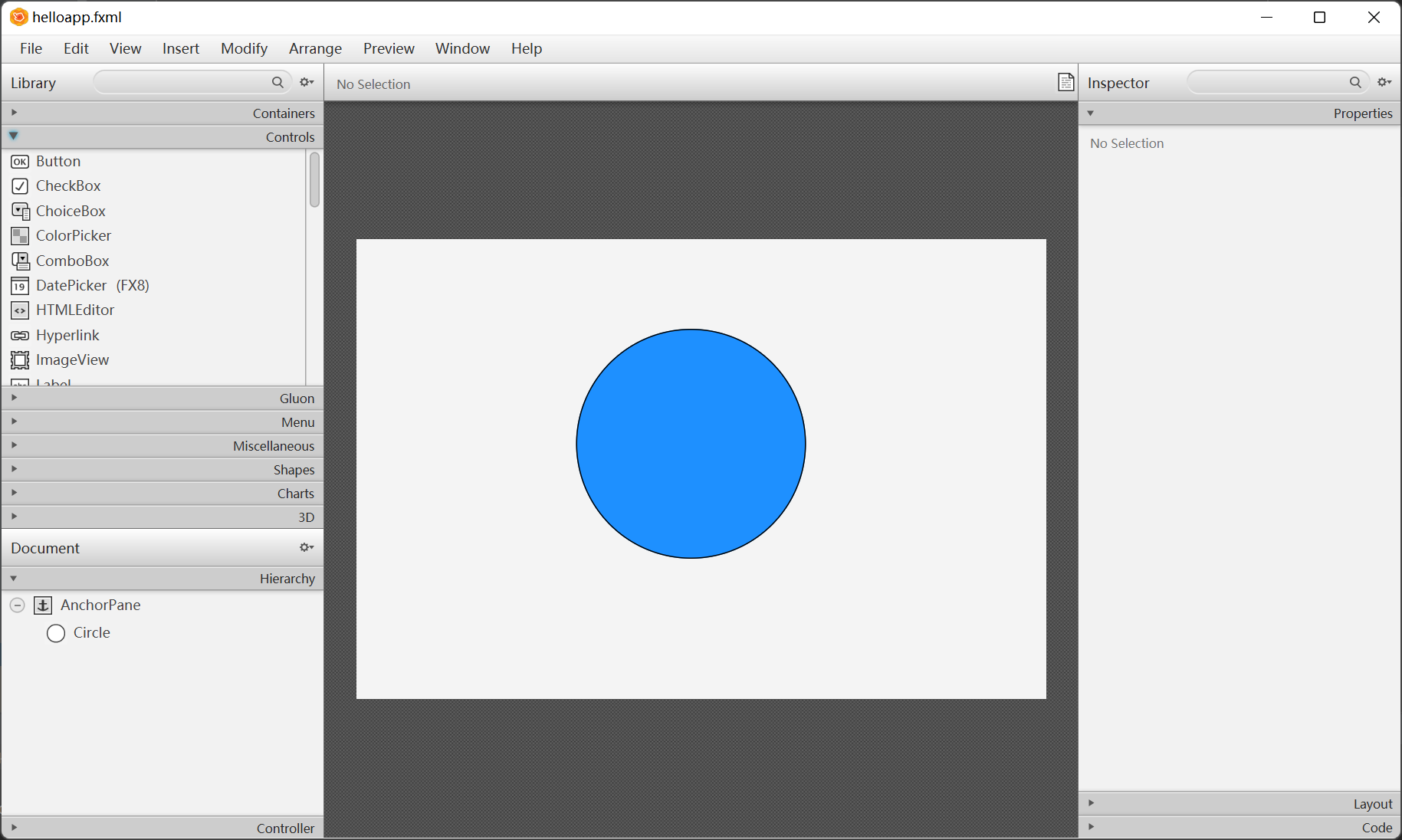


**图6-2 系统实现**

# 7 界面（F2）模块

## 7.1 程序描述

该部并无实际用途，只为使输出结果在Java图形界面方便以后模块化。该系统使用到的界面设计软件如图7-1所示。



**图7-1 Javafx软件界面设计图**

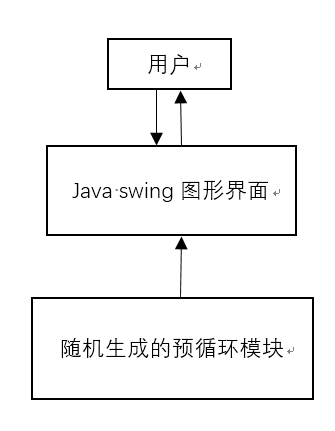
## 7.2 功能

功能的定义是对象能够满足某种需求的一种属性。凡是满足使用者需求的任何一种属性都属于功能的范畴。满足使用者现实需求的属性是功能，而满足使用者潜在需求的属性也是功能。功能定义，就是指根据收集的情报资料，透过对象产品或零件的物理特性或现象，找出其功能的本质，并用简明、准确、科学的词语进行表达。功能定义的目的是确定功能构成，为功能评价奠定基础，为构思创新方案创造条件。

该系统包含一个Java swing图形界面，可以使用户更直观的看到结果。

## 7.3 流程图

用户点击按钮后，会触发相应的事件，从而出现Java swing图形界面，便于用户直观地看到运行结果。而统随机生成的运算也会通过Java swing图形界面反馈给用户。具体如图7-1所示。



**图7-1 流程图**

## 7.4 具体实现

先设置定义窗口，在设置窗口位置、大小。在设置好的窗口之上新建文本区、设置文本区自动换行、设置字体、设置文本区不可更改，不过也无法做题、为文本区添加滚轮、将文本区添加到窗体。

应当遵循的界面设计规范，软件界面的设计，既要从外观上进行创意以达到吸引眼球的目的，还要结合图形和版面设计的相关原理，从而使得软件设计变成一门特别的艺术。软件用户界面的设计应当遵循以下几个原则：

用户导向原则：

明确到底谁是使用者，开发者需要站在用户的观点和立场上来考虑设计软件。要做到这一点，必须要和用户沟通，了解他们的需求、目标、期望和偏好等。网页的设计者要清楚用户之间差别很大，他们的能力各有不同。比如有的用户可能会在视觉方面有欠缺(如色盲)，对很多的颜色分辨不清:有的用户的听觉也会有障碍，对于软件的语音提示反映迟钝;而且相当一部分用户的计算机使用经验很初级，对于复杂一点的操作会感觉到很费力。另外，用户使用的计算机机器配置也是千差万别，包括显卡、声卡、内存、网速、操作系统以及浏览器等都会有不同。设计者如果忽视了这些差别，设计出的网页在.不同的机器上显示就会造成混乱。

KISS原则：

KISS原则就是"Keep It Simple And Stupid"的缩写，简洁和易于操作是网页设计的最重要的原则。毕竟，软件建设出来是用于普通网民来查阅信息和使用网络服务。没有必要在网页上设置过多的操作，堆集上很多复杂和花哨的图片。该原则一般要求，网页的下载不要超过10秒钟(普通的拨号用户56Kbps网速);尽量使用文本链接，而减少大幅图片和动画的使用;操作设计尽量简单，并且有明确的操作提示;软件所有的内容和服务都在显眼处向用户予以说明等。

布局控制：

关于排版布局方面，要避免排版设计的过于死板、照抄他人。如果网页的布局凌乱，仅仅把大量信息堆集在页面上，会干扰用户的视觉浏览体验。

根据视觉原理，图形与一块文字相比较，图形的视觉作用要大一些。所以，为了达到视觉平衡，在设计页面时需要以更多的文字来平衡一幅图。另外，按照中国人的阅读习惯是从左到右，从上到下，因此视觉平衡也要遵循这个道理。

和谐与一致性：

通过对软件的各种元素（颜色、字体、图形、空白等）使用一定的规格，使得设计良好的界面看起来应该是和谐的。

一致的设计结构，可以让浏览者对软件的形象有更为深刻的记忆；一致的菜单栏，可以是界面设计显得简洁大方，方便用户更加快捷的找到自己需要的功能；一致的操作设计，可以让用户快速学会在整个软件的各种功能操作。当然，软件设计的一致性并不意味着刻板和一成不变，有的软件在不同栏目使用不同的风格，或者随着时间的推移不断的改版设计，会给用户带来新鲜的感觉。

界面设计：

如果用户不更改题型设置的话，程序会随机出二十到加法题（用户若想改下题型，可以在题目类型设置里操作选择，选择加法题、减法题、或者加减法混合题，也可以选择生成的题目数量，在题目数量设置里选择，题目数量分别有20道、40道、100道三种情况）供用户作答。用户在作答完成点击提交后程序会自行判断给出作答情况，若在作答后用户对作答结果不满意的话，用户可以点击菜单栏中的请空键，这样刚刚所打答的答案会被清空，这样用户就可以再次重新做一遍，或者直接选择重新生成新的题目在做，在作答的时候用户如果需要验算，可以在点击菜单栏中的草稿纸键在草稿纸菜单界面进行相关的验算。

# 8 目标与范围

确认目标与范围是指项目正式启动时，与项目关键干系人、核心团队共同确认项目目标与范围的活动。

由于公司级项目在策划工作完成后项目经理要协同项目管理办公室（PMO）进行项目准备度检查，项目管理办公室（PMO）与项目管理委员会对项目策划工作的结果给予反馈，包括对项目的目标与范围方面的反馈意见。因此在项目启动流程中，需要参考项目准备度检查的结果对项目目标与范围进行确认，并考虑是否需要进行调整或细化。更重要的是，在关键干系人与项目核心团队中，就项目目标与范围达成一致的理解和认识。

## 8.1 测试目标

测试目标包括：功能、性能、界面、易用性、兼容性、安全性、可用性/可靠性、可维护性、可扩展性等

通过此次测试我们希望达到的目标是第一程序完好顺利的运行，功能完备，没有运行错误，第二功能可以正常的使用，检测输出的数据是否符合输出测试的要求，满足用户的需求。

## 8.2 测试范围

本次测试主要针对系统中随机产生一套50道100以内加法、减法或者加减混合运算功能的完备情况，性能是否良好。

## 8.3 性能需求

针对项目的设计要求和用户需求，我们测试该程序必须满足以下测试要求：

※ 输出结果要正常显示。

※ 手动输入可以正常输入进行运算。

※ 手动得出结果正确。

※ 结果可以正常可视化

※ 草稿纸功能随时可以正常打开

※ 试题导入导出功能正常

※ 每套习题中没有重复的算式。

※ 每个习题的答案都正确。

※ 菜单每次都可以正常使用。

## 8.4 测试输出

第三组用例如表8-1所示：

**表8-1 用例表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 测试用例序号 | 1 | 2 | 3 | 4 | 5 |
| 测试输出数据 | 44、21 | 22、14 | 3、98 | 52、29 | 14、62 |
| 预期结果 | [0,100] | [0,100] | [0,100] | [0,100] | [0,100] |
| 实际结果 | 符合 | 符合 | 符合 | 符合 | 符合 |

# 9 测试资源

1.测试资源分析

不同的测试任务对应着不同的资源需求。测试用例设计时，测试工程师需参考、应用产品待办列表、需求大纲、用户故事、Sprint计划等文档资料。

测试环境搭建时，需要获取支撑产品运行的软、硬件资源；实施测试管理时，需采购或使用开源的项目管理平台，当技术技能不足时，需参与技能培训，甚至提出招聘需求等。

测试工程师开展测试活动前，必须确保所需的测试资源到位，否则可能无法如期开展测试活动。

2.测试风险分析

软件在设计研发过程中几乎都存在风险。风险理解为某些不良事件、危险或可能危害相关事务的活动等发生的可能性以及其可能带来的不良后果，风险可能发生，也可能不发生，是一个潜在的问题。

所有的软件研发活动，都存在不同级别的风险。风险级别取决于发生不确定事件、危险的可能性及产生影响的严重度。

测试过程中可能存在的风险通常来源于3种类型：项目风险、产品风险、外因风险。

## 9.1 测试分类

比较常见的、典型的系统测试包括恢复测试、安全测试、压力测试。下面对这几种测试进行一一介绍：

1）恢复测试

恢复测试作为一种系统测试，主要关注导致软件运行失败的各种条件，并验证其恢复过程能否正确执行。在特定情况下，系统需具备容错能力。另外，系统失效必须在规定时间段内被更正，否则将会导致严重的经济损失。

2）安全测试

安全测试用来验证系统内部的保护机制，以防止非法侵入。在安全测试中，测试人员扮演试图侵入系统的角色，采用各种办法试图突破防线。因此系统安全设计的准则是要想方设法使侵入系统所需的代价更加昂贵。

3）压力测试

压力测试是指在正常资源下使用异常的访问量、频率或数据量来执行系统。在压力测试中可执行以下测试：

①如果平均中断数量是每秒一到两次，那么设计特殊的测试用例产生每秒十次中断。

②输入数据量增加一个量级，确定输入功能将如何响应。

③在虚拟操作系统下，产生需要最大内存量或其它资源的测试用例，或产生需要过量磁盘存储的数据。

## 9.2 测试工具

软件测试工具是通过一些工具能够使[软件](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6/12053)的一些简单问题直观的显示在读者的面前，这样能使测试人员更好的找出[软件错误](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6%E9%94%99%E8%AF%AF/1793242)的所在。软件测试工具分为自动化软件测试工具和[测试管理工具](https://baike.baidu.com/item/%E6%B5%8B%E8%AF%95%E7%AE%A1%E7%90%86%E5%B7%A5%E5%85%B7/3230732)。自动化软件测试工具存在的价值是为了提高测试效率，用软件来代替一些人工输入。测试管理工具是为了复用[测试用例](https://baike.baidu.com/item/%E6%B5%8B%E8%AF%95%E7%94%A8%E4%BE%8B/1928697)，提高软件测试的价值。一个好的软件测试工具和测试管理工具结合起来使用将会使软件测试效率大大的提高。

该系统的测试工具如表8-2所示：

**表8-2 测试工具**

|  |  |  |
| --- | --- | --- |
| 序号 | 用途 | 工具名称 |
| 1 | 性能测试 | Junit |
| 2 | 界面测试 | Javafx |

## 9.3 测试环境

测试环境（Testing environment）是指测试运行其上的[软件](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6/12053)和硬件环境的描述，以及任何其它与被测软件交互的软件，包括驱动和桩。测试环境是指为了完成软件测试工作所必需的计算机硬件、软件、[网络设备](https://baike.baidu.com/item/%E7%BD%91%E7%BB%9C%E8%AE%BE%E5%A4%87/7667828)、历史数据的总称。

稳定和可控的测试环境，可以使测试人员花费较少的时间就完成[测试用例](https://baike.baidu.com/item/%E6%B5%8B%E8%AF%95%E7%94%A8%E4%BE%8B/1928697)的执行，也无需为测试用例、[测试过程](https://baike.baidu.com/item/%E6%B5%8B%E8%AF%95%E8%BF%87%E7%A8%8B/10921538)的维护花费额外的时间，并且可以保证每一个被提交的缺陷都可以在任何时候被准确的重现。

测试设备如表8-3所示：

**表8-3 测试器**

|  |  |  |
| --- | --- | --- |
| 设备名称 | 硬件软件配置 | 用途说明 |
| Pc机 | IntelliJ IDEA工具 | 操作系统 |

附录 小组成员分工

张赛威：系统分析与核心代码实现

工作量占比：25%

张义龙：设定目标与范围及功能测试与完善

工作量占比：15%

张志毅：系统非功能需求及软件功能扩展

工作量占比：15%

赵浩勃：随机数生成与循环的代码实现

工作量占比：15%

赵艳威：软件界面设计与制作

工作量占比：15%

周昊：系统功能需求与文件导入及导出

工作量占比：15%