

拼题 A 系统教师用户使用手册

Programming Teaching Assistant

<https://pintia.cn>

PTA 程序设计类实验辅助教学平台
Programming Teaching Assistant

登录 注册 English

1146 位教师共同建设

10710 道高质量题目

309 所高校

拼题A

281952 位注册用户共同拼题

进入题库

最新公告 常见问题解答(FAQ) | 各种语言程序样例 查看全部

即便错过心仪的大学，也不要错过中国大学MOOC。 中国大学MOOC

IT界的“TOEFL” 计算机程序设计能力考试 PAT 计算机程序设计能力考试 Programming Ability Test

沧海竞舟 华山论剑 珠联璧合 高手云集

© 2016-2017 杭州百腾教育科技有限公司 浙ICP备17025079号-1

2017 年 12 月 08 日

目 录

零、	概述	2
一、	常见的使用脚本	2
1.	开学前准备：题目自测、建立作业集.....	2
2.	开学第一周：导入学生名单	3
3.	学期中管理：调整内容、查看实时成绩、设置无监督学生自测.....	4
4.	设置期中/期终考试	4
二、	创建新题目集	5
1.	新建题目集	5
2.	分享题目集	7
三、	创建教学班	8
四、	创建新题目	10
1.	判断题	11
2.	单选题	11
3.	多选题	12
4.	填空题	12
5.	程序填空题	13
6.	函数题	16
7.	编程题	18
8.	主观题	19
9.	LaTex 公式编辑	21
五、	批改主观题	21
六、	查看/导出成绩	22
七、	*随机组卷	23
八、	*监考设置	24
九、	*代码查重	28
1.	PTA 内部查重	28
2.	外部查重	30
附录 1:	特殊裁判程序模板	31
附录 2:	拼题 A 系统单位会员特权及收费标准.....	32
附录 3:	拼题 A 系统功能零售价目表.....	33

零、 概述

拼题 A 全称为“Programming Teaching Assistant”，网址：<https://pintia.cn>。目前用 Chrome 浏览器或火狐浏览器浏览效果最佳；IE 浏览器以及使用 IE 内核的 360 浏览器等存在由缓存引起的显示问题。

旧版 PTA 用户首次在新版系统登录时，请选择“旧版用户迁移”，输入旧版用户名、密码、以及新版的注册邮箱，通过邮件激活新版账户。当前版本采用**注册邮箱登录**。如有问题请联系管理员（chenyue@zju.edu.cn）。

系统用户分为 3 种角色：管理员、教师、学生。本手册面向“教师”。

系统支持教师需要的 4 种基本操作，顺序为：

- (1) 建立（并分享）自己的题目集
- (2) 建立自己的班级
- (3) 布置作业
- (4) 统计成绩

付费用户可使用 3 种额外保障功能：

- (1) 随机组卷
- (2) 监考
- (3) 代码查重

教师不能创建或改变一门课程的知识点框架。如果需要的课程不在系统中，或者需要修改现有课程的知识框架，则必须联系管理员（chenyue@zju.edu.cn）进行相应操作。

教师对自己创建的题目拥有版权，题目创建时的默认状态是“私有”。如果不希望他人使用，则创建者保持“私有”，该题目在题库中即对他人不可见。如果自愿开放给他人共享，则修改题目时可在“引用权限”处选择“**共享题目**”。

管理员有权限审核、修改所有题目。经过管理员审核后的题目，其“状态”将为“**已审核**”，且教师（包括作者）无权限再次修改此题。审核与否不影响题目的使用与引用，仅表示题目质量保证责任归属：**拼题 A 系统管理团队对所有“已审核”题目的质量负责，题目作者对其“未审核”题目的质量负责。**由于题库管理员精力有限，不一定能及时审核全部题目，故**审核与否与题目质量无直接联系。**

以下先给出最常用的使用脚本，随后介绍 7 大功能的使用步骤。

一、 常见的使用脚本

1. 开学前准备：题目自测、建立作业集

第一步：对于首次使用系统的教师，请先自行注册一个账号，然后将**真实姓名、注册邮箱、所属学校的全称、手机号码**发到管理员邮箱（chenyue@zju.edu.cn），索要教师权限。

第二步：假设每周要布置一批作业，每周截止后要统计学生成绩，则为每周的作业建立一个题目集（见第二节“创建新题目集”），选择为“固定时间”，起止时间设置为对应周的开始和结束时间。

第三步：选题。可以在题目集内点击加号“+”进行题目选取；也可以从左栏选择“题库”可浏览题库中所有共享题目。右上方“搜索”提供了按关键字（包括作者、标题、知识点、内容）搜索的功能。“高级搜索”则提供了按知识点搜索的功能。如果没有合适的题目，则自己出题（见第四节“创建新题目”）。**注意：系统限制单个题目集不能超过 500 题。**

第四步：（选做）建立一个自己后台测试用的题目集（见第二节）。选择“固定时间”，将起止时间定为从现在起到学期结束。将自己出的题目随时放进这个测试用题目集里，自己先测试提交。确保正确再加到学生用的题目集里。

第五步：（选做）当一个课程组内有多位教师，且大家需要布置相同的作业时，可以由某位教师创建好一个题目集，再分享（见第二节“创建新题目集”）给其他教师用户，避免大家重复劳动。

现在就准备好等待开学了。

2. 开学第一周：导入学生名单

第一步：将学生名单整理成“学号 姓名”的格式，导入教学班的用户组（见第三节“创建教学班”），同时获得**教学班验证码**。

第二步：如果想让学生每周只看到本周（及以前）的作业，不能提前看到后面几周的作业，则教师要在用户组中每周定期授权。只有在教师给学生用户组授权后，学生才能看到相应的作业。当然也可以一次授权让学生看到所有的作业，反正在作业开始前，学生只能看到作业名称，看不到题目。

第三步：上课时告诉学生以下信息：

- 刷题 A 网址：<https://pintia.cn>;
- 要求学生自己注册，并在**个人中心**（点击个人昵称可进入）通过教学班验证码**绑定学号**，才可以看到作业。

PTA 程序设计类实验辅助教学平台
Programming Teaching Assistant

chenyue

个人中心

昵称: chenyue

邮箱地址: chenyue@zju.edu.cn

密码管理: [重置密码](#)

中国大学 MOOC: [绑定](#)

应邀做题: [接受邀请](#)

绑定学号

浙江大学

姓名

学号

验证码 (通过教师获得)

[绑定](#)

注意：同一个学生可能从不同课程的老师那里得到多个验证码，但只需要绑定一次，其后的验证码可以忽略。

3. 学期中管理：调整内容、查看实时成绩、设置无监督学生自测

- **调整内容：**学期进行中，教师可随时查看学生完成作业的情况，并调整题目集的内容，包括：**添加、删除**题目（学生的成绩由系统自动调整）；**修正**题目测试数据并启动**重测**功能（**注意：**如果需要重测的工作量超过 30 分钟，建议在午夜时分，线上用户不多的时候进行）；如果多数学生都不能通过某题的某组测试，教师可以**开启**对应的“要点提示”，并提醒学生去自己的提交结果中查看。
- **收作业：**每周作业结束后，在该作业的“排名”页面上方有“**导出**”按钮，点击可在下拉框中选择导出各种文件，包括学生的答卷（每个学生提交的具体内容存在一个以学生学号命名的 html 文件里）、成绩单、正答率、得分代码（每题最早得到最高分的代码）、试卷分析。如果希望在线查看某个学生的卷面，可在排名页面点击学生的分数链接，即进入“**得分详情**”页面。**注意：**由于系统处理分数有时出现延迟造成分数不同步的现象，建议老师在判题完全结束后，点“**重建排名**”以确保成绩显示的正确性。
- **代码查重：**付费用户可以检查每个题目集中编程类题目的提交是否存在抄袭（见第八节“代码查重”）。
- **布置模拟考：**可以设计几套模拟考题，让学生自己去训练。这时可选择“固定时长”，将起止时间设置为某一周的开始和结束，设置时间长度（一般为 2 小时，或根据教师需要确定）。通知学生，在某一周内自己任意选择连续的一段时间，把模拟考试做掉。

4. 设置期中/期终考试

考试通常是同一门课程的若干个不同的教学班一起考，所以统一考试需要课程组长来创建考试、考试用户组、教师用户组。

第一步：课程组长创建考试题目集（见第二节），并加入考题。

第二步：课程组长创建课程组老师的用户组（见第三节），将老师们的账号作为“普通”用户导入，并授权该组可以旁观考试。

第三步：为每个教学班创建一个用户组，分别导入对应班级学生的学号和姓名，并为这些用户组授予在考试题目集中答题的权限。**付费用户若使用监考系统，则可能需要根据考场座位的分布拆分部分用户组（见第七节“监考设置”）。**

第四步：如果学生从未使用过本系统，则需要在考试前告诉学生：务必选择**考试登录**，选择正确的学校，输入学号、姓名，登录后可看到考试。**注意：**只有**考试开始前半小时**才允许用学号姓名登录，登录后将**无法看到任何练习题目集**。另一方面，如果学生已经使用过本系统并做过了学号绑定，则其用个人帐号登录系统并进行考试亦可，同样在**考试前半小时**开始无法看到任何练习题目集。如果教师不希望在考试期间屏蔽其他练习题目集，可在创建题目集时选择“练习”而不是“考试”（见第二节）。

第五步：考试进行中的管理与作业管理相同。结束后，由课程组长统一从“排名”页面下载成绩单、全部答卷等文件，分发给各班老师。

二、 创建新题目集

拼题 A 的题目集根据“时间限制”的不同设置分 2 种类型：

- (1) **固定时间**：学生可在指定的时间段内任意提交。教师可随时添加/删除题目。此类题目集适用于每周小作业布置。**请注意**：除了函数题和编程题之外的**其它 6 类题目**，在题目集进行中虽然可以反复提交，但看不到结果，且当前提交会**覆盖**以前的提交。**只有在题目集结束之后才能查看结果**（主观题要在题目集结束之后才由老师人工判题），**仅以最后一次提交的结果计入总成绩**。函数和编程题的成绩则随时可查，最后取第一次最高分的提交结果。**注意**：题目集的时间间隔不能超过 **365 天**。
- (2) **固定时长**：学生在指定的时间段内，选择连续的固定时长完成题目的提交。此类题目集主要适用于不集中管理的在线考试（如 MOOC 的期末考试，对分布在世界各地的学生，一般规定在指定的一周内，用连续的 120 分钟完成）。对于普通的线下考试，考虑到学生进入考场开始考试的时间不一定整齐划一，将考试时间段略放宽并设置为固定时长会比较公平（如安排在 08:00-10:00 的考试，可以设置为 07:45-10:00 这个区间内用连续 120 分钟完成，这样可以给学生 15 分钟陆续入场的时间，同时能保证每个人的考试用时均相同）。**注意**：题目集的时长不能超过 **24 小时**。

题目集可用于**练习**和**考试**，区别仅在于是否具有**屏蔽功能**——类型为“考试”的题目集，开考前半小时至该考生考试结束期间，其将无法看到任何类型为“练习”的题目集。

1. 新建题目集

以下是创建新题目集的操作流程：

第一步： 点击“**题目集**”列表页面上方的“**新建题目集**”；



The screenshot shows the PTA interface with a sidebar on the left containing '题目集', '题库', and '用户组'. The main content area displays a table of existing question sets. A red box highlights the '新建题目集' button at the top of the table.

标题	类型	状态	操作
浙大版《C语言程序设计（第3版）》题目集	教材	一直可用	修改
浙大版《C语言程序设计实验与习题指导（第3版）》题目集	教材	一直可用	修改
基础编程题目集	练习	一直可用	修改
数据结构与算法题目集（中文）	练习	一直可用	修改
Data Structures and Algorithms (English)	练习	一直可用	修改
中国大学MOOC-陈越、何钦铭-数据结构-起步能力自测题	练习	一直可用	修改
浙大版《数据结构（第2版）》题目集	教材	一直可用	修改
Final programming test	练习	进行中	修改

第二步：填写题目集名称（不超过 80 个字符，例如“浙江大学 2015 年数值分析练习集”）；选择各种类型；设置时间限制；如果限制必须用某种编程语言，则选择相应的语言。最后点击“**创建**”。注意：题目集一旦创建，其各种类型设置即**不可更改**。

基本信息

名称 *

类型 * 练习 考试

练习期间不会隐藏其他题目集；练习不可提前交卷。

随机组卷 * (VIP 用户或付费可用) 开启 关闭

时间类型 * 固定时长

~ 分钟

固定时间

多选题判分 * 规则一 - 做对得全分，做错不得分 规则二 - 做对得全分，少选得一半分，多选不得分 规则三 - 每个选项独立判分

编译器限制 * 全选

<input checked="" type="checkbox"/> C (gcc)	<input checked="" type="checkbox"/> C++ (g++)	<input checked="" type="checkbox"/> C (clang)	<input checked="" type="checkbox"/> C++ (clang++)
<input checked="" type="checkbox"/> Java (openjdk)	<input checked="" type="checkbox"/> Python (python 2)	<input checked="" type="checkbox"/> Python (python 3)	<input checked="" type="checkbox"/> Ruby (ruby)
<input checked="" type="checkbox"/> Bash (bash)	<input checked="" type="checkbox"/> Plaintext (cat)	<input checked="" type="checkbox"/> Common Lisp (clisp)	<input checked="" type="checkbox"/> Pascal (fpc)
<input checked="" type="checkbox"/> Go (go)	<input checked="" type="checkbox"/> Haskell (ghc)	<input checked="" type="checkbox"/> Lua (lua)	<input checked="" type="checkbox"/> Lua (luajit)
<input checked="" type="checkbox"/> C# (mcs)	<input checked="" type="checkbox"/> JavaScript (node)	<input checked="" type="checkbox"/> OCaml (ocamlc)	<input checked="" type="checkbox"/> PHP (php)
<input checked="" type="checkbox"/> Perl (perl)	<input checked="" type="checkbox"/> AWK (awk)	<input checked="" type="checkbox"/> D (dmd)	<input checked="" type="checkbox"/> Racket (racket)
<input checked="" type="checkbox"/> Vala (valac)	<input checked="" type="checkbox"/> Visual Basic (vbnc)	<input checked="" type="checkbox"/> Kotlin (kotlinc)	<input checked="" type="checkbox"/> Swift (swiftc)
<input checked="" type="checkbox"/> Objective-C (gcc)			

公告

第三步：点击“**创建**”后将出现“**题目列表**”，点击对应题目类型下方的加号“+”，在弹出窗口中勾选需要添加的题目，最后点击“**添加到题目集**”即添加到题目集相应类别下。教师可根据自己的需要拖拽改变题目顺序、自定义题目编号、修改题目的分数分布、打开或关闭要点提示，最后“**修改**”。如果需要自己创建新题目，请跳到第四节“创建新题目”；

请注意：部分函数和编程题的作者可能为每组测试数据提供了“**要点提示**”，即该组数据主要测试哪些方面。例如“测试点 0：与样例等价；测试点 1：边界测试-最小规模；测试点 2：边界测试-最大规模”等。在教师添加题目时，**默认所有的提示都是关闭的**。教师可以选择开启某一题的某一组测试数据的提示，则随后学生们在查看自己提交详情时，可在相应的数据测试结果后看到该提示。

2. 分享题目集

先建好题目集，然后从“**题目集**”列表中进入该题目集对应的“**修改**”页面。在“**更新**”键左下方有“**生成分享码**”和“**使用分享码**”两个按钮。



点击“**生成分享码**”后将生成一个分享码。生成时有一则友情提示：“其他用户可通过分享码导入本题目集内所有题目。（他人创建的私有题目除外）”。意即使用分享码的教师将得到此集内“共享”的题目、以及生成分享码的教师自己出的属性为“私有”的题目。但**私有题目不得转发**，即生成分享码的教师不可将自己获得的他人创建的私有题目转手分享给其他人。

将分享码发送给其他需要的教师即可。

注意：若删除当前的分享码，再次生成，则此前的分享码失效。

若要采用其他教师分享的题目集，则必须先自己建立一个题目集，然后从“**题目集**”列表中进入该题目集对应的“**修改**”页面。点击“**使用分享码**”并输入正确的分享码后将出现以下弹窗：

标号	标题	知识点	难度	分数
判断题 4分				
<input checked="" type="checkbox"/>	1-1	To merge 55 runs using 3 tapes for a 2-way merge, the original d	外部排序	2
<input checked="" type="checkbox"/>	1-2	If only one tape drive is available to perform the external sort	外部排序	2
单选题 6分				
<input checked="" type="checkbox"/>	2-1	Given 100,000,000 records of 256 bytes each, and the size of the	外部排序	3
<input checked="" type="checkbox"/>	2-2	In external sorting, suppose we have 5 runs of lengths 2, 8, 9,	外部排序	3

勾选需要的题目并点击“**导入**”即可。

注意：第一，系统是在分享码被使用时，才去原始题目集复制题目的。若原始题目集在分享码生成后

发生了改变，则改变也会在分享码被使用时被复制。第二，接受分享的题目集不一定为空集，系统会自动检查、并不导入重复的题目。

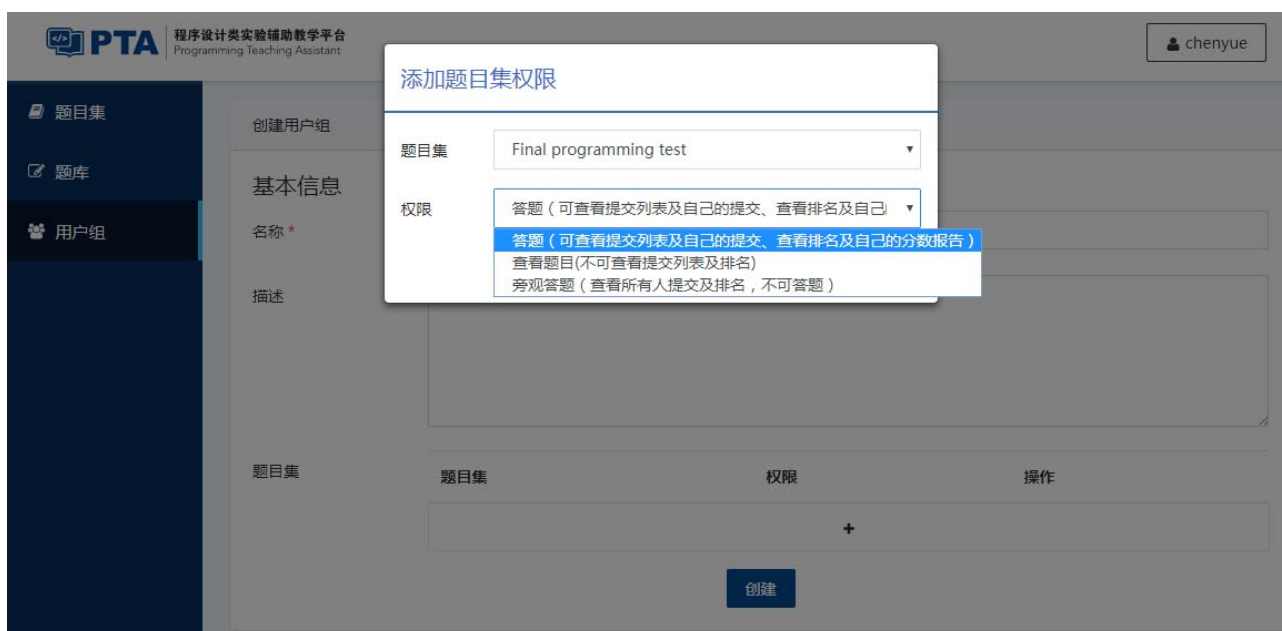
三、 创建教学班

先建好题目集，然后导入需要使用该题目集的学生名单。步骤如下：

第一步： 点击“**用户组**”，然后“**创建用户组**”；

第二步： 填写用户组名称及描述（例如取名为“ZD-DS2017”，描述为“浙大 2017 年数据结构学生”）；

第三步： 点击“题目集”下方的加号“+”，在弹出窗口的下拉框中选择需要的题目集，并选择相应的权限，点击“**添加**”。可以为同一批学生同时选择若干个不同的题目集（如果有很多次作业），并赋予不同的权限。完成后点击“**创建**”；



第四步： 在“添加用户”窗口添加普通用户，在“添加学生用户”窗口添加学生用户。根据提示输入同事或学生的信息，点击“**添加**”。导入的用户名单出现在下方。当加入学生用户后，系统自动生成**验证码**。教师应将此验证码发给学生，可将学号绑定到个人账户。

添加用户

每个用户一行，每行一个已注册用户的邮箱地址。

hqm@zju.edu.cn
xjc@zju.edu.cn

添加

添加学生用户

每个学生一行，每行包含学号和真实姓名，用空格隔开。

学生可通过验证码 963126 将学号绑定到个人账户。

1234567891 路人乙
1234567892 路人丙

添加

用户信息

用户	邮箱	操作
1234567890 路人甲	-	删除

注意到“添加用户”上方还有一个“邀请码”的创建区，如下图。

邀请码

邀请码	创建时间	操作
+		

添加用户

每个用户一行，每行一个已注册用户的邮箱地址。

此功能主要为 MOOC 教师和企业人力资源部门设计，教师一般不需要。

企业招聘时，可以将生成的邀请码群发给应聘者；应聘者在拼题 A 系统登录后，在个人中心填入邀请码，接受邀请，就被系统自动加入用户组，拥有了该用户组的答题权限。

MOOC 教师也是类似，将邀请码公布给学员，学员接受邀请后即可进入 MOOC 用户组。**注意：**中国大学 MOOC 学员可以在“个人中心”将拼题 A 系统账号与中国大学 MOOC 账号进行“绑定”，教师从拼题 A 导出的成绩单中将带有一个特殊的 MOOC-ID，方便将成绩导入爱课程平台。

四、 创建新题目

注意：

1. 所有上传的测试数据都必须是 Unix 格式（UTF-8）的英文文本，且压缩为 ZIP 格式。分数文件中可以有中文提示，但也必须转为 UTF-8 格式，否则不能正常显示。
2. MAC 压缩包中可能含有多余文件，会造成解压错误，故请尽量避免使用。
3. 测试数据集支持单文件更新，同名的文件可以覆盖之前上传的文件，新文件会作为新的测试用例补充到测试数据集。

第一步： 点击“题库”；点击 7 种题目类型之一，后点击“新建题目”，开始创建新题目。

PTA 程序设计类实验辅助教学平台
Programming Teaching Assistant

chenyue

判断题 单选题 多选题 填空题 程序填空题 函数题 编程题

新建题目 批量加入题目集 可用题目: 2272 题 搜索

标题	知识点	作者	难度	分数	状态	更新时间	操作
生成树问题：给定无向图 $G = (V, E)$...	局部搜索	叶德仕	4	4	已审核	2017/7/3	
\$\$\$K\$\$\$-中心问题：给定 $$$$N$$$$ 个城...	近似算法	叶德仕	4	4	已审核	2017/7/3	
考虑在有 $$$$n$$$$ 个顶点、 $$$$m$$$$ 条边的...	随机算法	叶德仕	4	4	已审核	2017/7/3	
用分治法解决一个输入规模为 $$$$N$$$$ 的...	分治法	杨欣豫	3	3	已审核	2017/7/3	
用分治法解决一个输入规模为 $$$$N$$$$ 的...	分治法	杨欣豫	3	3	已审核	2017/7/3	

注意：每种题目都需要选择难度、作者及其单位、知识点（包括课程）。涉及编程的“程序填空题”、“函数题”、“编程题”还要求给题目设定简短的标题。

基本信息

难度 *

1

作者 *

单位 *

zju - 浙江大学

知识点 *

添加知识点

1. 判断题

题目编辑框上方是效果预览，可在输入的同时查看题面展现效果。

创建新题目时，编辑框内出现默认文字，其中“~@[\$(#)”是系统识别标志，“\$”为“T”或者“F”，对应判断题的答案；“#”是一个正整数，对应本题的分数。将相应内容替换为自己需要输入的内容，点击底部的“**创建**”即可。

题目预览 - 题面描述及答案（答题者只可查看题面描述）

这是一个判断题的样例。答案为T，分值为5分。（5分）

T F

编辑 B I H | `</>` x² “ ” ☰ ☷ 🔗 🖼️ 📄 🗑️

这是一个判断题的样例。答案为T，分值为5分。 ~@[T](5)

2. 单选题

与判断题类似。创建新题目时，编辑框内出现默认文字，其中“@[\$(#)”是系统识别标志，“\$”为大写英文字母，对应题目答案；“#”是一个正整数，对应本题的分数。将相应内容替换为自己需要输入的内容，点击底部的“**创建**”即可。

题目预览 - 题面描述及答案（答题者只可查看题面描述）

这是一个单选题的样例。答案为C，分值为2分。（2分）

A. 替换为错误项
 B. 替换为错误项
 C. 替换为正确项
 D. 替换为错误项

编辑 B I H | `</>` x² “ ” ☰ ☷ 🔗 🖼️ 📄 🗑️

这是一个单选题的样例。答案为C，分值为2分。 @[C](2)

A. 替换为错误项
B. 替换为错误项
C. 替换为正确项
D. 替换为错误项

3. 多选题

与单选题类似。创建新题目时，编辑框内出现默认文字，其中“*@[\$.#]”是系统识别标志，“\$”为大写英文字母，对应题目答案；“#”是一个正整数，对应本题的分数。将相应内容替换为自己需要输入的内容，点击底部的“创建”即可。

题目预览 - 题面描述及答案 (答题者只可查看题面描述)

这是一个多选题的样例。答案为AC，分值为2分。(2分)

- A. 替换为正确项
- B. 替换为错误项
- C. 替换为正确项
- D. 替换为错误项

编辑 B I H | `</>` x² “ ” ☰ ☷ | 🔗 🖼️ 📄 | 🗑️ ✕

这是一个多选题的样例。答案为AC，分值为2分。 *@[AC]2)

- A. 替换为正确项
- B. 替换为错误项
- C. 替换为正确项
- D. 替换为错误项

4. 填空题

与选择题类似。创建新题目时，编辑框内出现默认文字，其中“@@[\$|\$.#]”是系统识别标志，“\$”为字符串，对应题目答案——每个空最多可以有10个备选答案，匹配上任何1个就得分；“#”是一个正整数，对应本题的分数。将相应内容替换为自己需要输入的内容，点击底部的“创建”即可。

题目预览 - 题面描述及答案 (答题者只可查看题面描述)

这是填空题模板。 (3分) 最多可设十个可选答案，用 "|" 隔开。

每题可以有多个空。 (2分)。题目创建后不可修改填空数量。

编辑 B I H | `</>` x² “ ” ☰ ☷ | 🔗 🖼️ 📄 | 🗑️ ✕

这是填空题模板。 @@[答案一|答案二|答案三](3) 最多可设十个可选答案，用 "|" 隔开。

每题可以有多个空。 @@[答案一|答案二|答案三](2)。题目创建后不可修改填空数量。

注意 1: 填空题创建后，创建者仍然可以修改其中的内容，但是**填空的个数不能更改**。

注意 2: 系统会自动将学生输入的字符串进行预处理，删除开头和结尾的白字符后才与标准答案进行匹配。标准答案开头和结尾的白字符也会被自动删除。但**字符串中间的白字符不会被处理**。

5. 程序填空题

程序填空题要求给出一个指定接口的函数的功能描述，函数代码部分挖空，由学生填写。教师须编写**测试程序**，调用包含学生代码的函数，运行**一组**测试用例，由系统判定得分。

系统判题的原理是：如果有 N 个空，则系统会将教师的测试程序编译运行 N 次，每次将 1 个空替换成学生的提交，其他 $N - 1$ 个空替换成标准答案。运行时将结果与教师提供的一组测试用例比对，正确则得到该空对应的分数，否则得 0 分。

第一步： 给出题干描述。创建新题目时，编辑框内出现默认文字，其中代码部分夹在一对“`”之间（“`c++”表示是 C++ 代码）。“@[$\$$](#)”是系统识别标志，“ $\$$ ”为该空的标准答案代码；“#”是一个正整数，对应本空的分数。

第二步： 在“题目判定”部分填入“测试程序”的代码（注意选择正确的编译器）。所谓“测试程序”，即调用题干中的函数，依次用学生填空替换标准答案，运行某输入并且产生可以判定的输出的完整程序。具体请见给出的“实例”。

第三步： 如果输出结果唯一，则可直接采用默认的“服务器判题程序”（即“使用已有检查程序”）。否则需要“自定义”，编写特殊裁判程序（请见“附录 1：特殊裁判程序模板”）。

第四步： 如果有必要，则改写程序运行的时间限制和内存限制；否则按默认值执行。

第五步： 上传 **Unix 格式 (UTF-8) 文本、且压缩为 ZIP 格式**的测试数据（注意：只接受 1 个输入文件 0.in 和 1 个输出文件 0.out）；

第六步： 点击“**创建**”生成新的程序填空题。

注意: 填空题创建后，创建者仍然可以修改其中的内容，但是**填空的个数不能更改**。

题目预览 - 题面描述及答案 (答题者只可查看题面描述)

这是程序填空题模板。题目要求写在这里。例如：本题目要求打印“Hello World!”。例如：

```
#include <stdio.h>

int main()
{
    printf("Hello World!  (2分)");
     (2分);
}
```

编辑 **B** *I* **H** | `</>` x^2 | |

这是程序填空题模板。题目要求写在这里。例如：本题目要求打印“Hello World!”。例如：

```
```C++
#include <stdio.h>

int main()
{
 printf("@@[Hello World!](2)");
 @@[return 0](2);
}
```
```

实例：Bubble Sort

题干描述（本题有 4 个空，每空 3 分，用红色标出）

Bubble sort is a simple sorting algorithm. Suppose we have a list of integers and want to sort them in ascending order. Bubble sort repeatedly scans the list from the head to the tail, and swaps two adjacent numbers if they are in the wrong order. Please complete the following program to implement bubble sort.

```
```c++
typedef struct node *nodeptr;
struct node{
 int value;
 nodeptr next;
 /* some other fields */
};

nodeptr BubbleSort (nodeptr h)
{/* h is the head pointer of the list with a dummy head node */
```

```

nodeptr p, q;
int flag_swap;

if (!h->next) return h;
do{
 flag_swap = 0;
 p = h;
 while (p->next->next){
 if (@@[p->next->value > p->next->next->value](3)){
 flag_swap++;
 q = p->next;
 @@[p->next = q->next](3);
 @@[q->next = p->next->next](3);
 @@[p->next->next = q](3);
 }
 else p = p->next;
 }
} while (flag_swap > 0);
return h;
}
...

```

测试程序（4个空的内容用@@替换）

```

#include <stdio.h>
#include <stdlib.h>

/*----- 题面在此 -----*/
typedef struct node *nodeptr;
struct node{
 int value;
 nodeptr next;
 /* some other fields */
};

nodeptr BubbleSort (nodeptr h)
{ /* h is the head pointer of the list with a dummy head node */
 nodeptr p, q;
 int flag_swap;

 if (!h->next) return h;
 do{
 flag_swap = 0;
 p = h;
 while (p->next->next){
 if (@@){
 flag_swap++;
 q = p->next;
 @@;
 @@;
 @@;
 }
 else p = p->next;
 }
 } while (flag_swap > 0);
 return h;
}
/*-----*/

```

```

int main ()
{
 nodeptr h, t;
 int v;

 h = malloc(sizeof(struct node));
 h->next = NULL;
 scanf("%d", &v);
 while (v>0) {
 t = malloc(sizeof(struct node));
 t->value = v;
 t->next = h->next;
 h->next = t;
 scanf("%d", &v);
 }
 h = BubbleSort(h);
 for (t=h->next; t; t = t->next)
 printf("%d ", t->value);

 return 0;
}

```

输入文件: 0.in

```
11 6 4 8 1 15 9 7 10 2 3 5 -1
```

输出文件: 0.out

```
1 2 3 4 5 6 7 8 9 10 11 15
```

## 6. 函数题

函数题要求学生编写一个指定接口的函数，完成指定功能。教师须编写测试程序，调用学生提交的函数，运行一系列测试用例，由系统判定得分。

第一步： 给出题干描述。创建新题目时，编辑框内出现默认文字，其基本格式与程序填空题类似。描述包括 4 大部分：

- (1) 要求完成的函数的基本功能描述；
- (2) 函数接口定义；
- (3) 裁判测试程序样例 —— 大多数情况下，这里的代码跟实际裁判的测试程序是一样的，但也允许不一样的情况；
- (4) 输入和输出样例（注意：新版格式增加了“in”和“out”标记，详见实例）。

第二步： 在“题目判定”部分填入“测试程序”的代码。可以与题目描述中的“裁判测试程序样例”相同。注意学生提交的代码部分用“@@@”替代，具体请见给出的“实例”。

第三步： 填入“标准答案”，即要求学生提交的内容的标准答案代码。

第四步： 如果输出结果唯一，则可直接采用默认的“服务器判题程序”（即“使用已有检查程序”）。否则需要“自定义”，编写特殊裁判程序（请见“附录 1：特殊裁判程序模板”）。

第五步： 如果有必要，则改写程序运行的时间限制和内存限制；否则按默认值执行。

第六步： 上传 **Unix 格式(UTF-8)**文本、且压缩为 **ZIP 格式**的测试数据(编号为 0.in, 0.out, 1.in, 1.out, ...) 和分数分布文件(命名为 scores.txt, 每行格式为[case\_name] [score] #[要点提示]#, 空格分开。

其中[case\_name]如果省略，则默认使用 0, 1, 2 依次递增；[要点提示]不超过 40 个中文字符，简要描述测试要点，如果省略，则为无提示)。注意：输入输出文件总量不可超过 15MB。

第七步： 点击“**创建**”生成新的函数题。

### 实例：统计个位数字

#### 题干描述

本题要求实现一个函数，可统计任一整数中某个位数出现的次数。例如-21252 中，2 出现了 3 次，则该函数应该返回 3。

### 函数接口定义：

```
```c++
int Count_Digit ( const int N, const int D );
```
```

其中`N`和`D`都是用户传入的参数。`N`的值不超过`int`的范围；`D`是[0, 9]区间内的个位数。函数须返回`N`中`D`出现的次数。

### 裁判测试程序样例：

```
```c++
#include <stdio.h>

int Count_Digit ( const int N, const int D );

int main()
{
    int N, D;

    scanf("%d %d", &N, &D);
    printf("%d\n", Count_Digit(N, D));
    return 0;
}
```

/* 你的代码将被嵌在这里 */

```
```
```

### 输入样例：

```
```in
-21252 2
```
```

### 输出样例：

```
```out
3
```
```

测试程序：运行时“@@@”将被替换成学生提交的代码。

```
#include <stdio.h>
int Count_Digit (const int N, const int D);

@@@

int main()
{
 int N, D;
```

```
scanf("%d %d", &N, &D);
printf("%d\n", Count_Digit(N, D));
return 0;
}
```

标准答案

```
int Count_Digit (const int N, const int D)
{
 int cnt, n;

 if (N==D) return 1;
 cnt = 0;
 n = N<0? -N:N;
 while (n) {
 if ((n%10) == D) cnt++;
 n /= 10;
 }
 return cnt;
}
```

测试数据与分数:

| *.in         | *.out | scores.txt               |
|--------------|-------|--------------------------|
| -313532363 3 | 5     | 9 # sample, 数字<0 且不连续出现# |
| 1554555500 5 | 6     | 3 #数字>0, 有连续出现#          |
| 0 0          | 1     | 1 # N=0, 且输出 1#          |
| -22334466 9  | 0     | 1 #输出为 0#                |
| 100000000 0  | 8     | 1 #统计 0 的次数, 可能循环会多 1#   |

## 7. 编程题

编程题要求学生编写一个完整的程序，完成指定功能。系统自动调用运行一系列测试用例，并判定得分。

第一步： 给出题干描述。创建新题目时，编辑框内出现默认文字，其基本格式与函数题类似。描述包括 3 大部分：

- (1) 要求完成的任务描述；
- (2) 输入和输出格式；
- (3) 输入和输出样例（注意：新版格式增加了“in”和“out”标记，详见实例）。

第二步： 在“题目判定”部分填入“标准答案”，即要求学生提交的内容的标准答案代码。

第三步： 如果输出结果唯一，则可直接采用默认的“服务器判题程序”（即“使用已有检查程序”。否则需要“自定义”，编写特殊裁判程序（请见“附录 1：特殊裁判程序模板”）。

第四步： 如果有必要，则改写程序运行的时间限制、内存限制、语言限制以及代码长度限制；否则按默认值执行。

第五步： 上传 **Unix 格式(UTF-8)**文本、且压缩为 **ZIP 格式**的测试数据(编号为0.in, 0.out, 1.in, 1.out, ...)和分数分布文件(命名为 scores.txt, 每行格式为[case\_name] [score] #[要点提示]#, 空格分开。其中[case\_name]如果省略，则默认使用 0, 1, 2 依次递增；[要点提示]不超过 40 个中文字符，

简要描述测试要点，如果省略，则为无提示)。注意：输入输出文件总量不可超过 15MB。

第六步： 点击“创建”生成新的编程题。

### 实例：A+B

题干描述

本题目要求读入 2 个整数 A 和 B，然后输出它们的和。

### 输入格式：

输入在一行中给出 2 个绝对值不超过 1000 的整数 A 和 B。

### 输出格式：

对每一组输入，在一行中输出 A+B 的值。

### 输入样例：

```
```\nin  
18 -299  
```
```

### 输出样例：

```
```\nout  
-281  
```
```

标准答案：

```
#include <stdio.h>

int main()
{
 int A, B;
 scanf("%d %d", &A, &B);
 printf("%d\n", A+B);
 return 0;
}
```

测试数据与分数：

| *.in       | *.out | scores.txt    |
|------------|-------|---------------|
| 18 -299    | -281  | 6 #等于样例，和为负数# |
| -1000 1000 | 0     | 2 #和为0#       |
| -2 333     | 331   | 2 #和为正数#      |

## 8. 主观题

主观题是必须由教师人工判定并给分的题目类型，系统会自动计算总得分。与其他题型类似的是，创建新题目时，编辑框内也出现简单的默认文字，其中“@[#)”是系统识别标志，“#”是一个正整数，对应本题的分数。

此外，如果需要学生上传实验报告等文件，则须将上传选项设置为“开启”（默认“关闭”）。最后老师可以填写评分标准，作为给自己的提示。完成后点击底部的“创建”即可。



题目预览 - 题面描述及答案 ( 答题者只可查看题面描述 )

这是一个主观题模板。请在这里写题目描述。(5分)

编辑 B I H  $\langle / \rangle$   $x^2$  “ ” ☰ ☷ 🔗 🖼️ 📄 🗑️

这是一个主观题模板。请在这里写题目描述。@[](5)

### 实例：文献综述

#### 题干描述

针对给定范围内某一领域展开搜索，找到至少 1 篇近年综述和 10 篇近年论文，撰写至少 2000 字的文献综述（中文）。

要求独立完成（被发现任何抄袭行为者，直接 0 分）。@[ ](30)

可选择领域：

- \* 机器学习
- \* 数据挖掘
- \* 分布式与并行计算
- \* 大数据存储
- \* 信息检索
- \* 图像处理
- \* 语音识别
- \* 智能传感器网络
- \* 网络安全
- \* 系统安全
- \* 优化算法
- \* 程序分析
- \* 计算可视化
- \* 医学信息化
- \* 语义网
- \* 社交网络分析
- \* 云计算
- \* 大型软件可信性
- \* 嵌入式系统
- \* 电子商务

评分标准:

评分点 1: 阐述领域研究意义、背景、发展脉络 (3 分)。按下列标准扣分:

- 意义阐述不明确: -1
- 研究背景及发展脉络未理清: -1
- 其它 (将在最终评语中说明)

评分点 2: 综述参考文献的研究内容 (7 分)、存在问题及可能的原因 (3 分)。按下列标准扣分:

- 对于参考文献的研究内容综述过于简单: -0.2 到 0.5/篇
- 未充分分析某文献的弱点: -0.2/篇
- 对文献的分析缺乏整体逻辑性: -1
- 参考文献未被全部引用: -1
- 其它 (将在最终评语中说明)

评分点 3: 阐述进一步研究的课题 (4 分), 展望领域发展方向 (3 分)。按下列标准扣分:

- 对进一步研究的课题结论不合理: -1
- 对进一步研究的课题只有结论没有推导: 最多-3
- 对发展方向的判断结论不合理: -1
- 对发展方向只有结论没有推导: 最多-2
- 其它 (将在最终评语中说明)

评分点 4: 阐述自己的见解和感想 (6 分)。按下列标准扣分:

- 个性思想不鲜明, 只是复述他人观点: -2
- 篇幅过短, 甚至不到 300 字: -2
- 语焉不详, 难以理解: -1
- 其它 (将在最终评语中说明)

评分点 5: 列出参考文献 (4 分)。按下列标准扣分:

- 未包括综述性文章: -1
- 前沿论文不到 10 篇: -1
- 论文比较陈旧过时: -1
- 论文与主题相关度不强: -1 或-2
- 其它 (将在最终评语中说明)

## 9. LaTeX 公式编辑

数学公式放在一对\$之间, 采用 LaTeX 公式编辑语法。如:  $\$A+B-C\times D/E, \quad O(N^{2i}), \quad \sum_{i=0}^{\infty} x_i^{k+1}, \quad \sqrt{\frac{a+s}{b-s}} \$$  的显示结果如下图所示:

$$A + B - C \times D/E, \quad O(N^{2i}), \quad \sum_{i=0}^{\infty} x_i^{k+1}, \quad \sqrt{\frac{a+s}{b-s}}$$

更多 LaTeX 语法请搜索百度文库“LATEX 数学符号的输入”。

## 五、 批改主观题

当一个题目集中包括有主观题时, 教师可在左侧栏中看到“**批改主观题**”的选项, 点击进入。



对于未批改的题目，其状态为“成绩待发布”。进入“查看”页面后，题面下方列出了全部当前的提交。教师可以随时进行“批改”（当然在考试结束前，学生随时可能修改答案，所以教师一般不会对考试进行中就批改学生的提交）。

教师完成批改后，对应学生的作业状态变为“待发布”，此时学生尚不能看到教师批改的结果。

点击“发布成绩”后，学生登录时就可以看到教师评语和自己的成绩了。

如果教师在发布成绩后又进行了修改，状态会回到“待发布”。每一次成绩的变更，只有在“发布成绩”后才能被学生看到。

## 六、 查看/导出成绩

第一步： 进入题目集的“排名”页面；

第二步： 点击“导出”，可导出学生的答卷、成绩单、正答率、得分代码（即第一次得到当前最高分的代码）、试卷分析；

第三步： 如果希望查看某个学生的卷面，可在排名页面点击学生的总分链接，即进入“得分详情”页面。

注意：

1. 由于系统在高频使用中会产生成绩延迟同步的现象，所以为保证最后成绩显示正确，教师可在考试/练习结束后点“重建排名”，之后再导出成绩。
2. 如果在用户组中取消了该组在某题目集的答题权限（即相当于从题目集中删除了用户组），则无法正常导出该组用户的任何答题数据。

| 全部  | 导出       | 题目集查重 | 重建排名 | 判断  | 单选  |
|-----|----------|-------|------|-----|-----|
| 排名  | 用户       | 答卷    |      | 判断题 | 单选题 |
| 439 | 我 dsmooc | 成绩单   |      | -   | -   |
| 1   | Lucifer  | 正答率   |      | 10  | 64  |
| 1   | lyw321   | 得分代码  | 100  | 10  | 64  |
| 1   | ...      | 试卷分析  | 100  | 10  | 64  |

## 七、 \*随机组卷

此功能仅限付费用户使用。

随机组卷功能提供了3重随机乱序的选择，即：题目池中随机抽题、卷面显示的题目顺序随机打乱、卷面显示的选择题选项随机打乱。当选择的题目数量充分多时，重卷的概率就非常小。教师可以在创建题目集时选择开启随机组卷（即随机抽题功能）及其他随机选项。

### 基本信息

名称 \*

类型 \*  练习  考试

随机组卷 \*  开启  关闭

题目顺序随机

答案顺序随机

时间类型 \*  固定时长

在随机组卷状态下，每点击一次加号“+”就开启了一个“随机题目池”，从弹窗中选中的所有题目会被加入同一个题目池。教师可点击题目池上方的“添加 修改”向池中添加题目、或修改该池的抽取题目数和分数的设置。

| 随机组卷题目列表 |                                                                  |      |    | 共 2 分 |       |
|----------|------------------------------------------------------------------|------|----|-------|-------|
| 标号       | 标题                                                               | 知识点  | 难度 | 分数    | 操作    |
| 随机题目池    | 3 选 1                                                            |      |    |       | 添加 修改 |
|          | 在团队任务中未完成本人分工的任务，却在报告中给出虚假分工和承诺，此种行为属于作弊。                        | 团队作业 | 1  | 1     | 修改 删除 |
|          | 在团队任务中未能阻止、或举报组员抄袭行为，属于协助作弊，应与直接作弊者一同处罚。                         | 团队作业 | 1  | 1     | 修改 删除 |
|          | 学生A在作业提交截止前，从网上找了一份答案提交，后又提交了自己独立完成的、并未满分的答案，并主动向老师坦白自己的答案并未满分。A | 个人作业 | 1  | 1     | 修改 删除 |
| 随机题目池    | 3 选 1                                                            |      |    |       | 添加 修改 |
|          | 学生A在做作业时，从网上找了一份答案学习，然后修改了自己的答案提交。A应被判作弊。                        | 个人作业 | 1  | 1     | 修改 删除 |
|          | 学生A在作业提交截止、自己的得分确定后，请老师重新开放作业集，从网上找到答案学习，并且提交了其他人的答案和自己修改后的答案。A应 | 个人作业 | 1  | 1     | 修改 删除 |
|          | 学生A在网上的开放题库中自主练习，并且在博客中发布了解题报告和代码。后该题目被老师作为作业发布，该生博客中的代码被学生B抄袭。B | 个人作业 | 1  | 1     | 修改 删除 |
| +        |                                                                  |      |    |       |       |

**注意：**考试开始后对题目池中的题目进行增加，实际上无法生效，因为每个学生的卷子在其点开考试的瞬间就生成并保存了。考试开始后只要有考生开始答题，就无法再删除题目。

## 八、 \*监考设置

此功能仅限付费用户使用。

第一步： 在监考系统中设置考试

1. 在创建题目集后，编辑状态下可以点击“**管理监考系统**”，即进入监考系统的“**考试信息**”设置界面。

### 基本信息

|         |                                                     |                                                       |
|---------|-----------------------------------------------------|-------------------------------------------------------|
| 名称 *    | <input type="text" value="Final programming test"/> |                                                       |
| 类型 *    | <input checked="" type="radio"/> 练习                 | <input type="radio"/> 考试                              |
| 随机组卷 *  | <input type="radio"/> 开启                            | <input checked="" type="radio"/> 关闭                   |
| 时间类型 *  | <input type="radio"/> 固定时长                          |                                                       |
|         | <input checked="" type="radio"/> 固定时间               |                                                       |
|         | <input type="text" value="2017-07-06 00:00"/>       | ~ <input type="text" value="2017-09-01 00:00"/>       |
| 监考系统    | <input type="button" value="管理监考系统"/>               |                                                       |
| 编译器限制 * | <input type="checkbox"/> 全选                         |                                                       |
|         | <input checked="" type="checkbox"/> C (gcc)         | <input checked="" type="checkbox"/> C++ (g++)         |
|         | <input checked="" type="checkbox"/> Java (openjdk)  | <input checked="" type="checkbox"/> Python (python 2) |

其中“题目集编号”是直接来自拼题 A 系统获得的，不能修改。设置好所有信息后，点击“**提交**”，就在监考系统中创建了针对该题目集的考试。**注意：**通常将监考的时间段设置得比拼题 A 中实际考试的时间段略长，例如监考时间提前 1 小时开始，方便监考老师提前登录系统。

 **OMS** | 监考系统  
Online Monitor System

chenyue

**监考面板**

**考试**

地点

客户端下载

### 考试信息

请填写考试基本信息，提交后您可进一步编辑考点及考生等考试详细信息。

|        |                                                                                                                    |
|--------|--------------------------------------------------------------------------------------------------------------------|
| 名称 *   | <input type="text" value="Final programming test"/>                                                                |
| 考试平台 * | <input checked="" type="radio"/> PTA <input type="radio"/> PAT <input type="radio"/> Zoj <input type="radio"/> POJ |
| 题目集编号  | <input type="text" value="7534"/>                                                                                  |
| 开始时间   | <input type="text" value="2017-07-06 00:00"/>                                                                      |
| 结束时间   | <input type="text" value="2017-09-01 00:00"/>                                                                      |

2. 对于初次使用监考系统、或需要更新考试地点信息的教师，点击左栏“地点”，然后创建考试地点。



OMS 监考系统  
Online Monitor System

监考面板  
考试  
地点  
客户端下载

### 地点信息

名称 \* 计算中心1-1机房

最大座位号 \* 123

提交

其中“名称”是不超过 80 个字符的字符串；“最大座位号”是指该地点机位编号的最大值（并不一定等于可用的座位总数，要求不超过 500），系统将自动在该值范围内生成考试座位号。

如果某机房有部分机位不可用，可在地点列表中点击相应机房的“修改”，进行座位设置，然后点击最下方的“保存”。如下图所示，图中的第 5、6、9、10 号座位将被禁用。



OMS 监考系统  
Online Monitor System chenyue

### 地点信息

名称 \* 计算中心1-1机房

最大座位号 \* 123

提交

座位设置

|                                     |    |                                     |    |                                     |    |                                     |    |
|-------------------------------------|----|-------------------------------------|----|-------------------------------------|----|-------------------------------------|----|
| <input checked="" type="checkbox"/> | 1  | <input checked="" type="checkbox"/> | 2  | <input checked="" type="checkbox"/> | 3  | <input checked="" type="checkbox"/> | 4  |
| <input type="checkbox"/>            | 5  | <input type="checkbox"/>            | 6  | <input checked="" type="checkbox"/> | 7  | <input checked="" type="checkbox"/> | 8  |
| <input type="checkbox"/>            | 9  | <input type="checkbox"/>            | 10 | <input checked="" type="checkbox"/> | 11 | <input checked="" type="checkbox"/> | 12 |
| <input checked="" type="checkbox"/> | 13 | <input checked="" type="checkbox"/> | 14 | <input checked="" type="checkbox"/> | 15 | <input checked="" type="checkbox"/> | 16 |
| <input checked="" type="checkbox"/> | 17 | <input checked="" type="checkbox"/> | 18 | <input checked="" type="checkbox"/> | 19 | <input checked="" type="checkbox"/> | 20 |
| <input checked="" type="checkbox"/> | 21 | <input checked="" type="checkbox"/> | 22 | <input checked="" type="checkbox"/> | 23 | <input checked="" type="checkbox"/> | 24 |
| <input checked="" type="checkbox"/> | 25 | <input checked="" type="checkbox"/> | 26 | <input checked="" type="checkbox"/> | 27 | <input checked="" type="checkbox"/> | 28 |

3. 回到“考试”，在考试列表中对应的考试项右侧选“修改”。点击“添加考点”，选择考试地点和对应的用户组，点击“保存”，就把一组学生分配到了一个考场里。还可以将本校教师（必须在拼题 A 系统中有教师帐号）加为某个考场的监考。**注意：监考系统只能导入有“学号”的“学生用户”。**



|        |                                   |
|--------|-----------------------------------|
| 地点 *   | 计算中心1-3机房                         |
| 用户组    | LL-Test                           |
|        | <a href="#">同步用户组与考生列表</a>        |
| 考生列表   | <a href="#">查看/导出/管理</a>          |
| 监考人员列表 | <a href="#">查看/管理</a>             |
|        | <input type="button" value="编辑"/> |

**注意：**如果一个教学班的学生分布在若干个不同的考场，则必须根据考场分配将学生分为若干个用户组。每个用户组内的学生不可能被自动分配到不同的考场。

4. 在“考点信息”下，有三个按钮，如下图。

### 考点信息

|                                          |                                         |                                         |
|------------------------------------------|-----------------------------------------|-----------------------------------------|
| <input type="button" value="生成全部考生座位号"/> | <input type="button" value="生成全部考生密码"/> | <input type="button" value="导出全部考生名单"/> |
|------------------------------------------|-----------------------------------------|-----------------------------------------|

- (1) 点击“生成全部考生座位号”。选择“**生成座位号**”将为全体考生生成座位；若临时新增了部分考生，则需要先在对应该考场点“**同步用户组与考生列表**”，再选择“**补充座位号**”，为新增考生分配座位，同时保持其他考生信息不变。**注意：**若从拼题 A 用户组里删除了某个考生，则**同步**后，监考系统里对应的位置会空出来，可以分配给其他需要的考生。
- (2) 点击“生成全部考生密码”。选择“**生成密码**”将为全体考生生成密码；若临时新增了部分考生，则需要先在“**同步用户组与考生列表**”、“**补充座位号**”之后，再选择“**补充密码**”，为新增考生分配密码，同时保持其他考生信息不变。
- (3) 点击“导出全部考生名单”，得到考生信息。

**至此，就完成了一场考试的监考设置。**

第二步： 下载安装监考客户端

点击“**客户端下载**”，选择合适的客户端进行下载安装：

- 选择“**下载最新版监考 PC 客户端**”，将压缩包解压复制到每一台考试用机，并将“oms-client.exe”的快捷方式复制到考试用机的桌面。**注意：**监考客户端需要 **.Net Framework 4.5.2** 版本支持，下载地址：<https://omsclient.pintia.cn/net452.exe>；考试用机的时间必须与当前时间一致。
- 在手机浏览器上打开网址 <https://oms.pintia.cn/>，登录后进入客户端下载页面，根据手机类型下载安卓或苹果客户端，进行安装。登录后点击 App 右上角菜单，根据提示操作即可。

监考面板

考试

地点

客户端下载

下载最新版监考 PC 32 位系统客户端 下载最新版监考 PC 64 位系统客户端 目前最新版本：0.9 更新日期：2017-12-03

请尽量按照机器实际环境下载！PC 端系统要求：.net 4.5.2 运行环境，Windows 7 或以上。建议内存大于等于 4G。

下载最新版监考安卓客户端 目前最新版本：0.2 更新日期：2017-08-22

下载最新版监考苹果客户端 正在等待 App Store 审核

V0.9  
2017-12-03  
本版本增加了新的监控方式，并针对 PAT 考试进行了部分 bug 修复。绝大部分已安装 0.8 版本的用户无需重新下载，程序将自动更新至该版本。

V0.8  
2017-09-01  
本版本如无重要 Bug 将是正式版前最后一个版本。本版本已含有自动更新等核心功能，监考功能进行了加强，同时也支持即将推出的一键登录等功能。0.5RC 版将在 9 月 5 日停止使用，如有使用旧版的用户请尽快手动更新至该版本。

V0.5 Release Candidate  
2017-07-23

### 第三步： 监考

1. 考试前，双击启动每台考试用机的客户端，弹出二维码。用手机 App 顺序扫描每台机器的二维码，完成登陆。正常登录后窗口应显示考生的学号、姓名、座位号，请注意核对是否与实际座位号一致。**注意：**万一 App 失效，无法扫描二维码，则需要逐一手动输入考生名单中的用户名和密码进行登录。



2. 考试前将座位分配表发布给考生，要求考生在指定座位就座。
3. 监考时，监考教师可以先在拼题 A 登录，查看学生答题情况；同时新开窗口，访问 <https://oms.pintia.cn>，在“**监考面板**”监视考生行为。

## 九、 \*代码查重

此功能仅限付费用户使用。

### 1. PTA 内部查重

在某题目集的“排名”页面点击“题目集查重”，将进入查重系统，对所有得分代码进行查重。所谓“得分代码”，是指得分等于当前得分的最早一次提交的代码。



The screenshot shows a table with columns for '排名' (Rank), '用户' (User), '答卷' (Answer Sheet), '成绩单' (Score Sheet), '正答率' (Correct Rate), '得分代码' (Score Code), '试卷分析' (Exam Analysis), '判断题' (True/False), and '单选题' (Multiple Choice). The '题目集查重' button is highlighted in red.

| 排名  | 用户       | 答卷 | 成绩单 | 正答率 | 得分代码 | 试卷分析 | 判断题 | 单选题 |
|-----|----------|----|-----|-----|------|------|-----|-----|
| 439 | 我 dsmooc |    |     |     |      |      | -   | -   |
| 1   | Lucifer  |    |     |     |      |      | 10  | 64  |
| 1   | lyw321   |    |     |     | 100  |      | 10  | 64  |
| 1   | ...      |    |     |     | 100  |      | 10  | 64  |

进入查重系统后，点击“开始查重”，将开始查重检测。



The screenshot shows the PDS Plagiarism Detection System interface. The '开始查重' button is highlighted in red. The interface displays the title 'ZJU-ADS-cy2017-WK8' and a table with columns for '题目集', '代码数', '状态', '查重范围', '提交时间', and '操作'.

| 题目集                | 代码数 | 状态  | 查重范围 | 提交时间               | 操作  |
|--------------------|-----|-----|------|--------------------|-----|
| ZJU-ADS-cy2017-WK8 | 50  | 检测中 | 全库   | 2017/8/22 19:51:23 | 题目集 |

请耐心等待至少 30 秒（启动时间较长），当刷新后发现状态变为“完成”时，点击“查看结果”，即可看到按相似度排序的列表。点击某个“提交号”可查看与之相似的代码列表。

在列表页上方点“导出报告”，稍候系统导出。结束后按钮变为“下载导出报告”，就可以随时下载文件保存到本地了。

导出报告

| 提交号                | 用户             | 题目  | 相似度 |
|--------------------|----------------|-----|-----|
| 935023136725897216 | 3160101782 成菲尔 | 6-1 | 71% |
| 934037566826582016 | 3160102243 梁超  | 6-1 | 71% |
| 932929673758810112 | 3160102935 郑鸿鹄 | 6-1 | 70% |
| 935135945677705216 | 3160103783 胡啸川 | 6-1 | 70% |
| 933177183584690176 | 3160104621 余思杰 | 6-1 | 69% |
| 935117341892644864 | 3160100921 汪会明 | 6-1 | 69% |
| 934946063579725824 | 3160102290 丁斯帅 | 6-1 | 64% |
| 933732260020064256 | 3160105047 石晓宇 | 6-1 | 64% |
| 933268898719043584 | 3160103780 李彦奎 | 6-1 | 64% |
| 935115865380196352 | 3160102346 肖文彬 | 6-1 | 63% |
| 935051926059466752 | 3160105865 徐亮  | 6-1 | 62% |
| 933166108860514304 | 3160104700 张雨霏 | 6-1 | 60% |

| 相似提交               | 用户             | 题目  | 相似度 | 来源   |
|--------------------|----------------|-----|-----|------|
| 935135945677705216 | 3160103783 胡啸川 | 6-1 | 70% | 本任务内 |
| 933177183584690176 | 3160104621 余思杰 | 6-1 | 54% | 本任务内 |
| 935051926059466752 | 3160105865 徐亮  | 6-1 | 50% | 本任务内 |
| 935111670736457728 | 3150100846 王宇蔚 | 6-1 | 45% | 本任务内 |
| 934037566826582016 | 3160102243 梁超  | 6-1 | 44% | 本任务内 |
| 935130977646252032 | 3160105367 李政彦 | 6-1 | 43% | 本任务内 |
| 935023136725897216 | 3160101782 成菲尔 | 6-1 | 41% | 本任务内 |
| 933268898719043584 | 3160103780 李彦奎 | 6-1 | 40% | 本任务内 |
| 934618920274477056 | 3160105513 常磊  | 6-1 | 36% | 本任务内 |
| 935115865380196352 | 3160102346 肖文彬 | 6-1 | 34% | 本任务内 |

点击相似提交的某个提交号，则可查看源码比对。点击学生姓名旁边的邮件图标，可以直接给该生发送邮件。

**相似度: 70%**  
**6-1: 3160102935 郑鸿鹄**

```

bool TopSort(LGraph Graph, Vertex TopOrder
{
 int toptop=0;
 int isroot[MaxVertexNum];
 int fanum[MaxVertexNum];
 for(int i=0;i<MaxVertexNum;i++){
 for(int i=0;i<Graph->Nv;i++){
 PtrToAdjVNode p;
 p=Graph->G[i].FirstEdge;
 while(p!=NULL){
 fanum[p-
 >AdjV]++;
 isroot[p-
 >AdjV]=0;
 p=p->Next;
 }
 }
 int Q[MaxVertexNum],front=0,tail=0;
 for(int i=0;i<Graph->Nv;i++){
 if(isroot[i])Q[front++]=i;
 }
 while(front!=tail){
 if(front>MaxVertexNum || t
 int now=Q[tail++];
 TopOrder[toptop++]=now;
 }
 }
}

```

**6-1: 3160103783 胡啸川**

```

bool TopSort(LGraph Graph, Vertex TopOrder
{
 LGraph a=Graph;
 int top=0;
 int deg[MaxVertexNum];
 for (int i=0;i<a-
 >Nv;i++){deg[i]=0;
 for (int i=0;i<a->Nv;i++){
 PtrToAdjVNode tmp=a-
 >G[i].FirstEdge;
 while (tmp!=NULL){
 deg[tmp-
 >AdjV]++;
 tmp=tmp->Next;
 }
 }
 int que[MaxVertexNum+10],l=0,r=0;
 for (int i=0;i<a->Nv;i++){
 if (deg[i]==0){
 que[r++]=i;
 }
 }
 while (l<r){
 int now=que[l++];
 TopOrder[top++]=now;
 }
 }
}

```

## 2. 外部查重

系统同样支持对从外部系统获得的代码进行查重检测。只要选择左栏的“外部查重”，之后点击“代码上传”

The screenshot shows the PDS (Plagiarism Detection System) interface. At the top left is the logo 'PDS 查重系统 Plagiarism Detection System'. At the top right is a user profile box with the name 'chenyue'. The left sidebar has two main sections: 'PTA 内部查重' and '外部查重'. The '外部查重' section is active. The main content area is titled '代码上传' (Code Upload). It contains three input fields: '标题' (Title) with a '测试一下' (Test) button, '课程' (Course) with a dropdown menu showing '数据结构' (Data Structure), and 'ZIP 文件' (ZIP File) with a '选择文件' (Select File) button and the text 'AC.zip'. Below these fields is a tip: '小提示：压缩包格式为 ZIP，压缩包内为代码文件，无二级目录，建议文件名为 [学号+时间.c/cpp/java等]。' At the bottom of the form is a red '提交' (Submit) button.

这里的“标题”仅为教师方便查找记录（相当于拼题 A 内部的题目集名称），选择“课程”是为避免不同课程的同一个学生的代码文件重名——系统会自动给教师上传的每个文件加前缀，前缀信息包含了学校缩写、课程缩写、教师昵称。而教师上传文件时，除了用学号命名外，最好再加上日期或者题目编号等便于区分识别的信息，否则重名的新文件将覆盖旧文件。

其他操作与“PTA 内部查重”相同。

## 附录 1：特殊裁判程序模板

```
/* 特殊裁判程序模板 */

#include <stdio.h>

#define ACCEPTED 0
#define WRONG_ANSWER 1
#define PRESENTATION_ERROR 2

int main(int argc, char const* argv[])
{
 /* 可直接复制以下语句 */
 FILE *stdi = fopen("input", "r"); /* 裁判输入 */
 FILE *stdo = fopen("output", "r"); /* 裁判答案 */
 FILE *outf = fopen(argv[1], "r"); /* 用户生成的答案 */
 int ret = ACCEPTED; /* 默认正确 */
 /* 以下添加具体裁判步骤 */

 /* 自己填写判断方法 */
 /* 在错误点将 ret 值改写为 WRONG_ANSWER 或 PRESENTATION_ERROR */

 /* 最后别忘记关闭文件 */
 fclose(stdo);
 fclose(outf);
 return ret;
}
```



## 附录 2：拼题 A 系统单位会员特权及收费标准

| 会员特权 | 内容                                                                                                                                                                                                                                                                                    | 收费标准                                                     |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------|
| 免费版  | <p>所有教师用户免费享有的基本功能包括：</p> <ul style="list-style-type: none"> <li>● 出题</li> <li>● 导入学生</li> <li>● 布置作业、考试</li> <li>● 导出成绩（包括成绩单、电子试卷、得分代码）</li> </ul>                                                                                                                                  | 免费                                                       |
| 标准版  | <p>在免费版的基础上，进一步提供：</p> <ul style="list-style-type: none"> <li>● 随机组卷</li> <li>● 监考</li> <li>● 代码查重</li> <li>● 本单位教师管理与数据统计</li> </ul>                                                                                                                                                 | 标准收费：3 万元/年<br>优惠价：<br>5 万元/2 年<br>6 万元/3 年<br>超过 3 年无优惠 |
| 标准+版 | <p>在标准版的基础上，进一步提供：</p> <ul style="list-style-type: none"> <li>● 为高校开发带有校徽的个性化门户；</li> <li>● 享受 4 年标准版使用及门户升级维护服务。</li> </ul>                                                                                                                                                          | 10 万元                                                    |
| 升级版  | <p>在标准+版的基础上，进一步提供本地化服务。包括：</p> <ul style="list-style-type: none"> <li>● 可下载本校教师原创题目备份到本地服务器；</li> <li>● 可在本地服务器导入考生备份数据，组织考试；</li> <li>● 在断网情况下，可切换到本地继续考试（但只有函数题、编程题可以在断网时得到即时反馈，其他类型的题目须在联网状态下提交到拼题 A 完成批改和计分）。</li> </ul> <p>享受 4 年升级维护服务。</p> <p><b>特别说明：</b>断网情况下无法使用监考和查重。</p> | 25 万元                                                    |
| 豪华版  | <p>为高校开发安装运行在本地服务器上的教学辅助系统，具有拼题 A 的免费教师权限享有的基本功能。享受 4 年升级维护服务。</p> <p><b>特别说明：</b></p> <ul style="list-style-type: none"> <li>● 该系统能够处理的最大并发量为 400~500，大约可以支持 2000 人同场考试。</li> <li>● 该系统包含随机组卷功能，不包括本地的监考和查重功能。但该系统可在联网状态下使用拼题 A 的监考、查重功能。</li> </ul>                                 | 65 万元                                                    |



### 附录 3：拼题 A 系统功能零售价目表

|      | 基本费（元） | 用量费（元） | 单次价格（元）                       |
|------|--------|--------|-------------------------------|
| 查重   | 50     | 5/文件   | $50 + 5 \times \text{上传文件数量}$ |
| 随机组卷 | 200    | 1/题次   | $200 + 1 \times \text{用题数量}$  |
| 监考   | 500    | 10/考生  | $500 + 10 \times \text{考生数量}$ |

注：监考外接系统将增加一次性开发费用 3000 元，永久性获得监考客户端和手机 app。

**用例 1：**每学期 200 学生，组织 14 周作业且每周查重，组织监考期中、期末两场考试，使用随机组卷，每卷 50 题。则**每学期**的费用计算公式如下：

- 查重： $(50 + 5 \times 200) \times 14 = 14700$ （元）
- 随机组卷： $(200 + 1 \times 50) \times 2 = 500$ （元）
- 监考： $(500 + 10 \times 200) \times 2 = 5000$ （元）

对应此例的情况，每年两个学期将花费 40400 元。建议加盟为会员，优惠 20000 元/年不限用量。

**用例 2：**每学期 100 学生，不做查重，仅组织监考期末一场考试，使用随机组卷，每卷 50 题。则**每学期**的费用计算公式如下：

- 随机组卷： $200 + 1 \times 50 = 250$ （元）
- 监考： $500 + 10 \times 100 = 1500$ （元）

对应此例的情况，每年两个学期将花费 3500 元，适用零售价格。

**用例 3：**考试使用外部 OJ 系统，仅引入“拼题 A”监考系统。每学期 50 学生，仅组织期末一场考试。则初次开发费用 3000 元，加**每学期**的费用计算公式如下：

- 监考： $500 + 10 \times 50 = 550$ （元）

初次使用监考的费用一共为 3550 元，此后每次 550 元。