

北京航空航天大学

CourseGrading

计算机、人工智能、大数据、信息安全专业
教学实验科研一体化支撑平台

完备支撑专业课程体系

随时随地在线实验环境

可扩展自动化评测框架

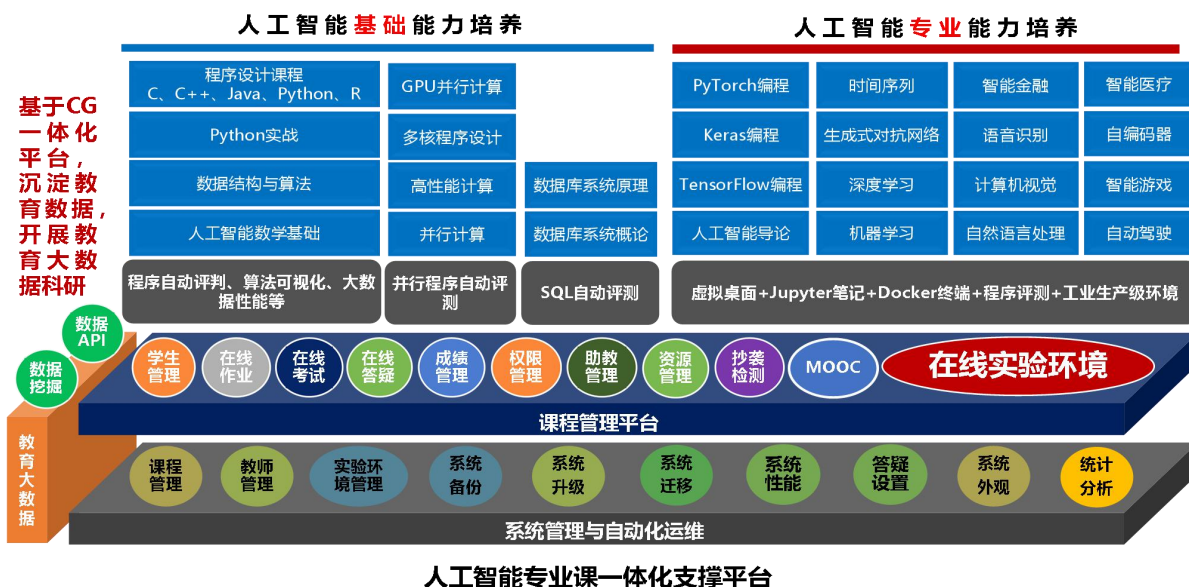
全面汇聚培养阶段数据

开放式教育大数据分析

www.educg.net

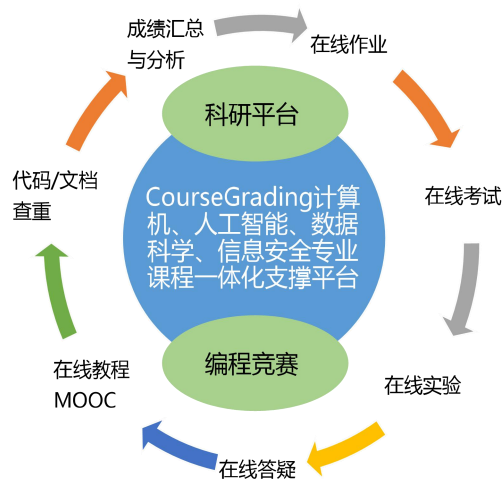
概述

“人工智能基础能力” + “人工智能专业能力” 培养，构建综合在线实验体系，提供人工智能专业教学一站式解决方案。



CourseGrading 计算机、人工智能、大数据、信息安全专业课程一体化支撑平台（简称 CG 平台）是融合课程管理、自动评判、在线实验、代码查重、文档查重、作业考试、科研竞赛、MOOC 教学、教育数据分析等功能为一体的教学实验科研平台。CG 平台服务于以下三个目标：（1）针对学生，切实提高学生工程实践能力和复杂工程问题解决能力；（2）针对教师，减轻教师评阅工作量，简化教师构建与扩展实验环境流程，使教师专注于教学与实验资源设计；（3）对于学院，通过覆盖全课程各环节的教学数据分析，呈现教学效果，辅助改进教学质量，支撑专业评估和认证。

为切实提高各专业学生的复杂工程问题解决能力，需大幅提高训练量，CG 平台对此提供以下 5 个支撑手段：（1）基于 B/S 架构的开放在线实验平台，支持任何学生任何时间和任何地点都可开展实验。（2）丰富、稳定、流畅的实验环境，包括虚拟桌面实验环境、Jupyter 笔记实验环境、Docker 实验环境、通用评测实验环境，满足不同课程场景下复杂实验项目的开展。（3）实时自动化评测技术和精准评测报告反馈机制，解决大规模训练量下的教师人工评阅与答疑的瓶颈问题，降低学生学习挫败感。（4）严格客观的监督机制，结合代码查重^{[1][6]}、文档查重、IP 监控等反作弊等技术，督促学生独立完成训练。（5）面向实际问题的实验体系，为每门课程设计了包含不同难度、覆盖教学各个知识点的实验体系，从基本原理到复杂工程问题梯度递进，引导学生逐步掌握并学会运用相关理论及工程知识。



“大数据基础能力” + “大数据专业能力” 培养，构建综合在线实验体系，提供大数据专业教学一站式解决方案。



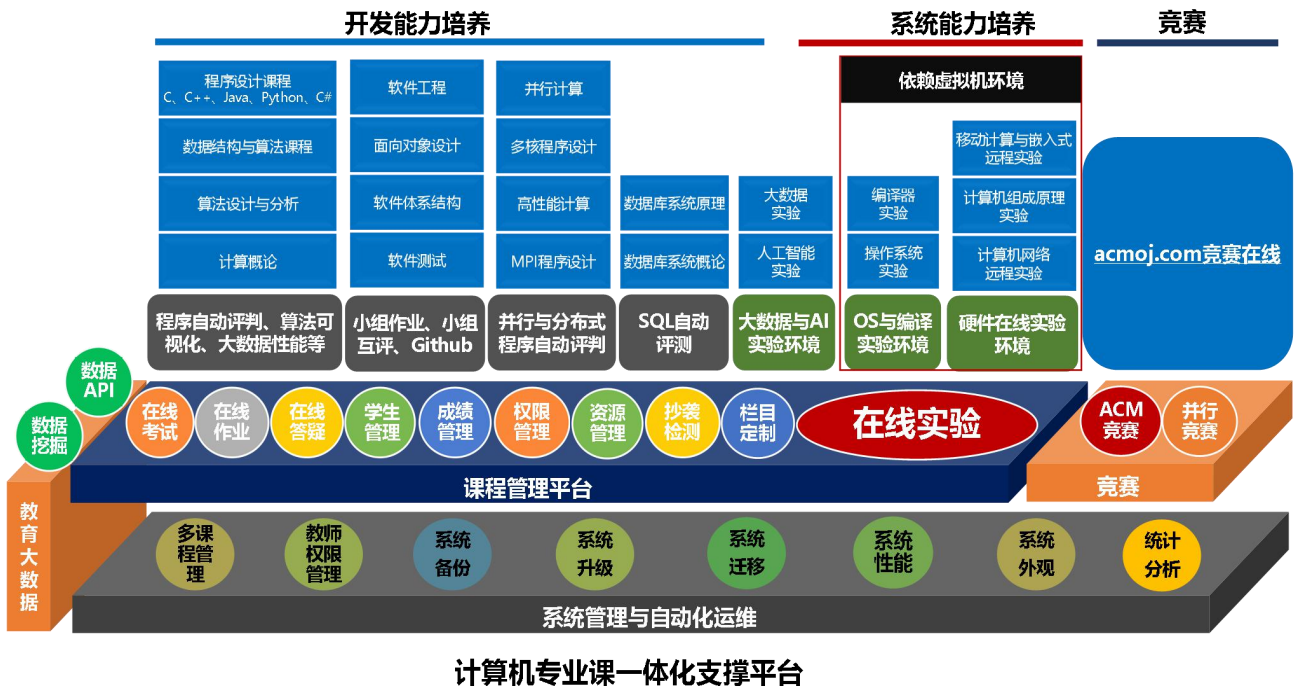
大数据专业课一体化支撑平台

为了使教师专注于教学实验资源设计，CG 平台提供了以下 3 种支撑手段：（1）基于通用自动化评测框架减轻教师评阅负担，消除教师人工评阅瓶颈，通用自动评测框架对学生的提交材料实时生成评测报告，评测报告中包含了错误原因、得分等反馈信息，引导学生自主调试和解决问题，减轻教师答疑负担。当前已支持高级程序设计语言自动化评测、算法性能自动化评测、并程序（MPI、多线程）自动化评测^[2]、SQL 语句自动化评测、操作系统内核代码自动化评测^[3]、Logisim 电路图自动化评测^[4-5]、Verilog 硬件描述语言自动化评测^[4-5]、MARS 汇编语言自动化评测^[4-5]、编译原理实验自动化评测、信息安全课程自动化评测；特别地，CG 平台开放了评测框架入口，支持评测机的插件式开发，教师可基于 Docker 或虚机为自己课程设计评测机，并作为插件接入平台。通用评测框架在后台解决了评测机的分布式部署、负载均衡、评测超时、容错等技术问题。（2）提供“开箱即用”的实验环境，教师可基于现有的实验环境镜像轻松扩展实验资源，例如基于 Jupyter 笔记实验环境，可一键导入 GitHub 上数万个覆盖人工智能和数据科学知识体系的免费、开源、高质量的 ipynb 可执行文档作为实验资源给学生练习^[8-25]；也可导入数千个覆盖 18 个行业的人工智能工业级综合案例^[26]。支持教师扩展不同实验环境下的实验镜像，并基于自定义的实验镜像构建自己的实验体系。（3）通过作业栏目的 13 种题型、考试栏目的 10 种题型、MOOC 栏目构建其他教学实验资源。

针对学院，CG 平台对专业建设中的课程体系实现了全覆盖，可促进不同课程体系间的衔接，追踪所有学生在各课程各环节中的详细学习与实验日志，完整汇集学生在整个培养阶段的学习过程数据、实验过程数据、项目实践数据、考试成绩数据，解决教育大数据的体系化和完备性的问题。通过面向全课程的大规模、细粒度的学习过程追踪和教学过程追踪，结合 CG 平台的教育大数据分析功能，呈现学院整体教学效果，辅助教学质量改进。可根据 CG 平台生成的分析结果或基于 CG 平台导出的数据进行离线分析的结果，形成专业评估和认证的支撑材料。

1 CourseGrading 平台

“系统能力”培养 + “软件能力”培养，覆盖计算机专业主要实验内容的软硬件综合在线实验体系

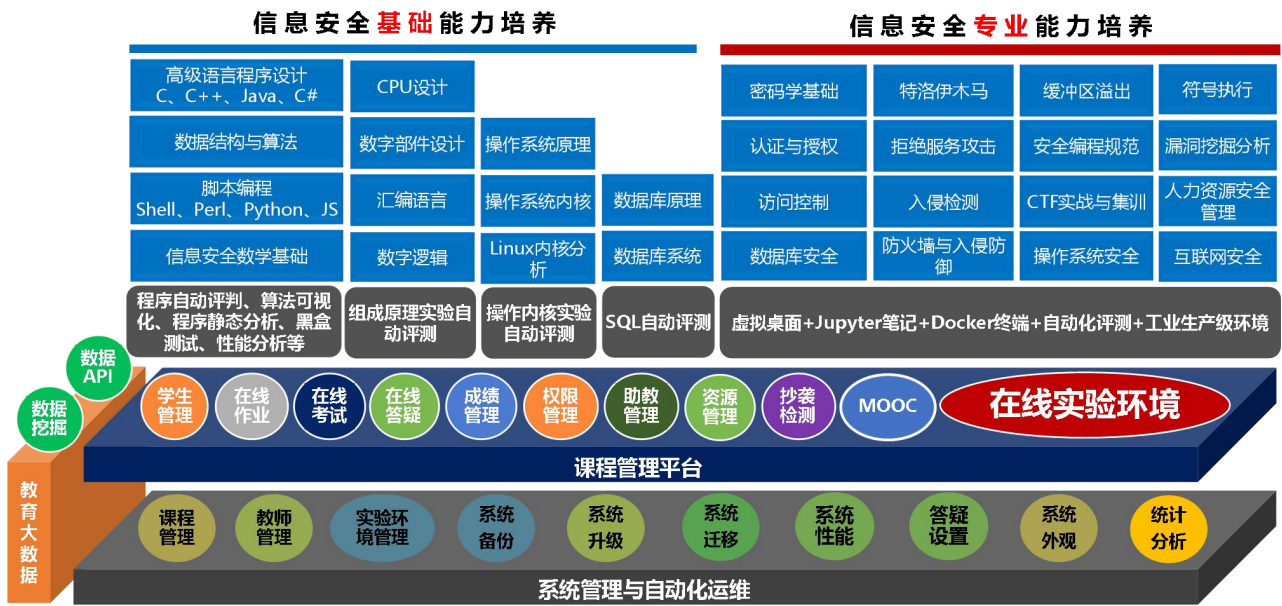


计算机专业课一体化支撑平台

CG 平台起源于北航计算机学院，并经过 10 余年的持续开发与优化，是当今功能最完善、性能最高、最易安装维护的计算机、人工智能、大数据、信息安全专业课程一体化支撑平台。CG 平台支持教学全过程的辅助、管理和监控，构建自主活泼的学习环境，促进学生主动学习和互相交流，激发学习兴趣，提高实践环节教学质量。CG 平台定位于**全面支撑**专业建设的大型综合教学实验平台，而非一个只能支撑若干门专业课程的实验系统。CG 平台对各专业建设中的计算机类基础课程和专业核心课程都提供了完备的支撑手段。例如对程序设计类课程提供了程序自动评测功能，对算法与数据结构课程提供了算法可视化、算法性能评测与热点分析等功能，对数据库课程提供了 SQL 自动评测，对并行计算与高性能计算类课程提供了多线程程序自动评测、MPI 程序自动评测等功能，对组成原理课程提供了 Logisim 与 Verilog 自动评测环境，对操作系统课程提供了内核实验代码自动评测功能；对各专业核心课程提供了稳定、流畅、丰富、灵活的实验环境，具体包括虚拟桌面实验环境、Jupyter 笔记实验环境、Docker 实验环境、通用评测实验环境，基于这些实验环境可支撑复杂实验项目的开展。

在课程体系方面，CG 平台致力于覆盖整个专业的所有课程，并针对每门课程的特点提供针对性的功能以有效支撑该课程实践环节的开展。在教学支撑方面，CG 平台具备契合于高校教学过程的课程管理系统，可提供一站式全过程教学支持。在各专业核心课程的实验方面，除了提供多种在线实验环境外，还在实验环境后端构建了工业生产级集群以支撑学生开展大型综合实验。在课程资源方面，提供了覆盖各专业知识体系的课程资源和实验资源。

“信息安全基础能力” + “信息安全专业能力”培养，构建综合在线实验体系，提供信息安全专业教学一站式解决方案。



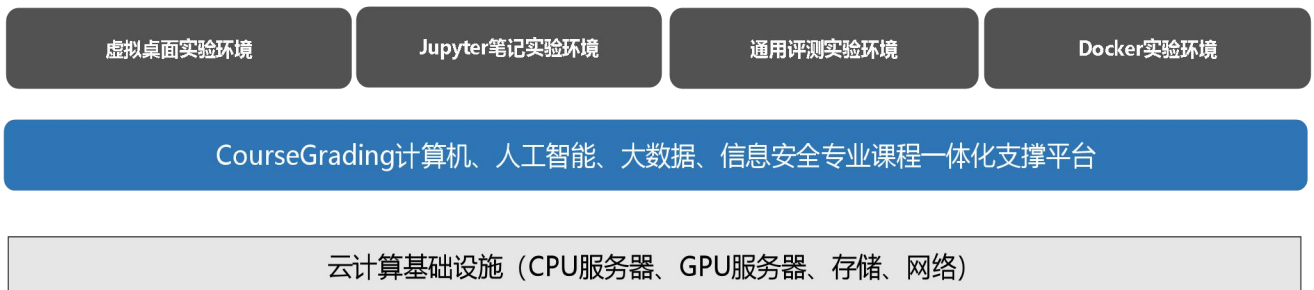
信息安全专业课一体化支撑平台

CG 平台基于 B/S 架构，所有学生活动都是基于浏览器进行，实现学生随时随地做实验的目标，使学生不再因机位数量、机房管理制度、实验排课时间、C/S 架构实验环境等因素而无法有效开展实践活动。作为教学辅助平台，CG 具有开放可扩展性，可在平台上扩展任意多门课程，可在课程中构建自己的课程资源，如题库资源、实验资源、MOOC 资源等。在 CG 平台上，可轻松接入任意多台服务器资源，通过 CG 平台的分布式自动化部署机制，将服务器构建为用于学生实验的实验环境，实现实验并发人数的弹性扩展。CG 平台不会限制并发人数，并发人数仅和服务器数量相关，接入更多服务器，即可支撑更多学生并发实验。

CG 平台除了实现教学资源 and 硬件资源按高校教学模式统一组织和管理外，还会采集平台上每门课程中每个学生和每个教师的活动数据，并将这些数据按统一格式存储于平台，形成本地教育大数据。教师可对本地沉淀的教育大数据进行挖掘，可用于评价学习过程、分析教学过程、改进教学质量等环节。

2 在线实验环境

在线实验环境概述



多种实验环境 提供多种实验环境，满足不同课程场景下复杂实验项目的开展，包括：（1）虚拟桌面实验环境；（2）Jupyter 笔记实验环境；（3）Docker 实验环境；（4）通用评测实验环境。这 4 种实验环境都是可扩展的，教师可自定义实验环境镜像，接入 CG 平台，并基于自定义的实验镜像构建自己的实验体系。

随时随地实验 B/S 架构实验系统，无需配置环境直接动手练习，打破传统实验模式的时空限制。学生分配的实验环境至少持续开放运行一个学期，学生在实验过程中所产生的数据会持久化存储，开放运行期间无需回收实验环境，保证实验过程的连续性和流畅性。

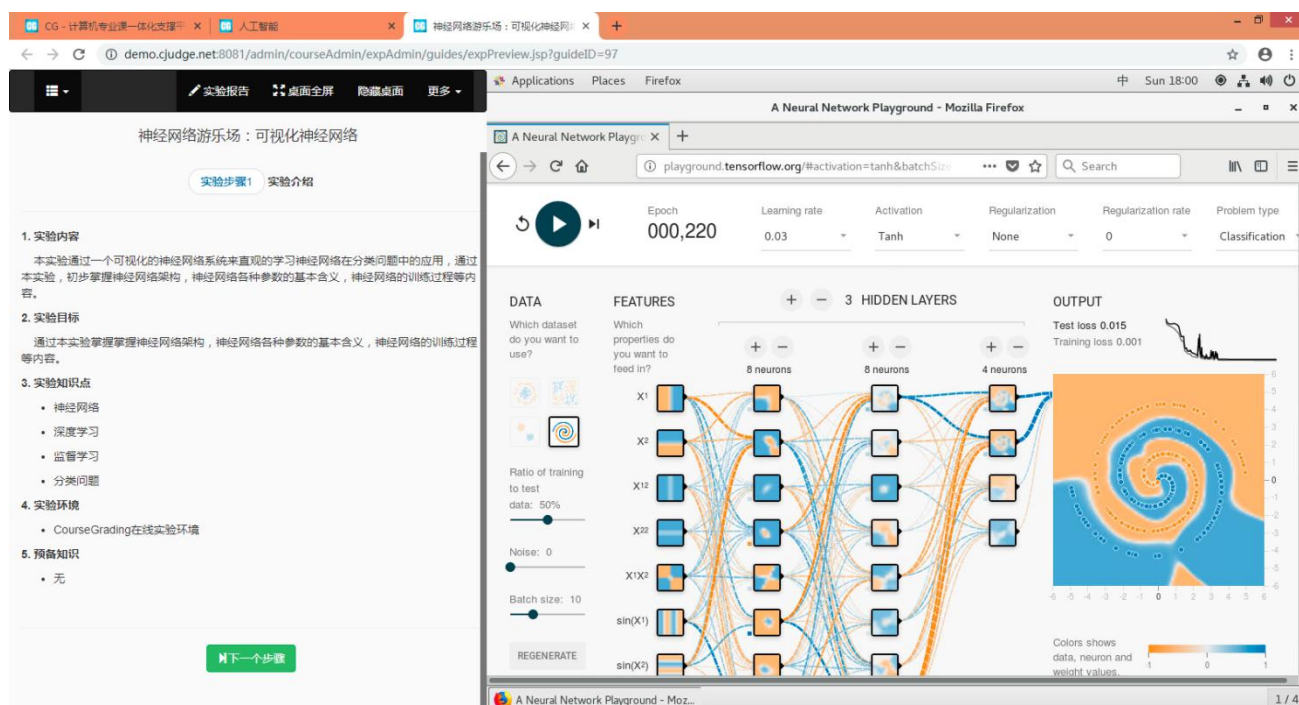
实验过程追踪 内置日志系统，细粒度追踪学生实验环节的过程数据，包括：实验进度、实验投入时间、代码行、程序性能、正确率、实验分布时间、提交次数等信息。

支持分布式部署 只需一个 IP（校级或公网）入口，支撑学生人数可无限扩展。

支持任何虚拟化技术 实验环境运行于 Linux 系统应用层，因此可对接任何虚拟化平台，包括：VMware、OpenStack、CloudStack、Docker、阿里云、腾讯云等，可充分利用学校现有的私有云或物理设备。

支持多种实验环境混合 对于部分复杂实验项目，单个实验环境无法支撑，需要通过多个实验环境混合使用才能开展。例如对于信安专业中的攻防实验、大数据专业中的 Hadoop 部署实验都需要多个节点扮演不同的功能，这种情况下可通过在虚拟桌面实验环境中放置多个 Docker 实例解决该问题。

虚拟桌面实验环境



流畅的实验体验 将实验手册、实验桌面、实验报告集成为一体的流畅实验方式，可提高学生实验兴趣。在实验桌面内置实验支撑材料。实验桌面可适应不同分辨率的屏幕，实时自适应浏览器大小变化，支持全屏显示、多人协同、桌面分享、桌面隐藏、远程拷贝、故障救援。

在线撰写实验报告 支持实验桌面一键截屏，支持使用 Markdown 编辑器撰写实验报告，支持使用 LaTeX 语法编写数学公式。

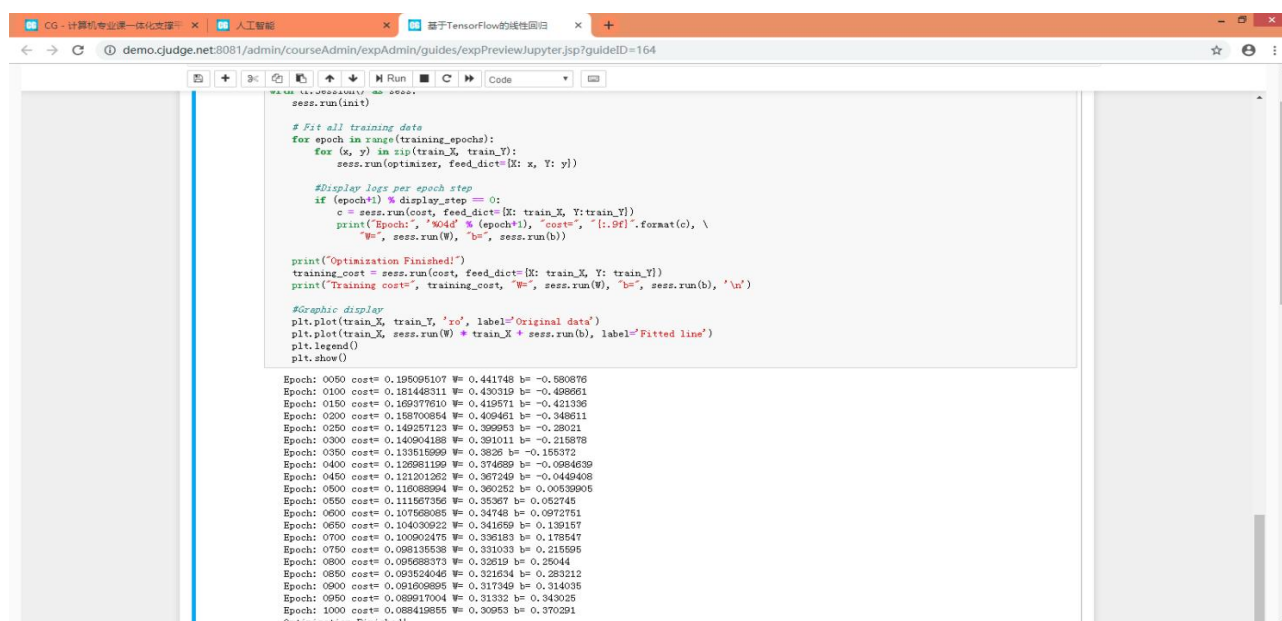
快速部署和归档 基于镜像的部署方式，可以快速恢复初始实验环境。实验资源规模可以根据选课学生数量调整，实验后能够长期保存学生实验过程资料。

高并发高性能 单台服务器支撑 300 人并发量，通过简单地配置 VNC 代理，支撑数千人并发量。本地部署后，实验桌面打开时间在 1 秒以内。在千兆网的校园网内，并发打开 1000 个实验桌面耗时在 3 秒以内。

开放自动评测接口 支持接入用户自定义的自动评测服务器，系统自动获取评测成绩。

虚拟桌面的可扩展性 教师可自己制作虚拟机镜像，并一键接入 CG 平台。教师可录入实验手册，并关联于自己的虚拟机镜像，由 CG 平台自动分配桌面给学生开展实验。

Jupyter 笔记实验环境



```
sess.run(init)

# Fit all training data
for epoch in range(training_epochs):
    for (x, y) in zip(train_X, train_Y):
        sess.run(optimizer, feed_dict={x: x, Y: y})

    #Display logs per epoch step
    if (epoch%1) % display_step == 0:
        c = sess.run(cost, feed_dict={X: train_X, Y: train_Y})
        print("Epoch: '%04d' % (epoch+1), 'cost':", "[%f]".format(c), \
              "W=", sess.run(W), "b=", sess.run(b))

print("Optimization Finished!")
training_cost = sess.run(cost, feed_dict={X: train_X, Y: train_Y})
print("Training cost=", training_cost, "W=", sess.run(W), "b=", sess.run(b), '\n')

#Graphic display
plt.plot(train_X, train_Y, 'ro', label='Original data')
plt.plot(train_X, sess.run(W) * train_X + sess.run(b), label='Fitted line')
plt.legend()
plt.show()

Epoch: 0050 cost= 0.195095107 W= 0.441748 b= -0.590876
Epoch: 0100 cost= 0.181485111 W= 0.430319 b= -0.498661
Epoch: 0150 cost= 0.162977610 W= 0.418571 b= -0.421336
Epoch: 0200 cost= 0.158700854 W= 0.409461 b= -0.348611
Epoch: 0250 cost= 0.149257123 W= 0.399953 b= -0.28021
Epoch: 0300 cost= 0.140904188 W= 0.391011 b= -0.218878
Epoch: 0350 cost= 0.133516599 W= 0.3828 b= -0.155372
Epoch: 0400 cost= 0.126981199 W= 0.374689 b= -0.0984639
Epoch: 0450 cost= 0.121201262 W= 0.367249 b= -0.0449408
Epoch: 0500 cost= 0.116088994 W= 0.360252 b= 0.00539905
Epoch: 0550 cost= 0.111873166 W= 0.35367 b= 0.052745
Epoch: 0600 cost= 0.107569085 W= 0.34748 b= 0.0972751
Epoch: 0650 cost= 0.104030922 W= 0.341859 b= 0.139157
Epoch: 0700 cost= 0.100902476 W= 0.336183 b= 0.178547
Epoch: 0750 cost= 0.098135538 W= 0.331033 b= 0.215695
Epoch: 0800 cost= 0.095688373 W= 0.32619 b= 0.25044
Epoch: 0850 cost= 0.093524046 W= 0.321634 b= 0.283212
Epoch: 0900 cost= 0.091609695 W= 0.317349 b= 0.314035
Epoch: 0950 cost= 0.089917004 W= 0.31332 b= 0.345025
Epoch: 1000 cost= 0.088419285 W= 0.30953 b= 0.370291
Optimization Finished!
```

高并发高性能 根据课程的不同，单台服务器支持 200~300 人的并发量，Jupyter 笔记实验环境打开时间在 5 秒以内。

所见即所得 无缝呈现手册和代码，学生可直接调试和运行代码，并可直接查看程序的实时运行结果。

可开展丰富的实验类型 支持图片、图像、图形、音频、视频、动画、游戏等多媒体资源，可开展图像识别、语音识别、视频目标跟踪、智能游戏等领域的实验。支持 LaTeX 语法、Matlab 编程，可开展数学基础实验。支持 HTML5、JavaScript，可制作精美的实验报告模板和交互式实验指导手册。

便捷的实验资料编辑 教师可直接编辑每个实验的实验手册和实验代码，实验代码以 Cell 的形式编排，可通过删除 Cell、修改 Cell 的方式，删除代码块或删除代码块中部分代码，引导学生根据实验指导手册重新编写空缺的代码内容，从而掌握相关知识点，达到学习目标和实验目标。

实验资源的可扩展性 可在 CG 平台上一键上传 ipynb 标准格式文档，该文档中包含实验指导手册和可直接运行的代码。可轻松导入 GitHub 上数万个免费、开源、高质量的 ipynb 可执行文档作为实验资源给学生练习；开源的 ipynb 文档覆盖机器学习、深度学习、迁移学习、强化学习、生成式对抗网络、自编码器、数据挖掘、自然语言处理、计算机视觉、语音识别、线性代数、数值分析、数理统计、优化算法、Tensorflow 编程、Keras 编程、PyTorch 编程、文本挖掘、社交网络数据挖掘、特征工程、推荐系统、数据科学、数据分析、Spark 技术等课程^[8-25]；教师也可导入数千个覆盖 18 个行业的人工智能工业级综合案例^[26]。

实验环境的可扩展性 教师可上传自己制作的 Jupyter Docker 镜像，并一键分布式部署于各 Jupyter 服务器上。然后在课程中导入 ipynb 实验文档或制作 ipynb 实验文档，并将文档关联于自己制作的 Jupyter Docker 镜像。

通用评测实验环境



The screenshot displays the CG platform's user interface. At the top, there is a navigation bar with the CG logo, the text '人工智能', and a user greeting '欢迎你, 宋馨宇!' with links for '个人信息' and '注销'. Below the navigation bar, there are links for '首页', '课程信息', '在线作业', '在线考试', '在线实验', '在线答疑', '成绩查询', and 'MOOC视频-深度学习'. A dropdown menu for '切换课程' is set to '人工智能'. The main content area shows a sidebar with '当前作业' and '历史作业' sections. The '当前作业' section is expanded to show '第一次TensorFlow编程作业'. The main content area contains the following text:

当前程序片段编程题：第一次TensorFlow编程作业—基于TensorFlow实现线性回归算法

1. 以下代码提供了基于TensorFlow实现线性回归算法的框架，数据集在程序中自行生成。该代码中有三部分空缺，分别为：

- 模型构建部分；
- 损失函数部分；
- 优化器部分。

请完成以上3部分空缺的代码，点击下方“提交”按钮，提交到后台进行自动评测。

Tips :

1. 在每个要填写完成的代码部分，已经给出相应的英文注释提示。请参考提示，认真编写代码。
2. 您提交的代码将会进行自动评测，该题目后端包含5个Test Cases，每个测试点2分，总计10分，按通过测试点数量计分。
3. 本题目中，提交的次数不受限制。
4. 请独立思考，完成题目。
5. Good Luck!

自动化评测的课程 目前支持自动化评测的课程包括：高级语言程序设计、数据结构与算法、并行计算、数据库、操作系统、计算机组成原理、编译原理、机器学习、信息安全课程体系。

提升学生工程实践能力 通过布置自动化评测作业，学生根据要解决问题，自己动脑、动手编写程序，调试程序，确保程序逻辑正确、可正确运行并得到期望结果。这一系列的自己动手实践环节，可有效提升学生工程实践能力。

减轻教师工作量 人工评阅与答疑是开展大规模训练活动的瓶颈，在某个课程中，数百个学生数次作业累计的代码行可能达到 10 万行以上，人工评阅答疑工作量非常大，成为进一步提高学生训练量的瓶颈。采用实时自动化评测技术和精准评测报告反馈机制，可减轻教师人工评阅工作量，引导学生自主调试，降低答疑工作量。教师可以多种视图查看 CG 平台给出的统计分析结果。

支撑排行榜机制 支持以教师设定的指标生成排行榜，每个学生可看到自己在排行榜上的实时位置，CG 平台会根据排行榜位置利用插值法为每个学生实时计算成绩。例如对于深度学习课程，可通过排行榜激励学生在保证模型精度的前提下，不断优化算法、调整参数，以缩短程序执行时间，提升模型在训练和执行方面的效率。

通用评测环境的可扩展性 支持教师基于 Docker 或虚机为自己课程开发评测机，教师可在评测机内集成每个实验的支撑材料、评测脚本、测试数据、验证数据。在评测脚本中设计评测指标以及根据各评测指标计算评测得分的加权公式。然后以插件的形式与通用评测框架对接，通用评测框架自动解决评测机的分布式部署、负载均衡、评测超时、容错等技术问题。评测框架将学生提交材料调度给教师评测机，并从教师评测机获取评测报告通过 Web 前端反馈给学生。

```
def grad_mse(centroid, points):
    dx = np.sum((centroid[0] - points[:, 0]) / np.sum((centroid - points) ** 2, axis=1) ** 0.5)
    dy = np.sum((centroid[1] - points[:, 1]) / np.sum((centroid - points) ** 2, axis=1) ** 0.5)
    return np.array([dx, dy])

x, y = map(float, input().split())
centroid = np.array([x, y])
pl = []
for i in range(0, 20):
    x, y = map(float, input().split())
    pl.append([x, y])
points = np.array(pl)
iterations = 20
eta = 0.03
for i in range(iterations):
    centroid1 = centroid - eta * grad_mse(centroid, points)
    centroid = centroid1
print("%.3f %.3f"%(centroid[0],centroid[1]))
```

[下载源文件](#) [在线浏览源代码](#) [更新时间 : 2019-05-21 17:58:44](#) [详细评判结果 >>](#)

共有测试数据:5
平均占用内存:24.580K 平均运行时间:0.72949S

测试数据	评判结果
测试数据1	完全正确
测试数据2	完全正确
测试数据3	完全正确
测试数据4	完全正确
测试数据5	完全正确

Docker 实验环境

```
jovyan@7853c2b2229b:~$ ls
01.ipynb  hello_world.py  linear_regression.ipynb
jovyan@7853c2b2229b:~$ cat hello_world.py
print("hello world!")
jovyan@7853c2b2229b:~$ python hello_world.py
hello world!
jovyan@7853c2b2229b:~$ python
Python 3.6.3 [Anaconda, Inc.] (default, Nov 9 2017, 00:19:18)
[GCC 7.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> print("hello world!");
hello world
>>> import tensorflow as tf
>>> a=tf.constant([1,2,3])
>>> b=tf.constant([1,2,3])
>>> c=a+b
>>> sess=tf.Session()
>>> sess.run(c)
array([2, 4, 6], dtype=int32)
>>>
```

基于Docker的Jupyter Notebook实验环境，学生可直接进入Docker中的Linux终端，并在终端上运行命令进行实验。

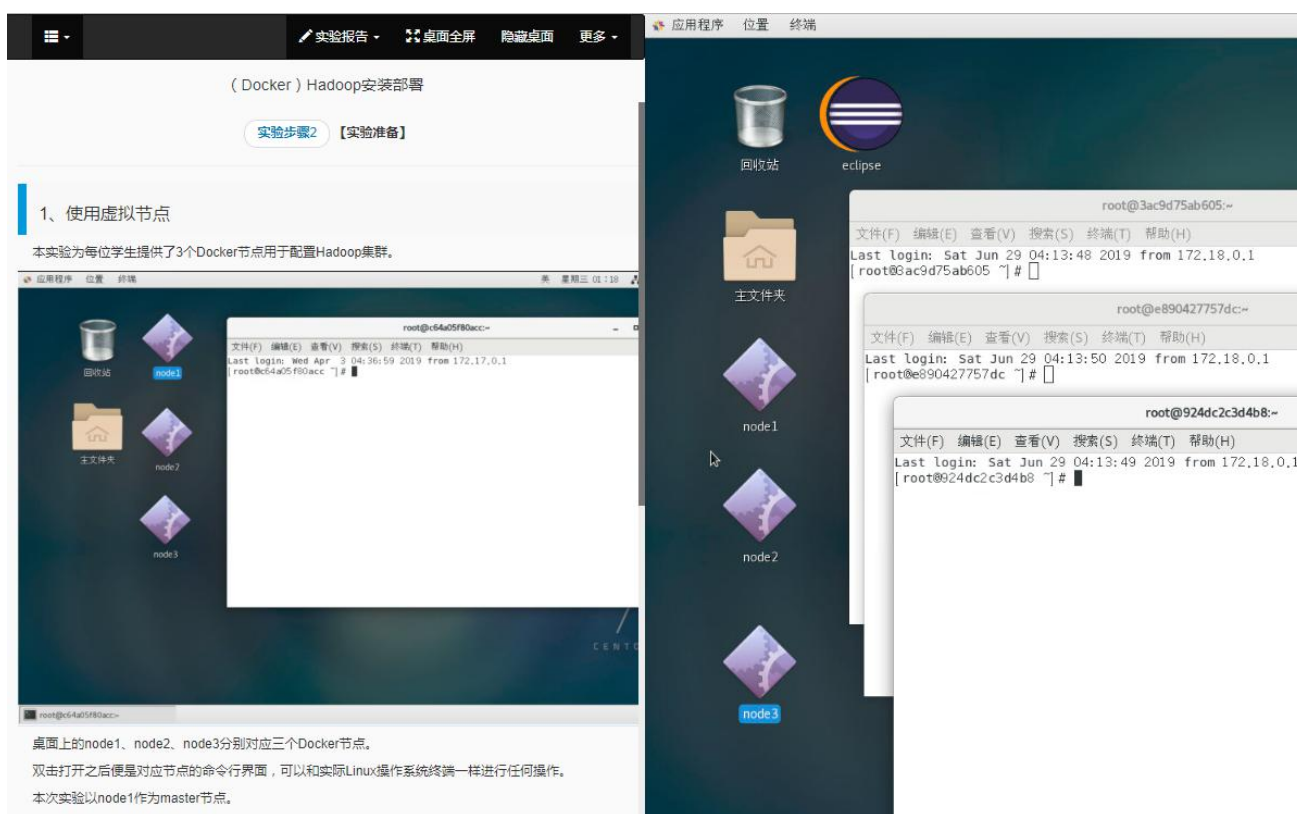
以上命令分别为：

1. 查看当前目录文件。
2. 查看Python脚本内容。
3. 运行Python脚本。
4. 打开Python终端，导入TensorFlow库，使用TensorFlow执行两个向量的加法。

在线打开 Docker 字符终端 基于 Docker，提供轻量级的 Linux 字符终端交互式实验环境，学生可在字符终端上键入相关命令进行实验。Docker 实验机是基于 Jupyter Docker 镜像而扩展的新镜像，其最大的好处是可以利用 Jupyter 提供的在线 Docker 字符终端访问功能和文件管理功能。

Docker 实验环境的可扩展性 支持教师上传自己制作的 Docker 镜像用于学生实验,CG 平台负责管理 Docker 实验机的启动、分配、存档、销毁。

与虚拟桌面相结合 对于需要多个 Docker 才能开展的复杂实验项目，可通过与虚拟桌面相结合，实现复杂实验项目的设计。例如，对于信安中的攻防实验，攻击端和防御端可运行在不同的 Docker 实例中，学生打开虚拟桌面实验环境后，分别打开两个 Docker 终端，开始攻防实验。对于大数据课程中的 Hadoop 部署实验，学生打开虚拟桌面实验环境后，可打开三个 Docker 终端，开始安装实验，如下图所示。

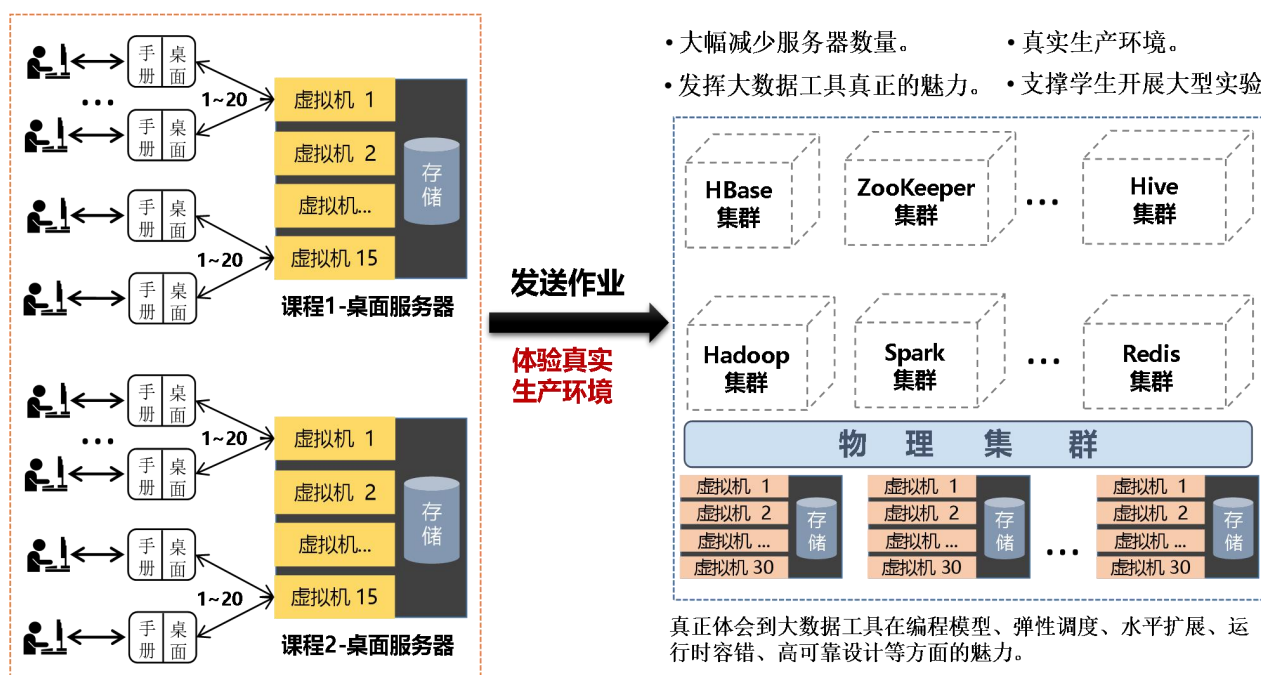


3 后台实验架构

大数据实验架构

传统的大数据的实验模式是给每个学生分配 1 台虚拟机，在虚拟机内预装大数据的实验工具。受限于单台虚拟机的存储能力和计算能力，每个学生只能进行简单的“伪大数据”实验。CG 平台构建了工业生产级实验环境，如下图所示，每个学生都可以使用工业级生产环境集群开展实验。和传统的实验方式相比，CG 实验架构具有以下几方面优势：

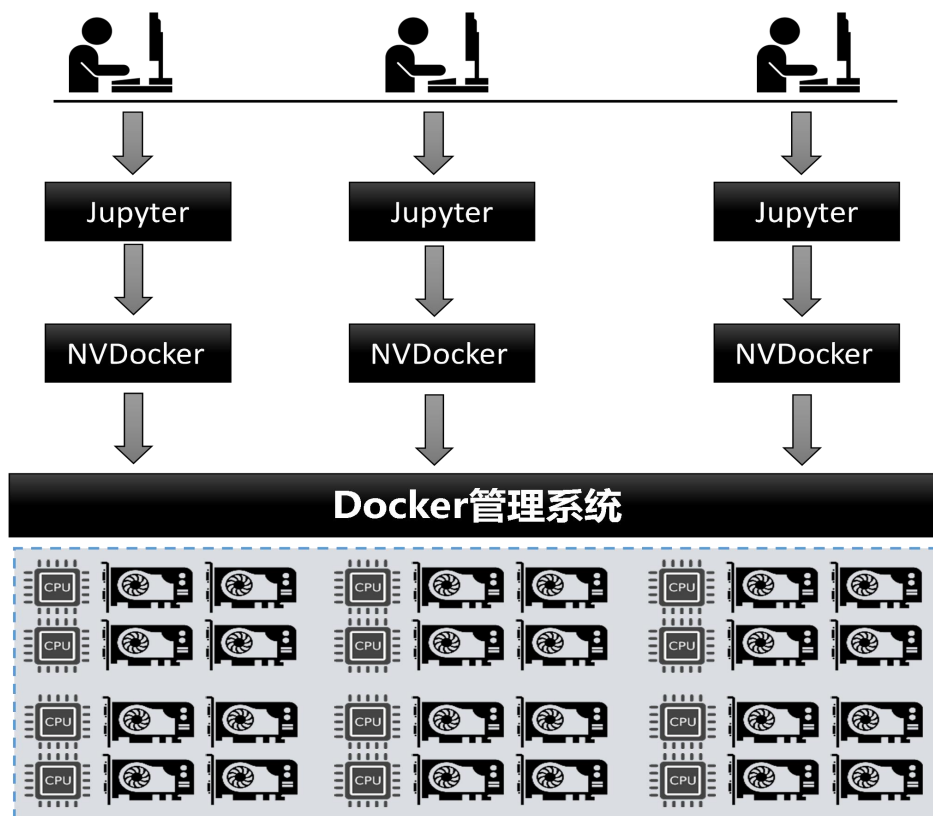
- **大幅减少服务器数量。**采用桌面环境和计算环境分离的架构，可大幅降低对硬件服务器的资源要求。下图中左侧的桌面服务器提供了学生实验的虚拟桌面环境，右侧的作业服务器提供了运行作业的计算环境。在 CG 平台的工业生产级实验架构中，每个桌面服务器上创建了 15 个虚拟机，每个虚拟机支持 20 个学生同时在线实验，即每台服务器可支撑 300 人同时进行基于虚拟桌面的在线实验。CG 大数据是目前唯一能够做到单台服务器支持 300 人并发的虚拟桌面在线实验环境。
- **真实生产环境。**如果提供 2 台 30 核的作业服务器，可构建包含 60 个节点的虚拟机集群。学生的作业将在 60 节点规模的集群上运行，而不是在 1 台虚拟机中运行。
- **体会大数据工具真正的魅力。**事实上，如果集群的规模不够大，大数据工具的处理性能比单机上由 C 语言实现的具有同样功能的程序的性能还要差很多。只有作业集群的规模足够大时，学生才会体会到大数据工具在编程模型、弹性调度、水平扩展、运行时容错、高可靠设计等方面的魅力。
- **支撑学生开展大型实验。**工业生产级集群为每个学生提供了更强大的存储能力和计算能力，为学生开展大型大数据实训项目提供了基本条件。



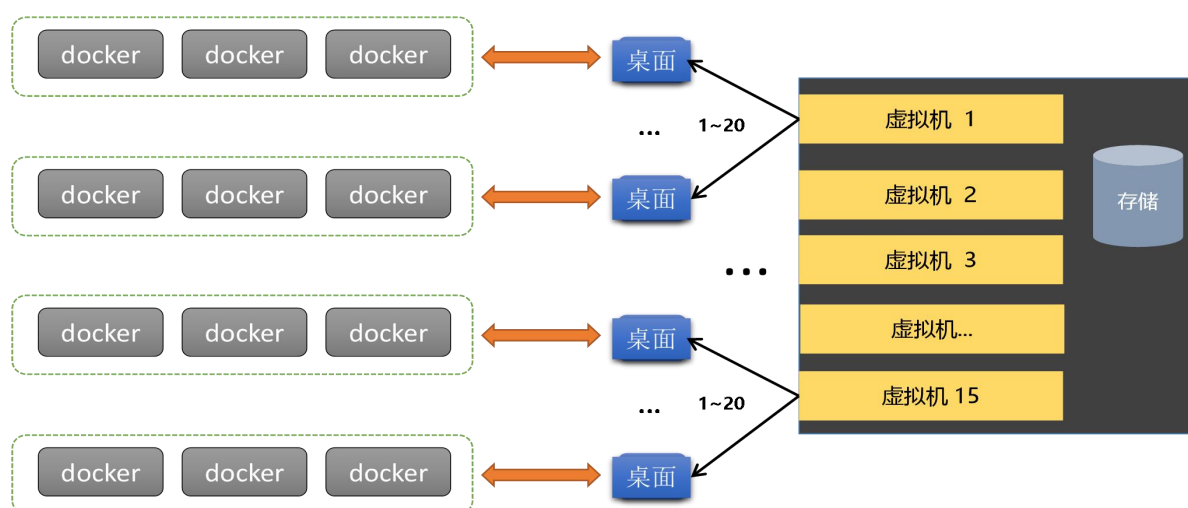
人工智能实验架构

人工智能的相关实验主要通过 Jupyter 笔记实验环境进行支撑。对于人工智能实验中的深度学习作业，CG 平台支持接入 GPU 加速卡，以提升模型训练效率。对于需要依赖于 GPU 的实验，学生在打开 Jupyter 笔记页面后，会通过 NVDocker 环境自动接入 GPU 加速卡。学生可在 Jupyter 笔记页面上编写基于 GPU 加速模型训练的代码，直接运行于 GPU 设备上，学生和教师不必再为使用 GPU 而进行繁琐的环境配置。CG 平台实现了对 GPU 资源的两种调度方式：

- **支持单卡多人的调度模式。**支持在同一个 GPU 上运行多个学生的深度学习作业，充分提高资源的利用率。
- **支持排队调度。**当学生人数大于 GPU 数量时，采用排队机制，出现空闲 GPU 时，以 FCFS 方式进行资源分配。



信息安全实验架构



如上图所示，采用虚拟机与基于 Docker 的资源配额技术对参与者进行均衡的资源分配，从而能够避免单一攻击者对网络资源、计算资源、存储资源的过度消耗，从而影响其他实验者，例如分布式拒绝访问攻

防实验在其他实验平台上不能很好的完成，因为单一的实验者很容易把网络及磁盘资源耗尽。同时，平台采用资源隔离技术和访问控制技术防止实验者进行通过虚拟机逃逸对实验环境进行破坏。

硬件配置测算

服务器资源配置与在线实验课程的数量是正比关系，计算资源配置要保证所有学生能够随时随地进行在线实验。假设大数据专业每届有 300 名学生，对于某门需要进行在线实验的课程，只需要两台物理服务器，其中 1 台为桌面服务器，另 1 台为作业服务器。假设该专业每届有 N 名学生，有 M 门课程需要在同一学期内基于虚拟桌面实验环境进行在线实验，且这 M 门课程的最大学生容量都为 N，所需服务器数量 S 如式 1 所示。

$$S = \left\lceil \frac{\left\lceil \frac{N}{20} \right\rceil \times M}{15} \right\rceil + 1 \quad (1)$$

而传统的“独占式”实验方式（每人一台虚拟机）所需要的服务器数量 S'如式 2 所示。

$$S' = \left\lceil \frac{N \times M}{15} \right\rceil \quad (2)$$

根据以上两式，可计算出不同人数和不同课程数量所需要的服务器数量，下表对这两种方案做了详细的算例对比。

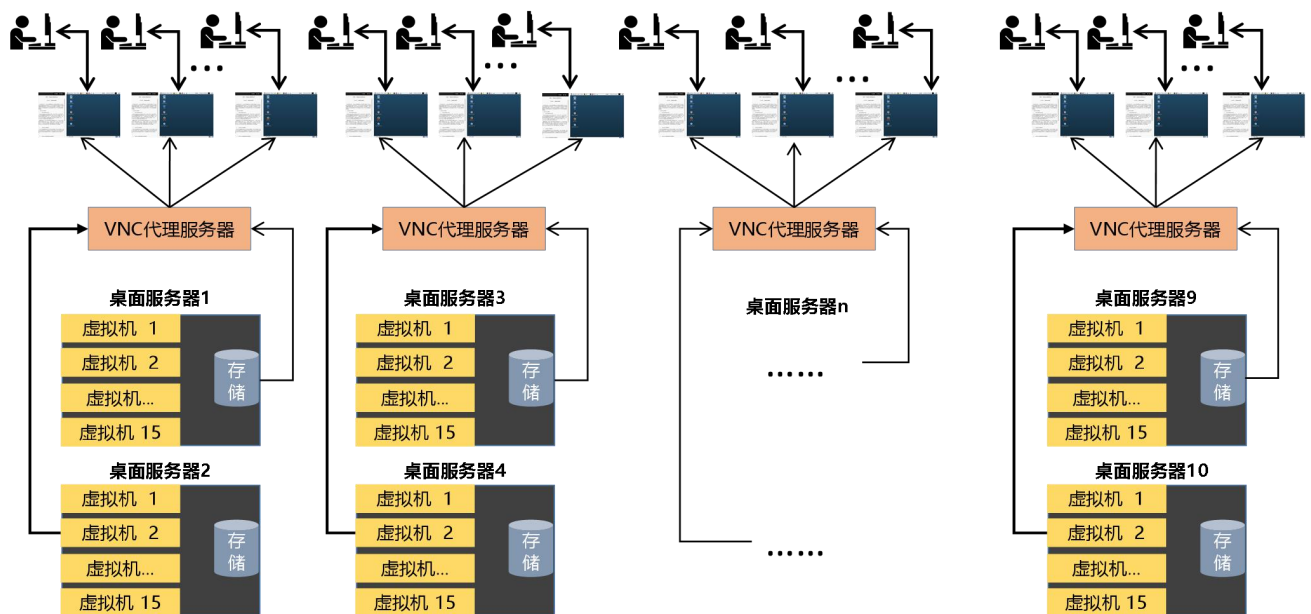
每届人数	同时开课数量	CG 方案服务器数量	“独占式”方案服务器数量
40	1	2	3
40	3	2	8
40	5	2	14
70	1	2	5
70	3	2	14
70	5	3	24
100	1	2	7
100	3	2	20
100	5	3	34
200	1	2	14
200	3	3	40
200	5	5	67
300	1	2	20
300	3	4	60
300	5	6	100

从上表中可以看出，CG 实验架构方案，相比独占虚拟机的方案，对资源的需求可以呈几倍到十几倍的减少。CG 实验架构综合以下 3 种策略，有效地降低了服务器资源的数量。

- **多用户。**多个用户共用同一个虚拟机，学生初次做实验时，动态绑定一个空闲的操作系统账号。
- **职能分离，分时共享。**大数据实验对计算资源（内存+处理器）有着较高的需求。通过将计算职能从虚拟桌面环境中独立出来，组建一个专用的高性能计算集群，所有人共享该集群，从而进行真正意义上的大数据实验。
- **实例重用。**将一门课程所需要的实验支撑材料全部集成到一个镜像内，布置实验作业时，系统可重用之前实验作业分配的同一个实例，即学生始终使用同一个虚拟机实例完成所有的实验作业。

此外，CG 平台可与高校现有的服务器资源或私有云资源对接，实现软件平台和硬件资源的解耦合，从而达到软件和硬件的独立建设、独立维护、独立升级换代的目标。在传统的一体机模式中，服务器和实验系统紧密耦合，导致后期难以维护和升级。

数千人并发



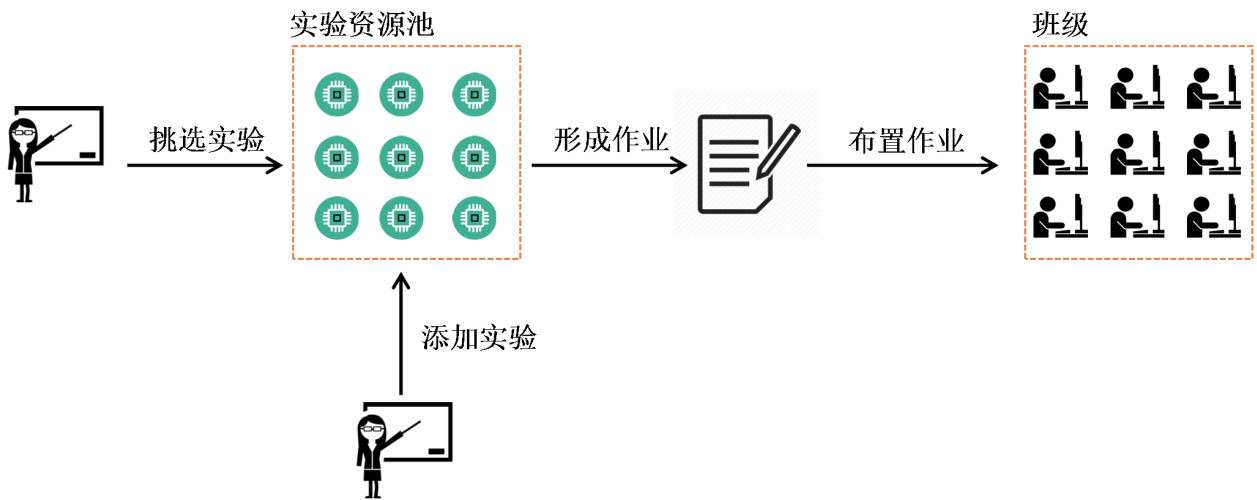
对于需要在全校开展人工智能与大数据基础课的高校，CG 平台提供了支撑数千人同时在线实验的解决方案。CG 平台通过引入 VNC 代理服务器，分担虚拟桌面的网络带宽压力，实现了对数千并发量的支撑。上图为支持 3000 人并发在线实验的部署架构示意图。

- 每个学生需要 1Mb/s 的带宽以保证虚拟桌面的流畅性。
- 3000 人同时在线实验，需要 10 台桌面服务器来提供 3000 个学生同时在线实验的虚拟桌面环境。
- 3000 人同时在线实验的总带宽需求为 3Gb/s，若 VNC 服务器为千兆网卡，通过 4 台 VNC 代理服务器来实现网络流量的负载均衡。

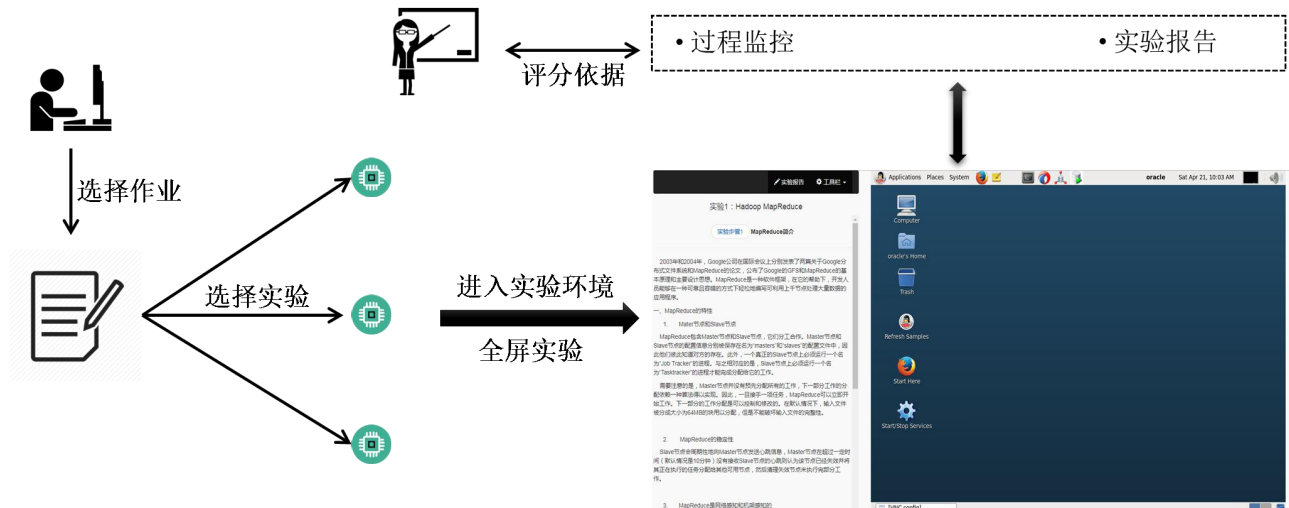
引入 VNC 代理服务器，除了解决数千人同时在线实验的问题外，还可解决学生在全球任意位置随时登陆平台进行在线实验的需求。由于 VNC 代理服务器实现了 CG Web 服务器和内网桌面虚拟机的连接，只需给 VNC 代理服务器配置公网 IP 地址，即可解决在全球任意位置都可进入在线实验环境的问题。在 CG 平台上，只需填入目标服务器 IP 地址和 root 密码即可，平台会自动将其构建为 VNC 代理服务器。

4 实验过程管理

实验过程管理

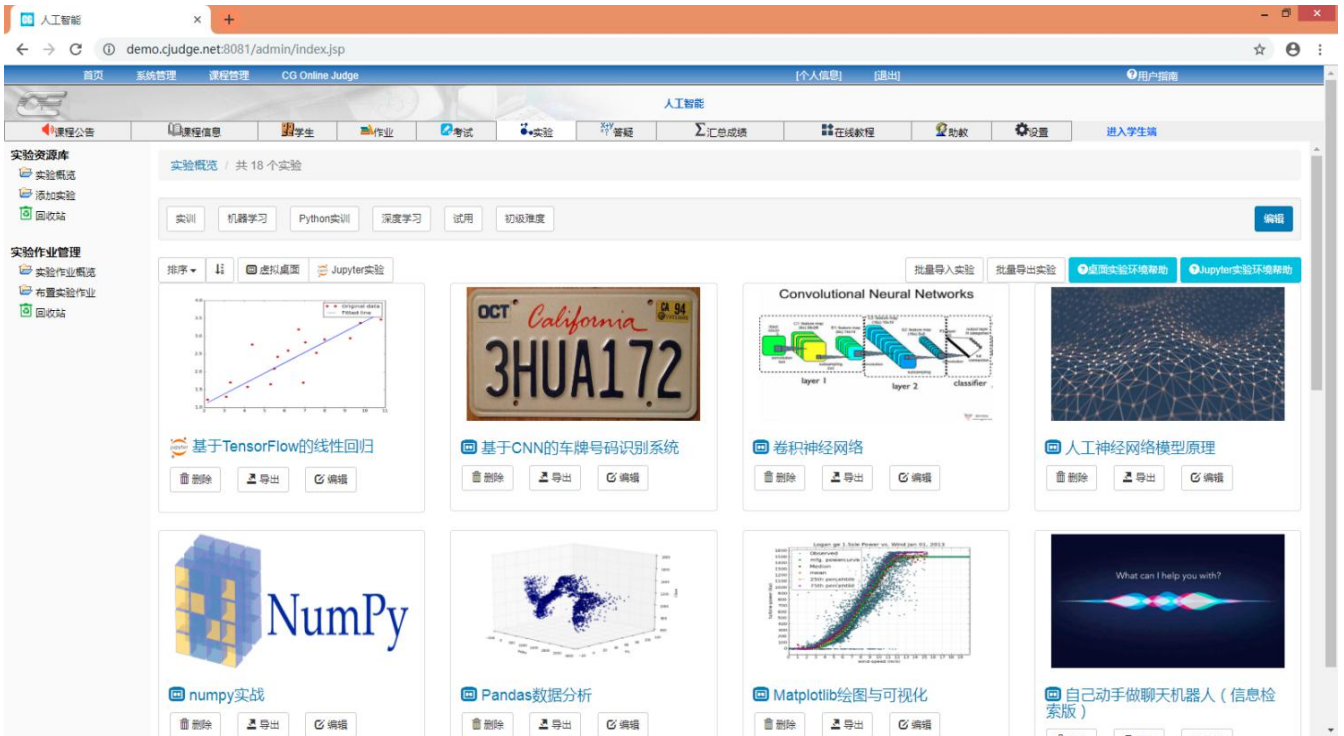


在 CG 平台上，教师可从实验资源池中挑选实验资源给学生布置实验作业。学生进入平台的实验栏目后，可查看实验作业，选择某个实验作业中的某个实验进行实验，在学生实验过程中，平台会进行全过程监控。学生可在线撰写实验报告，在撰写实验报告过程中，支持一键虚拟桌面截屏，并将截屏图片插入到实验报告中，支持基于 Markdown 编辑器和 LaTeX 语法撰写实验报告。教师可在教师端查看和评阅学生撰写的实验报告，查看系统追踪记录的学生实验过程数据（如实验投入时间，实验进度等），也可直接进入学生的实验环境查看学生编写的代码和程序的运行结果。综合以上信息，教师可对学生的实验进行综合评分。



实验资源概览

教师可在“实验”栏目下点击“实验概览”，查看当前已导入的或已创建的实验资源。实验资源分为两大类：虚拟桌面实验资源和 Jupyter 实验资源。所有实验资源可按照标签进行筛选，每个实验资源都可进行导出、编辑、删除等操作。实验资源概览的界面如下图所示。

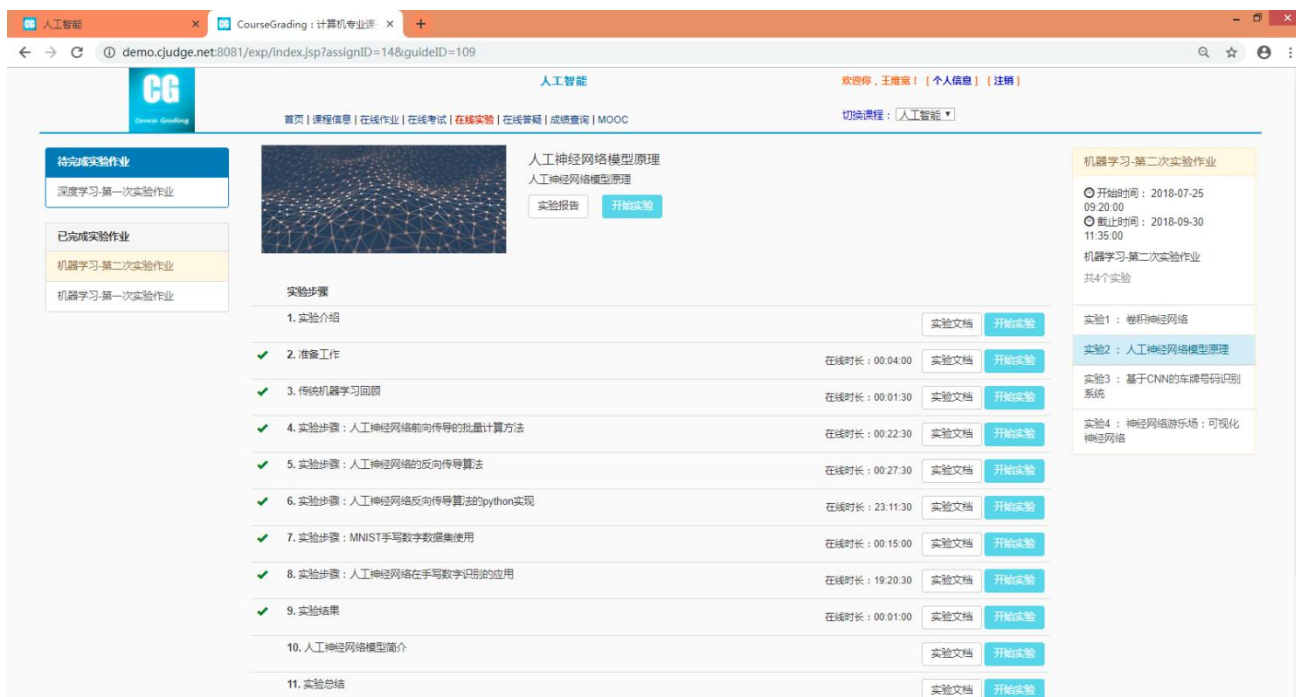


布置实验作业



布置实验作业的界面如上图所示，布置实验作业时，可选择要布置的班级、作业开始时间、作业结束时间、需学生完成的实验以及每个实验的分数。布置实验时，可选择“持久化”选项和“实例重用”选项，选项的具体含义参考图中文字。

学生进行实验作业

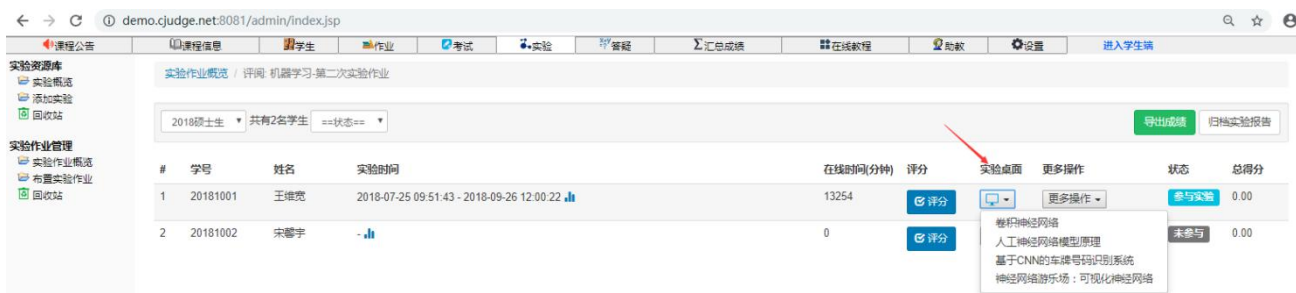


学生登录后，点击“在线实验”栏目，即可看到当前的实验作业列表，选择一个实验作业列表后，可看到该实验作业中包含的所有实验。点击某个实验，会在当前页面上显示该实验的介绍信息、每个实验步骤上的停留时间、实验进度信息、每个实验步骤的指导手册以及当前完成的实验报告。点击“开始实验”或者选择某个步骤点击“开始实验”，即可进入在线实验环境。

教师评阅实验作业

教师可在线评阅实验作业，如下图所示，在“实验作业概览”中，可以查看到当前所有的实验作业。选择某个实验作业，点击“评阅”后即可进入实验作业评阅页面。





在该页面上，可对每个学生进行单独评分，该界面上显示每个学生的实验投入总时间，实验状态以及得分信息。教师可以通过点击“实验桌面”图标，直接进入学生的虚拟桌面查看学生编写的代码或者运行结果，教师还可查看学生的实验投入时间、实验报告，如下图所示。



5 实验体系

计算机专业实验体系

程序设计类课程

- **代码自动实时评判** 支持 C、C++、Java、Python、C#、Pascal, Fortran 等语言的自动评判；支持实时评测，提交代码后立即反馈评测结果；支持代码综合剖析，可对提交代码从动态测试、代码风格、性能热点、静态分析、错误原因等角度进行全面综合剖析；支持输出结果模糊比对；支持输出结果格式错误检测；支持多文件打包上传评测；支持编程题、接口编程题、程序片段题、算法可视化等 4 种编程题型，可从多角度考察学生编程能力；支持标准输入、文件输入、命令行参数输入等 3 种输入方式；支持标准输出、文件输出等 2 种输出方式；支持设定程序最长运行时间；支持在线可视化编辑测试数据；支持程序运行过程中，捕获中间变量、表达式、数组的值并用于评测；支持反馈编译结果和运行时信息（如占用内存、CPU 时间、墙时间等）。
- **高质量题库资源** 历经 10 余年，教学中积累的数千道高质量程序设计题库，包含多种题型，覆盖所有知识点，是北京航空航天大学等高校历届学生考试和作业、推免研究生机试、考研复试中使用的题目。

数据结构与算法类课程

- **大数据性能评测** 支持上传大规模（数百 GB）文本文件或二进制文件作为程序评测输入；支持以程序性能作为评判指标，可设定程序性能占题目得分比重；提供排行榜激励机制，性能得分会以排行榜的形式展现，促进学生不断优化自己的算法。
- **数据结构与算法可视化** 支持对教学中的典型算法进行交互式模拟跟踪；支持栈、队列、堆和递归、索引、排序、图论、动态规划等共 45 种教学中常见数据结构和算法的交互可视化。
- **题库资源** 提供源自北京航空航天大学等高校的历经 10 余年教学所积累的包含多种题型、覆盖所有知识点的数千道高质量数据结构与算法题库。提供面向数据结构和算法设计类课程的大数据性能评测综合案例，输入数据量在百万级规模。

并行计算与高性能计算类课程

- **并行程序自动评判** 支持针对 SMP 架构的多线程并行程序和集群架构的 MPI 并行程序的自动评判；支持从正确性、性能和可扩展性三个方面对学生提交的程序进行自动评判；支持设定正确性、性能、可扩展性 3 个指标占程序并行得分权重；支持评判结果以图表形式可视化显示；支持排行榜激励机制；支持生成详细的评测报告。
- **虚拟评判环境** 基于虚拟机构建集群，将学生提交的并行程序发送到集群上运行以测试其正确性、性能以及可扩展性，可扩展性分为多核可扩展性和多节点可扩展性。
- **题库资源** MPI 并行编程案例题库包含矩阵相乘，MD5 密码破解等题目；多核并行编程案例题库包含积分求和，MD5 密码破解等题目。

操作系统

- **内核实验体系** 以培养学生的系统能力为指导，以让学生能够自己动手实现一个小型操作系统内核为目标，制定了内核实验体系。该体系包含启动和初始化、内存管理实验、进程管理、系统调用、文件系统、命令行解释程序等 6 个完整实验。通过将这 6 部分实验衔接起来，成为一个可运行的操作系统。
- **编写真实内核源码** 基于 Linux 0.11 内核源码设计的实验，在每个实验中，将内核源码中的部分核心代码挖空，通过给学生提供详细的实验指导手册，引导学生重新实现挖空部分的代码。
- **实验结果自动评测** 学生向平台提交实验代码后，评测系统会依次完成编译、仿真运行，并根据运行结果的控制台输出进行测试，最终给出评测结果。
- **实验过程自动管理** 每个实验满分为 100 分，学生提交代码评测的分数在 60 分以上时，系统自动为该学生推送下一个实验。
- **丰富的实验体系** 内核实验难度较大，因此还提供了较低难度的操作系统实验，包括操作系统模拟实验，Linux 应用实验（Bash 脚本和常用命令）。

组成原理

- **基于 B/S 架构的自动评测系统** 自动化评价学生提交的实验代码。在线自动化测试使得严谨完善的测试成为可能，同时测试结果的统计分析有力支撑了教学设计改进。
- **计算机组成实验体系** 涵盖 4 大类实验：数字逻辑、汇编语言、数字部件设计、CPU 设计。数字逻辑包含 14 个实验，汇编语言包含 13 个实验，数字部件设计包含 6 个综合实验，CPU 设计包含 4 个大型综合实验。所有实验支持自动评测。

数据库

- **SQL 自动评测** 支持数据定义语句，数据更新语句，数据基本查询语句，数据高级查询语句。
- **在线实验** 基于虚拟桌面的数据库实验环境，实验资源覆盖安全性、完整性、数据恢复、并发控制、存储过程、数据库监控与性能优化、视图与索引、数据库应用开发、数据库设计等实验。
- **自己动手编写数据库** 基于在线实验环境的大型综合实训案例，要求学生根据实验指导手册，自己实现一个类 SQLite 的小型内存式数据库，要求支持最常用的增删改查 SQL 语句。
- **题库与实验资源** 数百道 SQL 自动评测题库，覆盖常用的 SQL 语句类型。

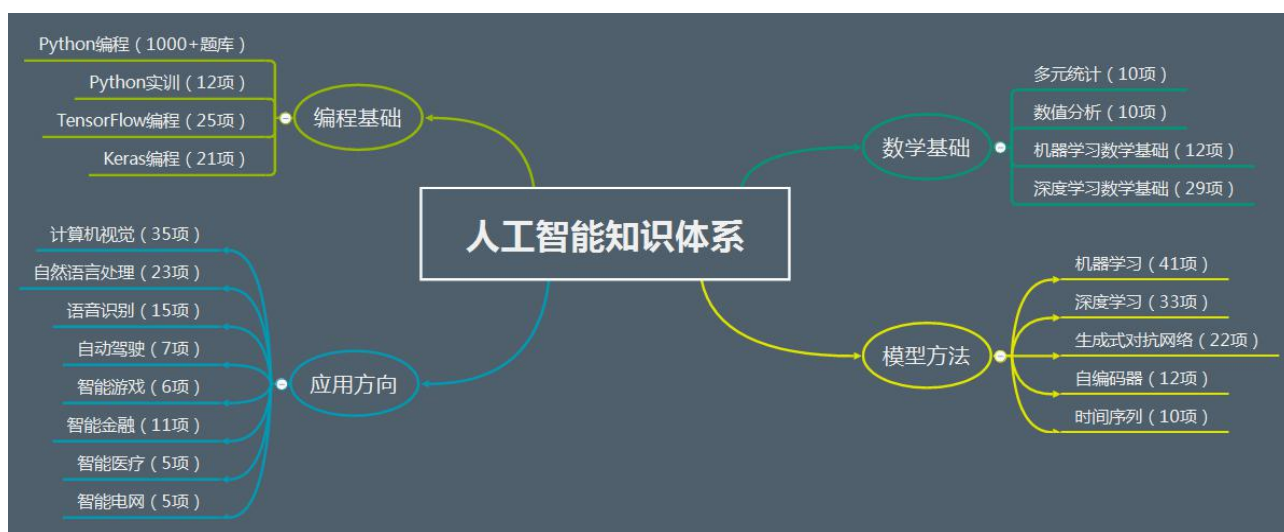
软件工程

- **全面支撑软件工程课程的教学和实验特点** 支持小组协作、小组互评、文档审查、代码审查、GitHub 项目管理以及项目增量式迭代开发。
- **在线开发环境与项目资源** 基于虚拟桌面在线实验系统，推出了一个“成为全栈工程师”的综合项目案例，这个案例让学生能够贴近工业应用，同时难度适当，让学生在学习软件工程方法的同时，也了解当今互联网应用的核心技术，而且可以根据各自的特长进行协作开发。

编程竞赛

- **独立的编程竞赛系统** CG 平台包含独立的 ACM 编程竞赛模块 CG-OJ，该模块与 CG 教学平台统一账号登录，助力于将本校学生的 ACM 编程竞赛数据沉淀于学校本地而非第三方网站。CG-OJ 模块采用国际流行的 OJ 系统风格，包含题库、排行榜、竞赛、讨论区等栏目，全面支撑本校学生的 ACM 集训和比赛。
- **完善的竞赛机制** 完全支持 ACM-ICPC 竞赛规则；支持公开赛和内部赛两种竞赛形式；支持普通账号录入题目和发起创办比赛申请，题目和比赛需要管理员审核。
- **支持代码查重** 为了保证竞赛的公平性，将 CG 教学平台的代码查重功能引入到 OJ 模块，比赛结束后，一键代码查重。
- **题库资源** 3000 道高质量的训练题库，覆盖所有知识点。每道题目附 C++ 样例代码，并标注知识点和难度等级。

人工智能专业实验体系



人工智能知识体系如上图所示，划分为编程基础、数学基础、模型方法以及应用方向 4 大类别。编程基础类别中的课程目标是掌握人工智能相关编程语言、框架以及库的学习，具体包括 Python 语言，TensorFlow 以及 Keras 等。数学基础包括多元统计、数值分析、机器学习数学基础、深度学习数学基础等内容。模型方法包括机器学习，深度学习，生成式对抗网络、自编码器以及时间序列 5 部分内容。应用方向包括计算机视觉、自然语言处理、语音识别、自动驾驶、智能游戏、智能金融、智能医疗、智能电网等 8 部分内容。

课程类别	实验资源
Python 基础	编程题 800 多道，选择题、判断题和填空题 200 多道，覆盖 Python 教学所有知识点。每道题目标注了难度、知识点以及测试数据，每个编程题带有正确的代码样例。所有题目均支持自动评测。
Python 实训	12 个 Python 综合项目实训案例，覆盖的领域包括但不限于：游戏开发、正则表达式、信息检索、Web 应用、数据分析、爬虫、数据可视化、数据库应用等领域，涵盖 Python 教学全部知识点。
数学基础	<p>多元统计： numpy+scipy+matplotlib+sklearn 入门；聚类分析；判别分析；主成分分析；因子分析；对应分析；典型相关分析； Logistic 回归； 路径分析； 上市公司经营业绩指标选择与综合评价</p> <p>数值分析： 插值法； 矩阵的 LU 分解及在解线性方程组中的应用； 矩阵的特征值及特征向量； 线性方程组的迭代求解法； 非线性方程求根； 最小二乘法； 数值积分法； 常微分方程的数值解法； 数值微分； 函数逼近与曲线拟合。</p> <p>机器学习数学： 线性回归数学原理； 决策树数学原理； 朴素贝叶斯数学原理； 逻辑回归数学原理； 支持向量机的数学原理； AdaBoost 的数学原理； K 均值聚类的数学原理； K 近邻数学原理； SVD 数学原理； PCA 数学原理； 协同过滤数学原理； 树回归数学原理。</p> <p>深度学习数学： 随机变量； 概率分布； 边缘概率和条件概率； 最大似然估计； 无偏估计； 条件概率的链式法</p>

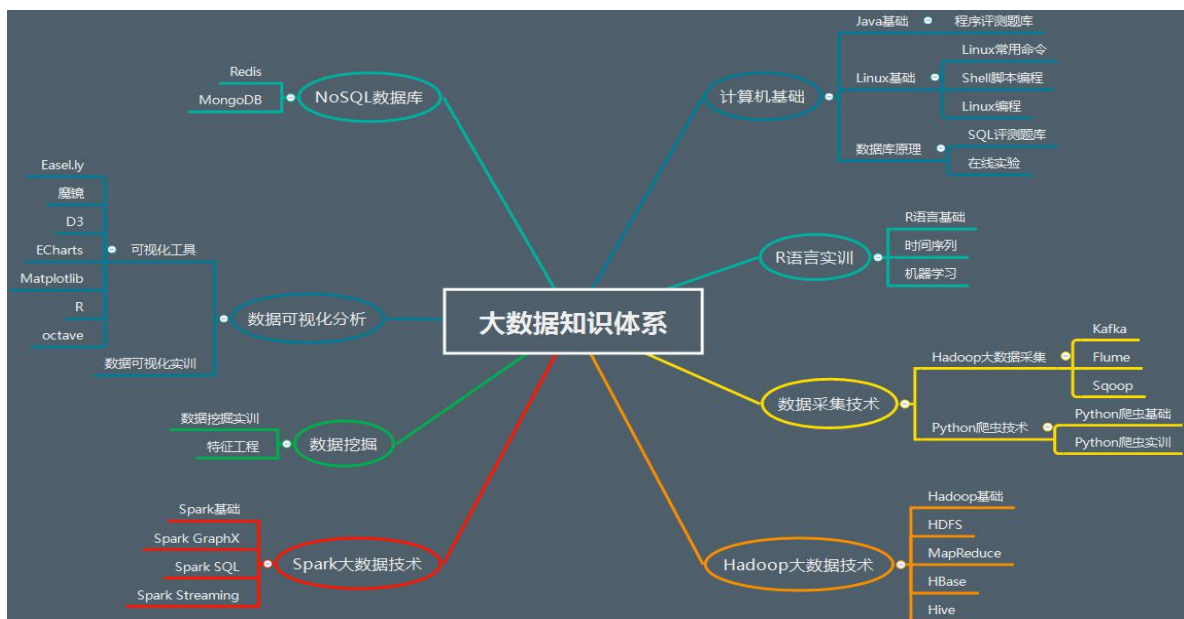
	<p>则；独立性和条件独立性；期望，方差，协方差；常用概率分布函数的数学性质；基于梯度的优化方法；约束优化原理；信息论基本数学原理；最大熵数学原理；条件随机场数学原理；梯度下降优化算法；噪声鲁棒性数学方法；稀疏表示的数学原理；Bagging 方法的数学原理；动量优化算法；自适应学习率算法；二阶近似方法；神经网络参数初始化方法；蒙特卡洛方法；结构化概率模型；近似推断；常用激活函数的数学性质；常见优化策略及原理。</p>
机器学习	<p>监督学习：kNN 算法；基于 kNN 的手写字识别；基于 kNN 的约会网站配对效果改进；基于 kNN 的乳腺癌诊断；基于 kNN 的 IRIS 分类；决策树算法；基于决策树的隐形眼镜选择；朴素贝叶斯算法；基于朴素贝叶斯算法的言论过滤器；基于朴素贝叶斯算法的垃圾邮件过滤器；基于朴素贝叶斯算法的新浪新闻分类；逻辑回归算法；基于逻辑回归的病马死亡率预测；支持向量机算法（SVM）；基于支持向量机的手写字识别；AdaBoost 算法；一元线性回归算法；多元线性回归算法；多项式回归算法；基于线性回归预测鲍鱼年龄；基于线性回归预测乐高玩具套装价格；树回归算法。</p> <p>无监督学习：K 均值聚类算法；使用 K 均值聚类对地图上的点进行聚类；Apriori 算法；基于 Apriori 算法的投票模式挖掘；基于 Apriori 算法发现毒蘑菇相似特征；FP-Growth 算法；基于 FP-Growth 算法 Twiter 数据挖掘；基于 FP-Growth 算法新闻网站点击流挖掘。</p> <p>数据降维：PCA 算法；基于 PCA 算法的半导体制造数据降维；SVD 算法；基于 SVD 的图像压缩。</p> <p>推荐系统：协同过滤算法；基于协同过滤的推荐引擎；基于 SVD 的餐馆推荐引擎。</p> <p>特征工程：利用机器学习做缺失数据补全；R 语言特征工程实践；基于 TF-IDF 的文本特征提取；基于 IRIS 数据集的特征工程实战。</p>
深度学习	<p>神经网络基础：神经网络游乐场；深度神经网络架构设计；深度前馈神经网络原理及实践；深度神经网络中反向传播的原理及实践；基于梯度的神经网络学习原理及实践；使用神经网络实现 XOR（异或）；深度学习中的正则化原理及实践；深度学习中的数据集增强原理及实践；深度学习中提高泛化能力的方法及实践；深度学习中避免过拟合的方法及实践；深度学习中对抗训练的原理及实践；深度学习中损失函数的选择及实践；深度学习中优化方法的选择及实践。</p> <p>卷积神经网络基础：卷积神经网络的原理、架构及实践；卷积神经网络的神经科学基础。</p> <p>循环神经网络基础：循环神经网络（RNN）原理、架构及实践；双向循环神经网络的原理、架构及实践；深度循环神经网络的原理、架构及实践；递归神经网络的原理、架构及实践；序列到序列（seq2seq）的模型原理与实践；长短期记忆网络（LSTM）的原理、架构及实践；循环神经网络中长期依赖问题的解决及实践；GRU 神经网络的原理、架构及实践。</p> <p>深度生成模型：玻尔兹曼神经网络原理、架构及实践；卷积玻尔兹曼神经网络的原理、架构及实践。</p>

	<p>深度学习实战：自己动手实现人工神经网络；自己动手实现卷积神经网络；自己动手实现循环神经网络；基于自实现的人工神经网络进行手写数字识别；使用卷积神经网络进行手写数字识别；使用循环神经网络进行手写数字识别；使用神经网络进行泰坦尼克号旅客生存率预测；基于深度学习模型的广告 CTR 预测。</p>
计算机视觉	<p>计算机视觉基础：OpenCV 图像基础操作；OpenCV 图像算数操作；OpenCV 颜色空间转换；OpenCV 几何变换；OpenCV 图像二值化；OpenCV 图像平滑；OpenCV 图像形态学变化；OpenCV 边缘检测、轮廓处理；OpenCV 图像变换(快速傅里叶变换)；TensorFlow 图像处理基本操作；TensorFlow 图像色彩调整；TensorFlow 标注框。</p> <p>基于深度学习的计算机视觉：基于 CNN 的车牌号码识别系统；基于 CNN 的 Discuz 论坛验证码识别系统；FaceNet 人脸识别；CIFAR-10 图像物体识别；基于 VGGNet 的 Oxford Flowers 17 分类任务；Resnet 模型及实战；LeNet 模型及实战；AlexNet 模型及实战；GoogleLeNet 模型及实战；Network in NetWork 模型及实战；RNN Pixels 模型及实战；Highway Network 模型及实战；Highway Convolutional Network 模型及实战；Faster RCNN 模型及实战；SSD 模型及实战；Yolo 模型及实战；FCN 模型及实战；CapsNet 模型模型及实战；图像目标检测实战；图像语义分割实战；基于卷积神经网络的图像风格迁移；基于 LSTM 的图像理解；基于自编码器的手写数字识别。</p>
自然语言处理	<p>自然语言处理基础：中文分词；最大熵原理；TF-IDF 模型；信息检索原理与实践；词表达方式；词向量计算；短文本聚类；问句的意图识别；文本相似度计算；语法分析与词法分析；seq2seq 模型原理与实践。</p> <p>自然语言处理实战：自动问答系统的完整设计；基于检索系统的聊天机器人；word2vec 模型实战；词嵌入模型实战；基于 seq2seq 实现聊天机器人；基于 LSTM 到 IMDB 情感数据集分类任务；基于双向 LSTM 应用到 IMDB 情感数据集分类任务；基于动态 LSTM 从 IMDB 数据集分类可变长度文本；基于 LSTM 网络生成新的美国城市名；基于 LSTM 网络生成新的莎士比亚手稿；基于卷积网络从 IMDB 情感数据集中分类词序列；中英文机器翻译。</p>
智能金融	<p>股价预测：基于一元线性回归的股价预测；基于多元线性回归的股价预测；基于强化学习策略的股价预测；基于 LSTM 的股价预测。</p> <p>综合实训：面向金融领域的智能客服（聊天机器人）；银行客户流失预警；信用卡账户违约预测；行为评分卡模型；银行信贷客户聚类分析。</p>
语音识别	<p>语音识别基础：语音降噪原理及实践；语音增益原理及实践；语音特征提取原理及实践；语音滤波器组和特征变换；语音解码原理及实践；HMM 声学模型原理及实践；DNN 声学模型原理及实践；CNN 声学模型原理及实践；RNN 声学模型原理及实践。</p> <p>语音识别实战：基于 Kaldi 框架的语音识别；基于 CMU Sphinx 框架的语音识别；基于 RNN 的语音识别系</p>

	<p>统-简单数字；支持常见英文单词识别的语音识别系统；基于深度学习的音乐流派分类系统；基于深度学习的音乐推荐系统。</p>
TensorFlow 编程	<p>基础知识：神经网络游乐场；TensorFlow 基础：张量、计算图、会话。</p> <p>基本模型：基于 TensorFlow 的最近邻算法；基于 TensorFlow 的线性回归算法；基于 TensorFlow 的逻辑回归算法；基于 TensorFlow 的随机森林算法；基于 TensorFlow 的 K 均值聚类算法；基于 TensorFlow 的梯度 Boost 决策树算法；基于 TensorFlow 的 word2vec 算法。</p> <p>深度神经网络模型：基于 TensorFlow 的多层感知器；基于 TensorFlow 的神经网络；基于 TensorFlow 的卷积神经网络；基于 TensorFlow 的循环神经网络；基于 TensorFlow 的双向循环神经网络；基于 TensorFlow 的动态循环神经网络；基于 TensorFlow 的自编码器；基于 TensorFlow 的 GAN；基于 TensorFlow 的 DCGAN。</p> <p>实用技术：Tensorflow 持久化：保存和恢复模型；Tensorboard 基础：可视化学习；Tensorboard 进阶。</p> <p>数据管理：TensorFlow 数据集接口；在 TensorFlow 中创建图像数据集。</p> <p>GPU：在 TensorFlow 中使用 GPU 的基础知识；在 TensorFlow 中基于多个 GPU 进行训练。</p>
Keras 编程	<p>Keras 基础：Keras 快速入门实践；Keras 常用接口实战；Keras 回调函数的使用。</p> <p>Keras 实战：在 Keras 中配合使用 TensorBoard 实现可视化；在 Keras 中实现手写数字识别；在 Keras 中实现 LeNet 模型；在 Keras 中进行 CIFAR-10 图像分类；在 Keras 中实现 VGG-16 模型；在 Keras 中实现 inception-v3 模型；在 Keras 中实现基于 GAN 的手写数字生成；在 Keras 中实现基于 GAN 的 CIFAR-10 图像生成；在 Keras 中实现 word2vec 模型；在 Keras 中实现词嵌入模型；在 Keras 中实现基于 RNN 的文本生成；在 Keras 中实现基于 LSTM 的情感分析；在 Keras 中实现基于 GRU 的 POS 标注；在 Keras 中实现基于 LSTM 的电量消耗预测；在 Keras 中实现基于递归神经网络预测空气质量；在 Keras 中实现基于记忆网络的智能问答系统；在 Keras 中实现 Google 的 Deep Dream 模型；在 Keras 中实现图像风格迁移模型。</p>
生成式对抗网络	<p>生成式对抗网络基础：GAN 的原理及架构；DCGAN 的原理及架构；WGAN 的原理及架构；WGAN-GP 的原理及架构；LSGAN-BEGAN 的原理及架构；paGAN 的原理及架构。</p> <p>生成式对抗网络实战：基于 GAN 生成二次函数曲线；基于 GAN 生成正太分布曲线；基于 GAN 生成手写数字；基于 GAN 人脸生成；基于 GAN 的教堂建筑生成；基于 GAN 的餐厅、厨房、会议室生成；基于 GAN 的 Deep Dream 模型；基于 GAN 的仿生人；基于 GAN 实现根据文本生成图像；基于 GAN 的面部表情变换；基于 GAN 的图像到图像；基于 GAN 实现草图上色；基于 GAN 的视频生成；基于 GAN 实现图片马赛克去除；基于 GAN 的动漫头像生成；基于 GAN 的音乐合成。</p>
时间序列	<p>时间序列基础：时间序列原理；时间序列常用模型。</p> <p>时间序列实战：基于 AIRIMA 模型预测湖北省 GDP；基于 Pandas 的时间序列分析；R 语言时间序列分析实</p>

	战；基于时间序列分析的正弦曲线预测；基于 LSTM 的时间序列预测最高股价；基于 LSTM 的时间序列预测气候；基于 LSTM 的国际航班乘客数量预测；基于 LSTM 预测彩票。
自编码器	自编码器基础： 自编码器原理、架构及实践；稀疏自编码器的原理、架构及实践；降噪自编码器的原理、架构及实践；正则自编码器原理、架构及实践；收缩自编码器原理、架构及实践；卷积自编码器原理、架构及实践；堆叠自编码器的原理、架构及实践；变分自编码器的原理、架构及实践。 自编码器实战： 使用自编码器学习流形；基于深度自编码器的推荐引擎；基于深度自编码器的图像数据压缩；基于自编码器的手写数字生成。
自动驾驶	自动驾驶基础： 自动驾驶基本原理；自动驾驶常用模型。 自动驾驶实战： Udacity 的自动驾驶汽车模拟器入门；Udacity 的三路摄像头数据采集；Udacity 采集数据的增强和其他预处理；自动驾驶神经网络模型搭建及训练；使用训练的自动驾驶模型控制 Udacity 模拟器。
智能游戏	基于深度强化学习自动玩转 FlappyBird 游戏；基于深度强化学习自动玩转 Atari 游戏；基于深度强化学习自动玩转 Chrome 浏览器的小恐龙游戏；AlphaGo 原理解析；基于深度强化学习自动玩转跳一跳小游戏；基于深度强化学习自动玩转抓球游戏。
智能医疗	基于神经网络的肺癌预测；基于卷积神经网络的大脑病变分割；基于光谱卷积网络的特征图分析；面向医疗图像的基于 GPU 加速的运动补偿策略；基于卷积神经网络的实时胎儿标准扫描平面检测。
智能电网	基于 LSTM 的电力负荷预测；电力窃漏电用户自动识别；基于卷积神经网络的电力设备图像识别；基于深度学习算法的电网故障诊断方法；基于深度学习的电力电子装置在智能电网中的适应性分析与设计。

数据科学专业实验体系



首先制定了大数据知识体系图谱，包含计算机基础（Java 基础、Linux 基础、数据库原理）、R 语言实训、数据采集技术、Hadoop 大数据技术、Spark 大数据技术、数据挖掘、数据可视化分析、NoSQL 数据库等共计 10 部分。根据该图谱设计了一系列的核心实验，实验的数量不在于多，而在于能够覆盖大数据知识体系的内容，且前后有一定的逻辑关系，学生按照该实验路线学习后，可基本掌握大数据的核心知识。实验资源涵盖了数据采集、存储、处理、分析的相关基本知识。同时，还设置了一系列大数据综合项目实训案例，通过这些实训案例让学生综合运用从课程中学习到的大数据知识对源自真实世界中的大数据集进行分析、处理并提炼出有效的结论。

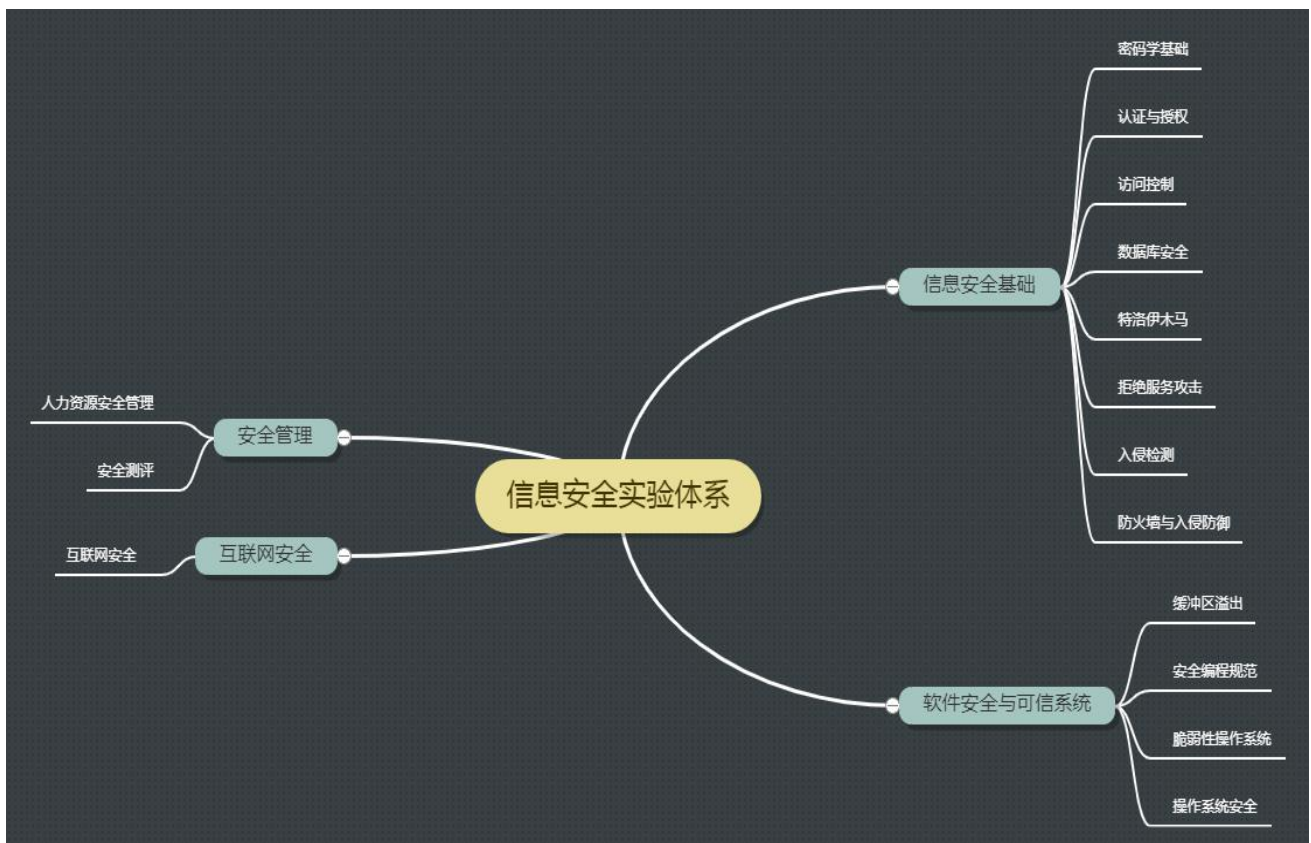
课程类别	实验类别
Java 基础	Java 语言程序题库不少于 1000 道。每道题目标注难度和知识点，并带有代码样例。包含填空、选择、编程三种题型，分为作业题库和考试题库两大类，所有题目支持自动评测。
Linux 系统及应用	操作系统模拟实验： 进程管理；处理机调度（进程调度）；分区存储管理；段式存储管理；请求页式存储管理；设备管理；文件管理。 操作系统应用实验： Bash 脚本编程实验；Linux 常用命令实验；
R 语言实训	R 语言基础： 常用命令；变量的基本操作；正则表达式；循环与条件；向量化操作；用 stringr 包处理字符串；读取、输出及操作；数组与数据框操作；merge 与 subset 的使用；创建数据集；文本分析；数据框 dataframe 的使用；设置镜像；ggplot 基础性操作；ggplot2 柱状堆叠图；日期与时间格式；逻辑操作；数据处理包 dplyr；因子和有序因子。 时间序列： 时间序列函数；时间序列的创建及时间序列模型。 机器学习： 多元线性回归；聚类分析；集合运算；k 近邻分类；贝叶斯分类；决策树分类。
数据库原理	数百道常用的 SQL 数据操作和定义语言题库，具体包括数据定义语言，数据基本查询，数据高级查询，数据更新语言，每道题目支持自动评测。 基于虚拟桌面的数据库实验环境，实验资源覆盖安全性、完整性、数据恢复、并发控制、存储过程、数据库监控与性能优化、视图与索引、数据库应用开发、数据库设计等实验。
NoSQL 数据库	Redis 基本操作： 表创建、增、删、改、查等； Redis 编程： 表的创建/删除/查看；表内数据的增删改查；订阅/发布；持久化。 MongoDB 基本操作： 集合创建、增、删、改、查等； MongoDB 编程： 集合的创建/删除/查看；集合内部数据的增删改查；聚合。
数据采集技术	Hadoop 大数据采集： 接收 AvroSource 的信息；文件数据 Flume 至 HDFS；Kafka 订阅/推送；与 MySQL 对接。 Python 爬虫： Python 爬虫基础；Python 抓取网页获取字幕语料；Python 抓取微信群聊天语料。
数据可视化分析	Easel.ly 使用方法；魔镜的使用方法；D3 可视化库使用方法；ECharts 的使用方法；Matplotlib 基本使用方法；Matplotlib 绘图与可视化进阶；R 语言数据可视化分析；octave 使用方法；二元分类效果的图像展示；鸢尾花数据可视化；标签云。
Hadoop 大数据技术	Hadoop 基础： Hadoop 基本操作：启动、停止、日志分析、Web 端查看；分析 Hadoop 配置文件中各参数的含义。 HDFS： HDFS 基本操作；目录编程接口；文件编程接口；其他编程接口；设计并实现一个 HDFS Shell。 MapReduce： WordCount 源码透析；基数排序；祖孙辈关系；设计并实现基于 MapReduce 的索引倒排；分析基于 MapReduce 实现的 PageRank 算法。 HBase： HBase 基本操作：创建表、增删改查、退出；HBase 编程：表的创建、删除、遍历；HBase 编程：增、删、改、查、Scan、过滤器；设计并实现一个 HBase Shell。

	Hive: Hive 的基本操作; Hive 应用实例: 词频统计。
Spark 大数据技术	Spark 基础: Spark Shell 的启动、读取文件、词频统计; RDD 的分区、依赖、计算、创建、执行、转换等; 基于 Spark 实现词频统计; RDD 的创建、执行、转换、持久化、存储; 基于 Java 语言的 Spark 独立应用程序。 Spark GraphX: 创建、结构、转换、聚合等 Spark SQL: DataFrame 创建、保存、常见操作; 学生信息管理系统综合案例 (整合 Hive) Spark Streaming: DStream 基本操作; 数据源; 编程。
数据挖掘	数据挖掘实训: 电力窃漏电用户自动识别; 航空公司客户价值分析; 中医证型关联规则挖掘; 基于水色图像的水质评价; 家用电器用户行为分析与事件识别; 应用系统负载分析与磁盘容量预测; 电子商务网站用户行为分析及服务推荐; 财政收入影响因素分析及预测; 基于基站定位数据的商圈分析; 电商产品评论数据情感分析。

信息安全专业实验体系

信息安全学科可分为狭义安全与广义安全两个层次, 狭义的安全是建立在以密码论为基础的计算机安全领域, 早期中国信息安全专业通常以此为基准, 辅以计算机技术、通信网络技术与编程等方面的内容; 广义的信息安全是一门综合性学科, 从传统的计算机安全到信息安全, 不但是名称的变更也是对安全发展的延伸, 安全不再是单纯的技术问题, 而是将管理、技术、法律等问题相结合的产物。

信息安全学科包括的范围很大, 其中包括机密泄露和完整性破坏的防范、网络环境下的信息安全、计算机操作系统安全、各种安全协议、安全机制 (数字签名、消息认证、数据加密等), 直至安全系统, 如 UniNAC、DLP 等。



CG 信息安全实验内容设计兼顾信息安全学科覆盖的全面性和学生实际操作能力的训练，把实验分为四个大的子类以及 14 个课程和具体实验，如上图所示，实验内容处于不断丰富、完善和与时俱进的提高中。

下表是 CG 信息安全实验内容已经具备的 14 个课程和 81 个具体实验。每个具体的实验包含以下内容：

- 实验介绍：本实验涉及的信息安全领域/技术、本实验要达到的目的，本实验的知识点、本实验的实验环境。
- 实验原理：本实验涉及的信息安全原理。
- 实验准备：开始实验前，学生必须完成的准备工作。
- 实验步骤：学生分步完成的实验过程，以及最后的评分过程。

课程类别	实验类别
密码学基础	对称加密算法（3DES AES 等）
	非对称加密算法（RSA 等）
	哈希算法（SHA1 等）
	SSL/TLS（证书的创建，验证，以及通讯等，编写 SSL 程序，配置 nginx 使用 HTTPS 等）
	国密算法（对称、非对称算法等）
	国密算法证书以及国密 SSL
	SSL/TLS VPN 的配置
	使用 FreeSwan 配置 IPSEC VPN
	配置 L2TP VPN
认证与授权	用 C 程序实现用户名口令认证
	使用 OpenSSL 实现基于证书的认证
	使用 OpenSSL 实现证书双向认证
	基于网页滑块的认证实现
	Linux 下基于 PAM 模块的认证
	基于 Kerberos 的认证与授权
	Radius 认证与授权
访问控制	传统的操作系统访问控制
	基于访问控制链表(ACL)的访问控制
	SELinux 的配置
	Linux LSM 模块开发
	利用钩子实现 Windows 访问控制
	基于 Linux CGROUPS 的资源配额管理
	基于 Docker, LXC 等容器技术的强资源隔离
数据库安全	利用 metasploit 侵入 Postgres 数据库
	SQL 注入攻击
	利用 metasploit 侵入 MySQL 数据库
	数据库中基于角色的访问控制
	通过数据库日志对攻击进行审计
	数据库备份数据的加密保护

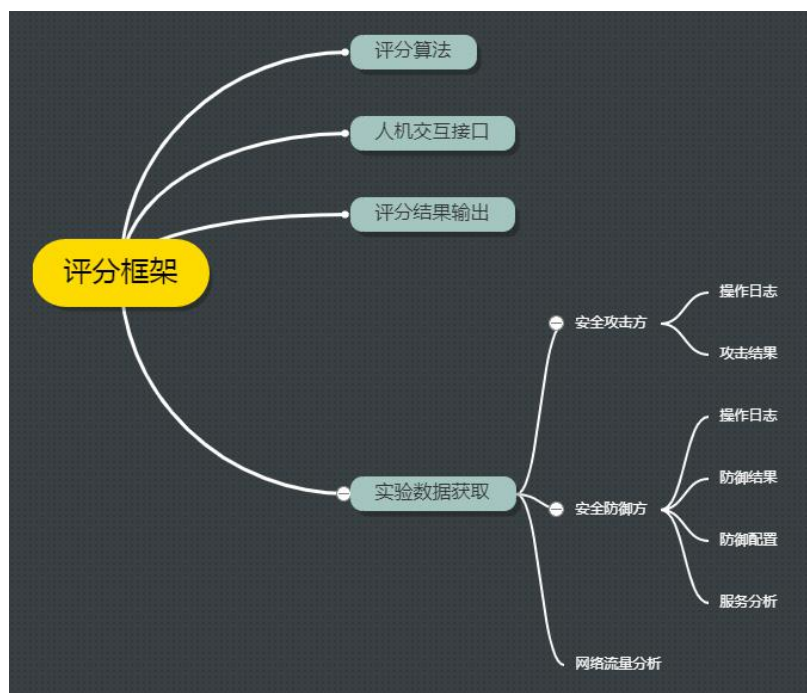
特洛伊木马	在 Windows 下通过社会工程的办法植入特洛伊木马
	在 Linux 下通过 metasploit 植入特洛伊木马
	通过特洛伊木马对主机进行访问控制
	主机上对特洛伊木马进行检测与移除
	构造并在主机上植入计算机病毒
	在主机上检测并移除计算机病毒
拒绝服务攻击	Syn 泛洪攻击代码编写与运行
	构造 4 层-7 层拒绝服务攻击
	构造 NTP 反射攻击
	构造 ICMP 洪水攻击
	DNS 服务器放大攻击
入侵检测	基于主机的入侵检测安装配置与检测
	基于网络的入侵检测安装配置与检测
	蜜罐的安装配置
	libpcap, libwireshark 等网络入侵检测库的安装使用
防火墙与入侵防御	Linux 网络防火墙 iptables 和 DMZ
	入侵防御系统的配置
	Linux netfilter 编程
	Linux 网桥和 ebtables 防火墙
	ARP 欺骗和保护
缓冲区溢出	shell code 的编写与缓冲区溢出
	堆溢出实验
	缓冲区溢出静态检查
	缓冲区溢出动态检查
安全编程规范	通过输入大小限制防范缓冲区溢出
	格式化字符串的使用
	整数变量数值溢出的处理
	指针的安全使用
	线程锁和进程锁的安全使用
	智能代码重用
	针对危险函数的攻击
	防御性编程与进攻性编程
	软件后门的利用与防范
脆弱性操作系统	vulnhub 系列操作系统的分析与攻击
	HNIX 操作系统的分析与攻击
操作系统安全	通过配置, 排查, 移除服务来加固操作系统
	操作系统审计系统的安装, 审计结果分析
	日志函数的使用
	操作系统的备份与存档
	操作系统安全测试
	操作系统补丁的制备, 管理与安装
人力资源安全管理	计算机系统用户的管理和权限分配
	局域网用户邮件与网络监控

安全测评	网络安全测评
	主机安全测评
互联网安全	安全 E-mail 和 S/MIME
	公开密钥体系结构与 X.509
	Web 脆弱性分析
	跨站攻击
	Tomcat 漏洞攻击
	CSRF 跨站请求伪造攻击
	DNS 假冒攻击
	网站加固

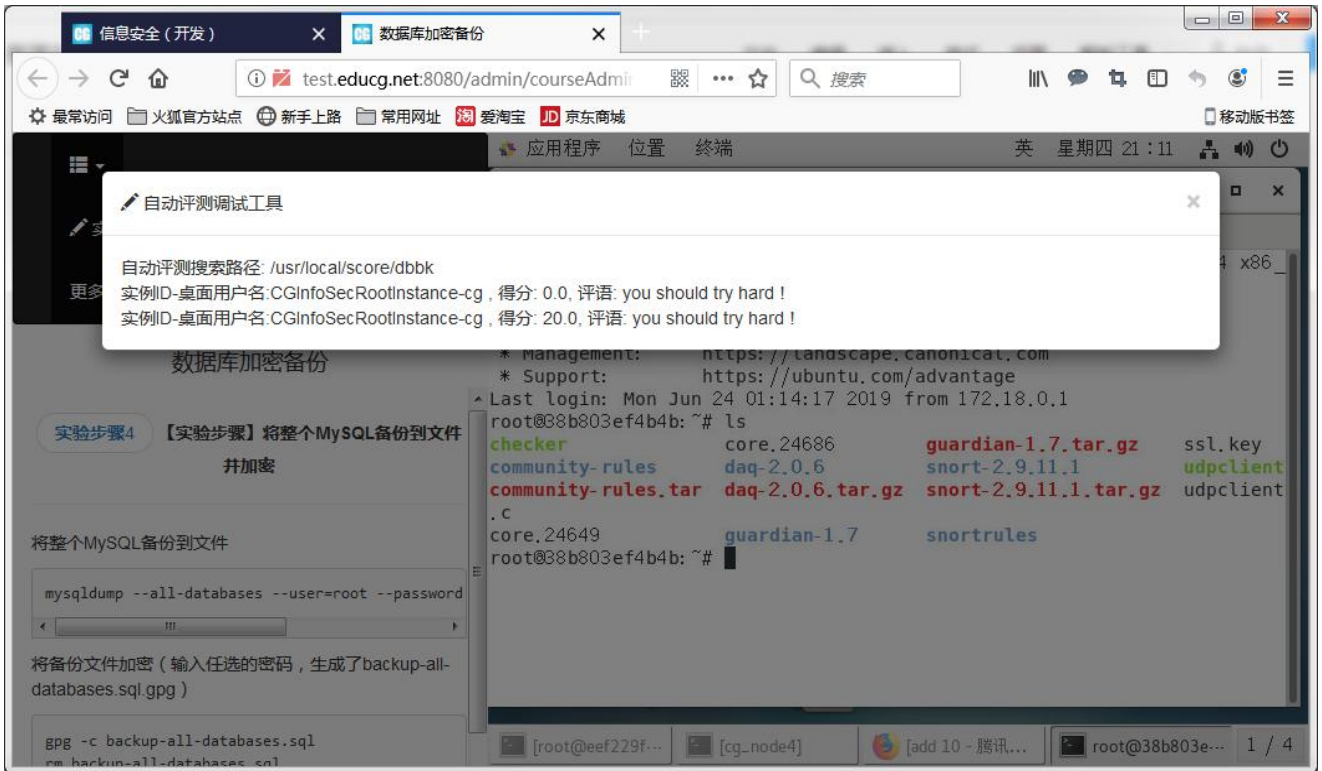
信息安全实验过程繁琐，涉及知识点多。由教师或助教进行人工评分耗时耗力，而且难免出错。CG 信息安全实验自动化评分系统采用多种手段进行自动化评分。CG 信息安全自动化评分系统具有如下特点：

- 多样性的评分手段。根据行为日志、攻击结果分析，服务可靠性变化情况，尽可能自动化地对实验参与者进行评分，以减轻教师负担，提高评价可靠性。评分自动化的技术手段包括并不限于：文件内容比对、服务可靠性分析、日志输出分析、以及网络包分析等。
- 最大幅度减少抄袭等作弊行为。对系统的非实验性后门，严格把关，防止参与者走捷径作弊。对同一门课程的实验，输入、输出和算法设计一定的变化量，避免作业提交内容相同。
- 科学合理的评分区间。按照实验的难易，学生实验的完成程度，学生对知识点的掌握程度，设计合理的评分区间。

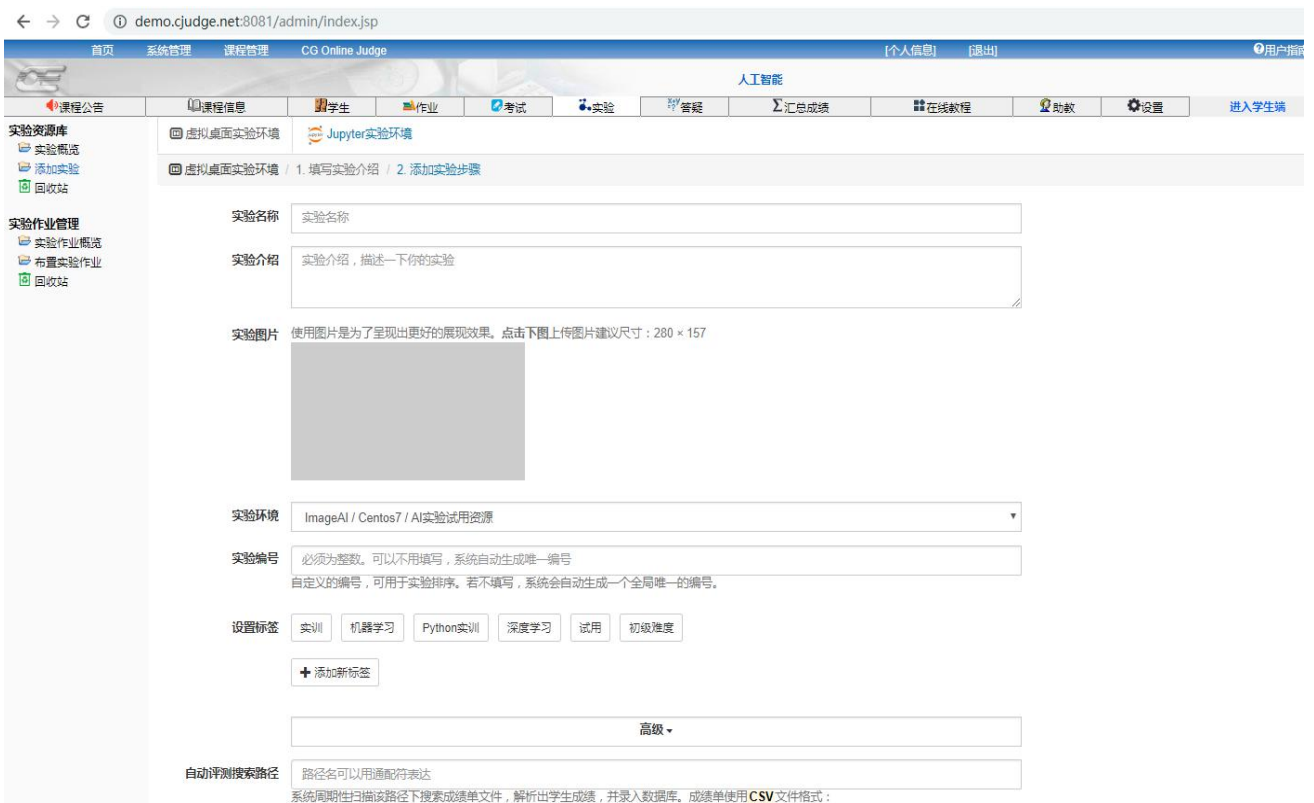
系统针对每个具体实验开发一个评分程序，该程序从安全攻击方和安全防御方以及网络转发模块获取多种的实验数据，采用加权平均等算法对学生的实验结果进行评价。下图是 CG 信息安全自动化评分系统结构：



下图是数据库加密与备份实验的调试性输出结果。从图中可以看出，学生如果只完成了部分实验，系统将根据其完成情况给出对应的评分。



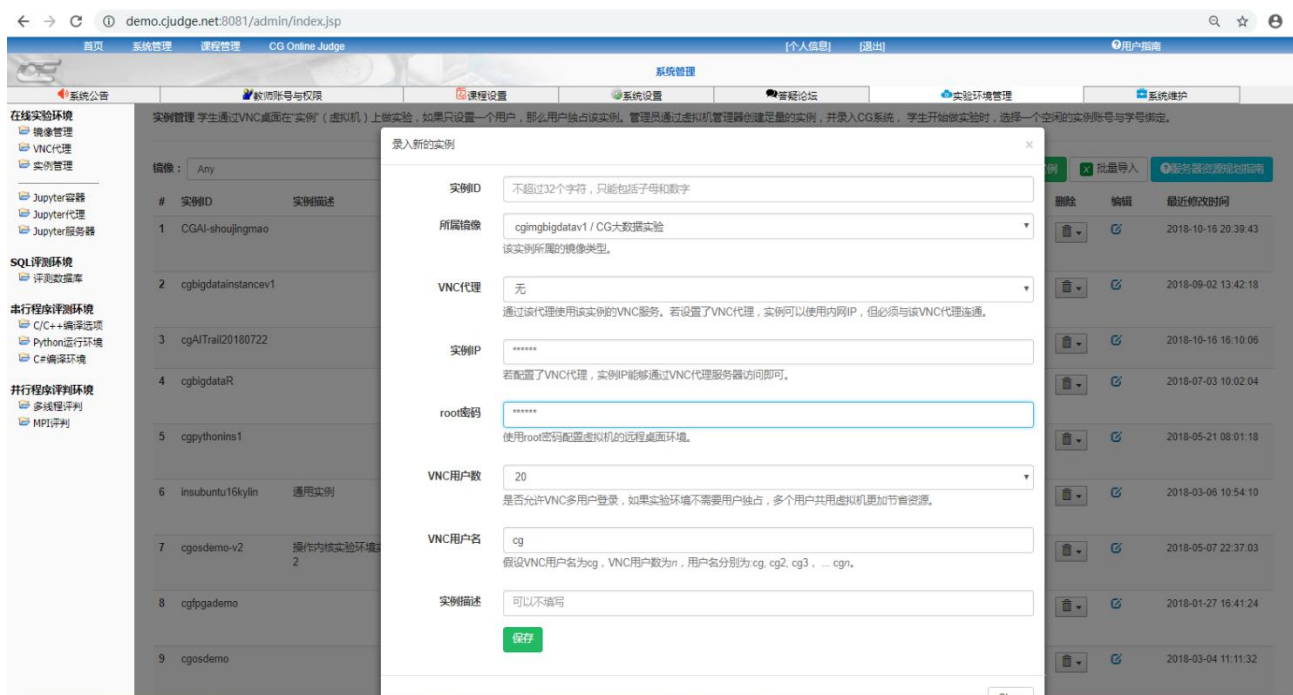
构建自己的实验体系



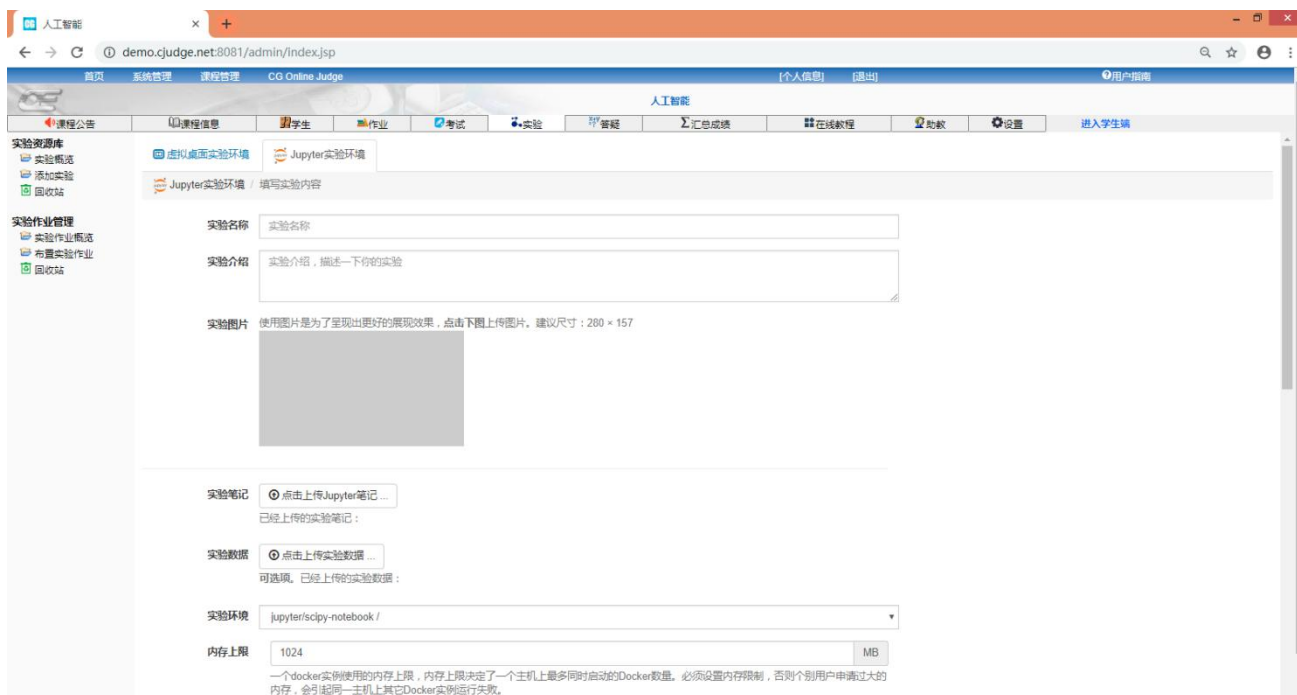
虽然 CG 平台提供了覆盖各专业多门课程和多个应用领域的实验资源，且会不断更新实验资源库，并将更新的实验资源库提供给各高校。但各专业的相关理论和应用，发展速度日新月异。CG 平台很难（也无法）提供能覆盖所有理论、所有模型、所有应用领域、所有框架和库的实验资源集。CG 平台更侧重于为各高校各专业的课程教师提供一个功能丰富、简单易用的教学实验平台，并在平台内提供多种便捷的实验资源构建功能，使各高校教师可在平台上轻松构建自己的实验体系。

在 CG 平台上添加虚拟桌面在线实验的方法如上图所示，教师可选择学生进行实验时的镜像环境。所选镜像可以是 CG 平台提供的镜像，也可以是教师自己制作的虚拟镜像。例如，如果需要制作基于 PaddlePaddle 的一系列深度学习实验资源并将这些实验用于教学，首先将每个实验需要的实验支撑材料，如实验脚本、实验数据集、实验代码等内容集成到已经安装了 PaddlePaddle 的虚拟机中。基于该虚拟机镜像构建多个虚拟机实例，然后将这些实例挂载到 CG 平台上。基于该实验镜像，创建步骤化的实验指导手册。在完成这些工作后，学生便可进入到教师构建的实验体系中进行实验。如果教师为实验制作了自动评测脚本，可将保存评测分数的文件路径录入到 CG 平台上，CG 平台将自动读取评测成绩，并将成绩更新到学生的实验得分中。

在 CG 平台上，录入虚拟机镜像时，教师可选择镜像的类型（Vmware, KVM, Docker 等）、操作系统以及适用课程范围；录入虚拟机实例的方法如下图所示，对于每个虚拟机实例，只需提供该虚拟机的 IP 地址、root 密码，CG 平台便可将该虚拟机的桌面“投射”到浏览器上，供学生做实验使用。因此，可无缝对接各类虚拟化环境和云环境。只要是 Linux 系统，就都可以“投射”，甚至可以是安装在 PC 机或笔记本上的 Linux 系统。在录入实例时，可选择每个实例支持的学生数量，数量范围从 1 到 100，由教师自己选择。



在 CG 平台上，录入 Jupyter 实验的界面如下图所示，录入实验资源时，可直接上传 ipynb 格式的文档和数据集，然后选择 Jupyter 的 Docker 版本，教师还可设置学生的实验运行环境所占用的内存大小。由于 Jupyter 笔记具备良好的交互性和资源隔离措施，可支持多媒体和 LaTeX，同时对网络流量的要求较低，鼓励教师基于 Jupyter 笔记制作实验资源。



6 开放可扩展

实验环境可扩展

- 虚拟桌面可扩展，支持教师接入自己制作的虚拟机，只需提供 IP、root 密码、并发人数即可接入。
- Jupyter 笔记可扩展，支持教师上传自己制作的 Jupyter 笔记 Docker 镜像。
- 通用评测框架可扩展，支持教师基于虚拟或者 Docker 制作评测机作为插件接入 CG。
- Docker 实验环境可扩展，支持教师上传自己制作的 Docker 镜像。

实验体系可扩展

- 教师可基于 CG 平台提供的或自定义的虚拟桌面镜像，制作实验指导手册。
- 教师可基于 CG 平台提供的或自定义的 Jupyter 笔记 Docker 镜像，导入或制作新的实验笔记文档。
- 教师可基于 CG 平台提供的或自定义的评测机，制作可自动化评测的复杂项目题。
- 教师可基于 CG 平台提供的或自定义的 Docker 镜像，制作新的实验文档。

硬件资源可扩展

- 平台及实验环境都运行于 Linux 系统应用层，不依赖于任何虚拟化平台，因此新服务器资源无论是直接

安装 Linux 系统还是安装虚拟化平台后，再创建 Linux 虚机，都可以直接接入 CG 平台进行接管。CG 平台的自动化部署脚本会对 Linux 系统进行初始化，构建为指定的实验环境。

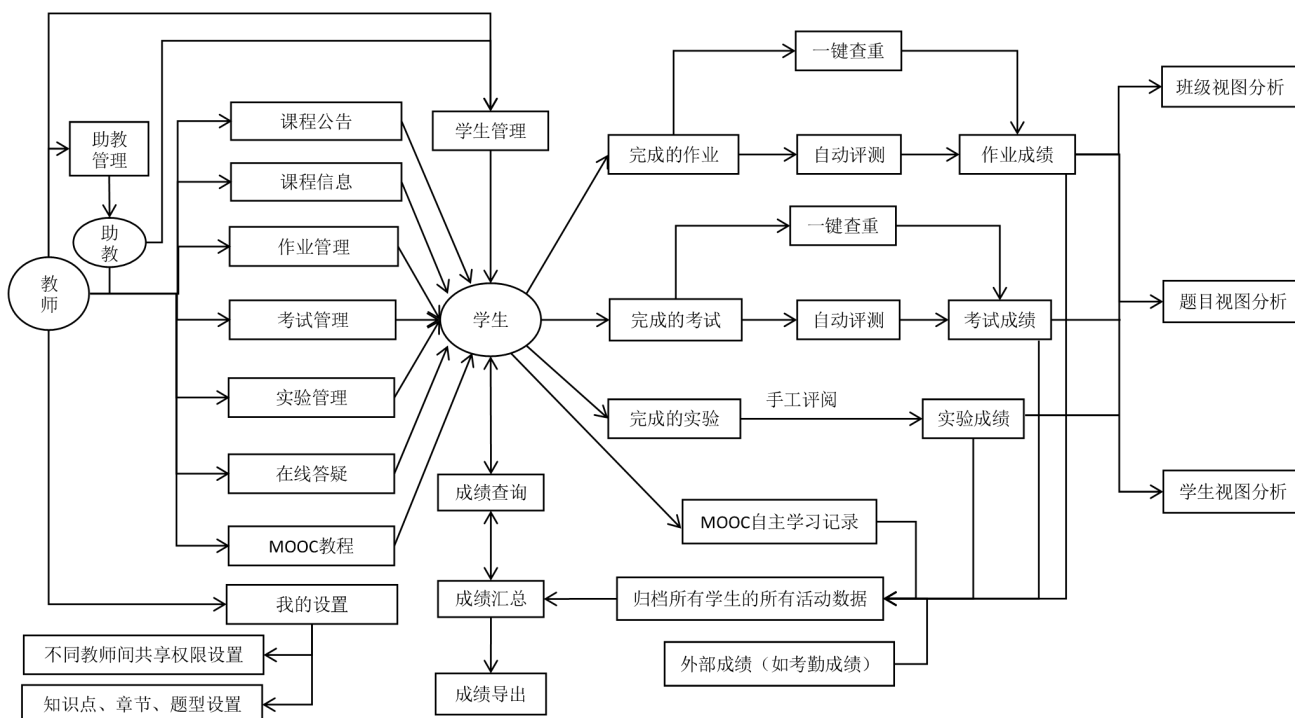
- 由于运行于 Linux 系统应用层，实现了软硬件松耦合，方便硬件独立扩充、升级、更新。在传统的一体机模式中，难以做到硬件的独立运维。

学生人数可扩展

CG 平台对并发人数不做限定，学生并发人数仅取决于硬件配置及数量。通过引入更多硬件服务器，即可支撑更高的并发实验人数。

7 契合教学过程

一站式教学支撑

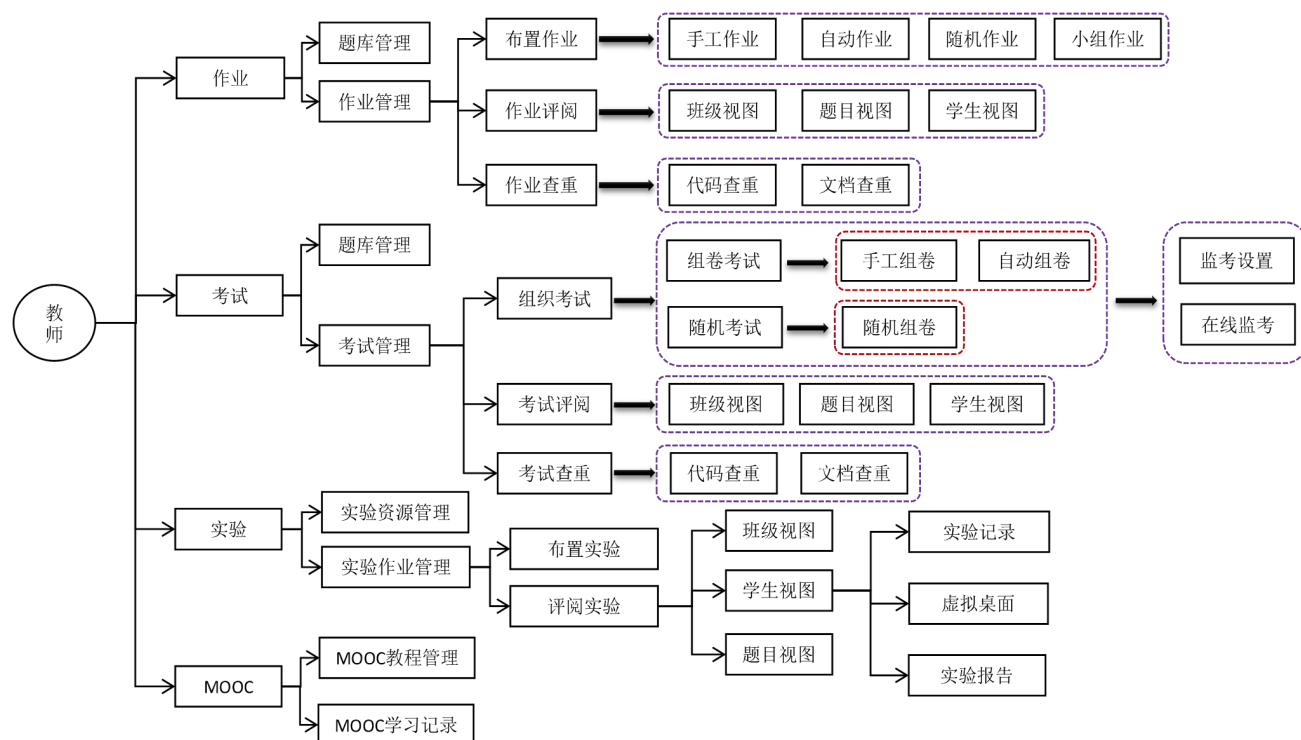


CG 平台内置完善课程管理系统为各专业各门课程地开展提供一站式全过程教学支持。CG 平台的课程管理在设计理念、功能设置方面集合了国内高校的教学经验与实际需求，极大减轻教师工作量，全方位支持在线考试。CG 平台具备强大的自动化运维功能，方便系统的日常维护，保障数据的可靠性与安全性。教学平台提供了基本的功能支撑和教学支撑，支撑各个专业各课程的教学与实践，实现在线教学实验资源的统一管理。

CG 平台的课程管理功能包括但不限于：公告管理，课程信息管理，学生管理，题库管理，在线作业，在线考试，作业成绩分析，考试成绩分析，成绩汇总，在线答疑，权限管理，助教管理，MOOC 资源管理

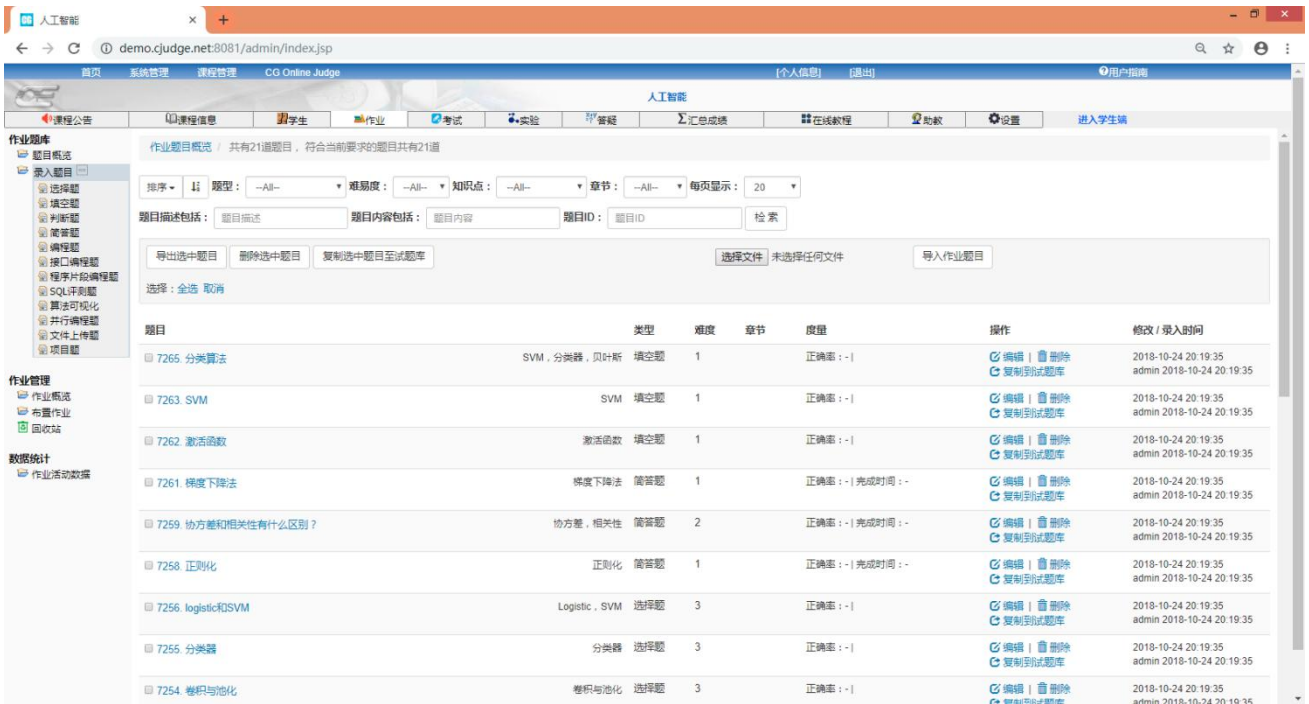
等功能。如上图所示，在 CG 平台上，教师可以发布课程公告和课程信息，给学生布置作业、考试、实验，在课程论坛中为学生在线答疑，发布 MOOC 教程让学生自主学习。学生的作业、考试可进行自动评测，所有提交的代码和文档可分别进行查重，部分实验支持自动评测。作业、考试、实验的成绩可以多种方式查看和分析，包括班级视图、题目视图以及学生视图。在课程结束后，可一键归档所有学生在课程中的活动数据，可对学生在不同教学环节（如作业、考试、实验、考勤、MOOC）的成绩进行加权汇总。学生可以查询最终汇总后的成绩。教师可设置助教辅助自己完成部分工作，减轻教师工作量。

作业、考试、实验、MOOC

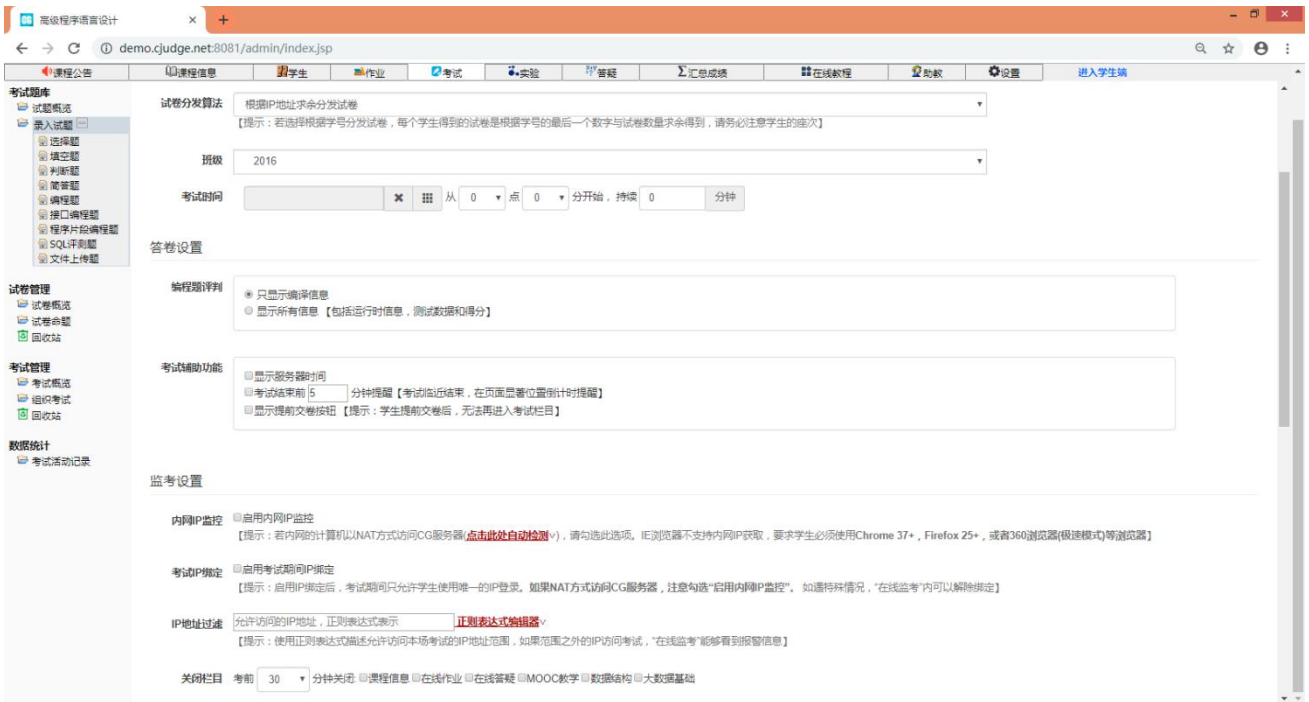


教师可分别在“作业”栏目和“考试”栏目中管理作业题库和考试题库。作业题库支持的题型包括选择题、填空题、判断题、简答题、编程题、接口编程题、程序片段题、SQL 评测题、算法可视化、组成原理题、并行编程题、文件上传题、项目题等 13 种题型。考试题库支持选择题、填空题、判断题、简答题、编程题、接口编程题、程序片段编程题、SQL 评测题、文件上传题等 9 种题型。其中，选择题、填空题、判断题、简答题、编程题、接口编程题、程序片段题、SQL 评测题、组成原理题、并行编程题皆支持自动评测。提供了多种布置作业的方式，包括手工选题、自动出题、随机作业、分组作业等 4 种方式。

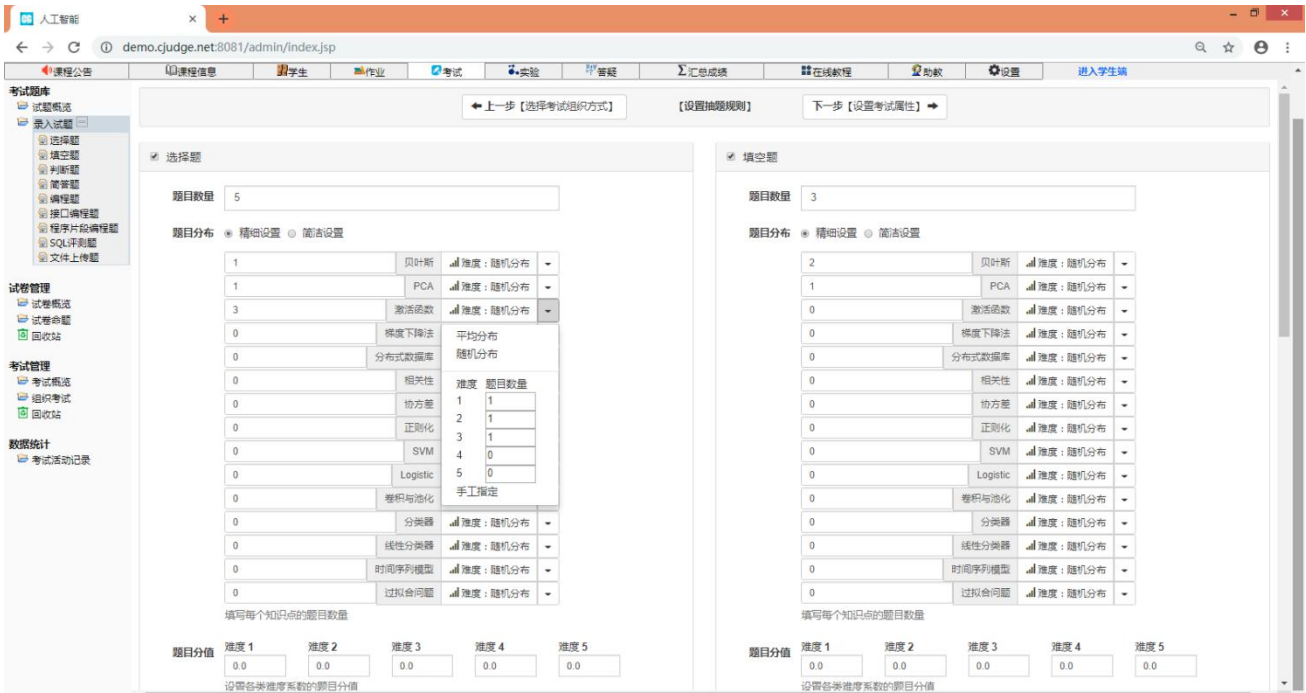
在 CG 平台上，教师进入作业栏目后的页面如下图所示。默认会显示当前已导入或教师自己创建的题库，教师可按题型、难易度、知识点、章节、题目描述、题目内容、题目 ID 筛选和搜索题目。在学生完成题目的过程中，系统自动对题目进行度量，度量指标包括提交率、正确率、完成时间、平均代码行、平均分数等内容。



CG 平台具备完善的在线考试功能：支持单场考试多套试卷、能够根据 IP 地址或学号分发试卷；支持 NAT 和反向代理特殊网络架构下的客户端真实 IP 侦测；支持按照知识点、章节、难易度、使用频度等规则，自动抽题组卷；具有监考功能，自动侦测学生考试中异常行为如异地计算机登录、交换账号登录等行为；支持 IP 访问控制，能够使用正则表达式，只允许特定范围的地址段访问考试；考试结束后，能够利用代码相似性检测功能，对提交代码进行相似性检查。组卷考试的设置界面如下图所示。



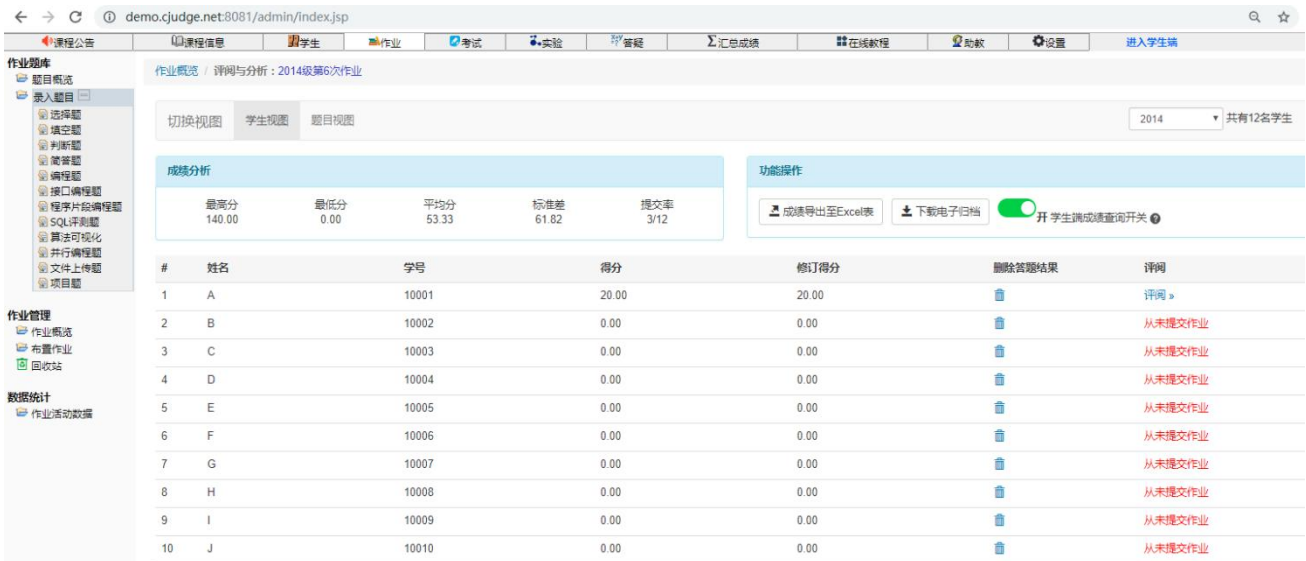
随机组卷考试的界面如下图所示，教师可指定组卷规则，包括每种题型的题目数量、题目分布、题目分值等信息。学生端进入考试后，将按照组卷规则随机组卷。



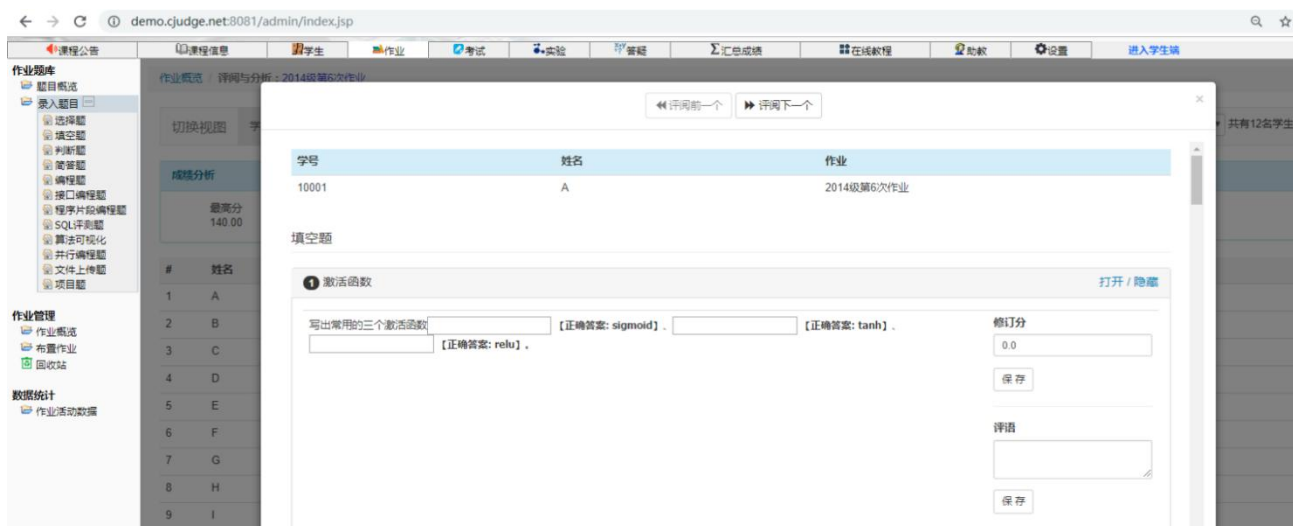
教师可添加 MOOC 视频、PPT、PDF 讲义，为学生构建自主学习环境，如下图所示。



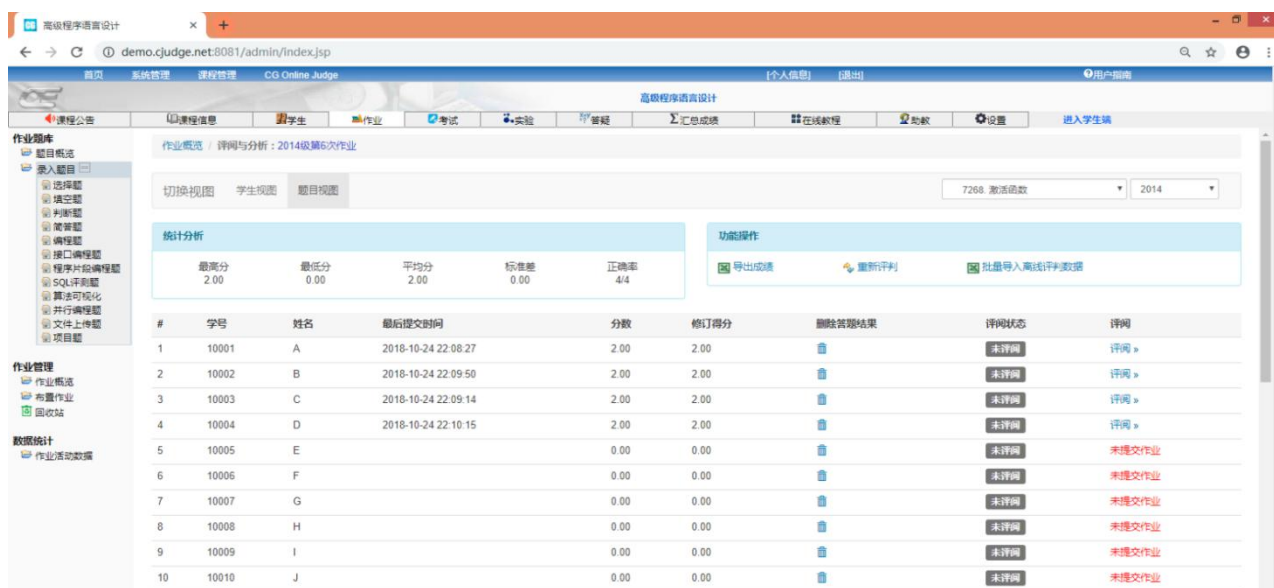
多种评阅视图



对于作业、考试和实验，CG 平台提供多种视图方式供教师进行评阅分析，包括班级视图、学生视图、班级视图。班级视图评阅如上图所示，可直观的看到当前班级作业的完成情况。学生视图评阅如下图所示，教师可在班级视图下选择某个学生单独对其完成的每道题目进行评阅。对于可自动评测的题目，会显示该学生的评测得分。教师可修订每个学生的得分，并添加评语。



题目视图评阅如下图所示，可选择作业中的某一道题目进行评阅，可从当前页面上看到不同班级对该题目的完成情况，系统自动度量该题目的最高分、最低分、平均分、标准差、正确率等信息。

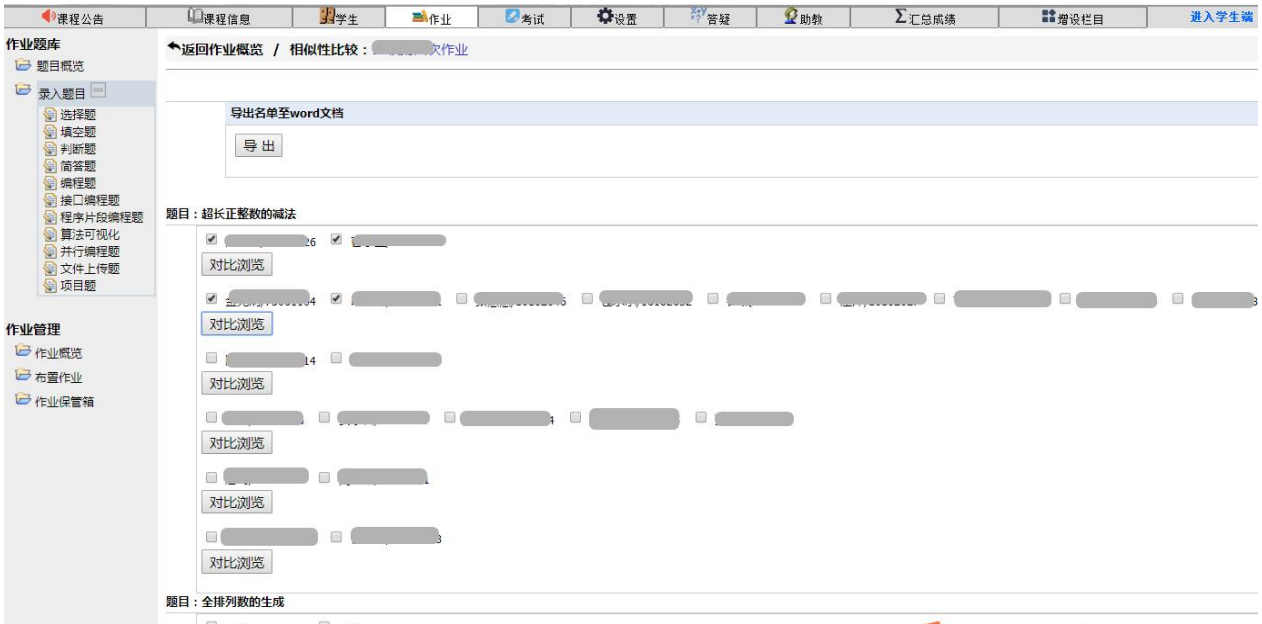


代码查重和文档查重

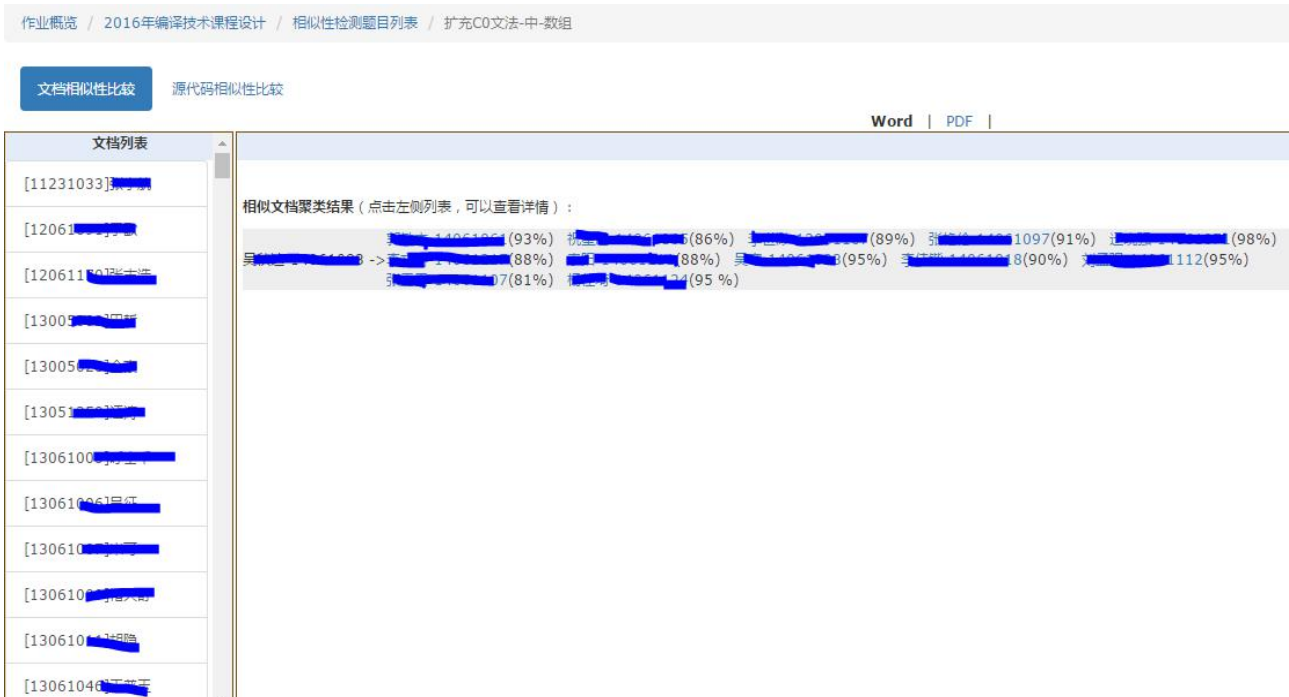
具备代码相似性检测和文档相似性检测功能，可对学生提交的代码和报告分别进行查重。代码查重可检测出修改注释、重新排版、标识符重命名、代码块重排序、代码块内语句重排序、常量替换、改变表达式中的操作符或者操作数顺序、改变数据类型、增加冗余的语句或者变量、表达式拆分、控制结构等价替换等经过深度修改的代码。

6	C++程序设计思想与方法_作业6	CS154	普通作业	从2018-11-13 13:00:00 至2018-12-30 12:00:00	编辑 复制 删除 邮件通知 作业成绩查询	评阅与分析 查重
7	C++程序设计思想与方法_作业5	CS154	普通作业	从2018-11-05 15:45:00 至2018-11-12 23:55:00	编辑 复制 删除 邮件通知 作业成绩查询	评阅与分析 查重
8	C++程序设计思想与方法_作业4	CS154	普通作业	从2018-10-30 18:40:00 至2018-11-08 23:55:00	编辑 复制 删除 邮件通知 作业成绩查询	评阅与分析 查重
9	C++程序设计思想与方法_作业3	CS154	普通作业	从2018-10-23 15:50:00 至2018-10-30 23:55:00	编辑 复制 删除 邮件通知 作业成绩查询	评阅与分析 查重
10	C++程序设计思想与方法_作业2	CS154	普通作业	从2018-10-12 09:00:00 至2018-10-21 23:55:00	编辑 复制 删除 邮件通知 作业成绩查询	评阅与分析 查重
11	C++程序设计思想与方法_作业1	CS154	普通作业	从2018-10-07 16:00:00 至2018-10-14 23:55:00	编辑 复制 删除 邮件通知 作业成绩查询	评阅与分析 查重

代码查重功能能够精准检测出相似的代码，并聚类显示，如下图所示。



文档查重能够检测出经过同义词替换、调整语句或者段落顺序、删除部分段落、更换标题等手段深度修改过的相似性文档，支持 PDF、WORD、PPT、Excel 四种常用的文档格式。



教学数据分析

2983. 北京地铁乘坐线路查询

平均代码行 109 正确率 0.81 平均完成时间 87分钟

类型 编程题 图 4

知识点 难度 章节 录入

2016-03-17 2016-07-15

16:58:24 15:23:21

Yan Haihua

基于历年使用的历史数据统计

【问题描述】
编写一个程序实现北京地铁最短乘坐（站）线路查询，输入为起始站名和目的站名，输出为从起始站到目的站的最短乘坐站换乘线路。注：1. 要求采用Dijkstra算法实现；2) 本题在实际测试时对数据文件进行了调整，使得输入的两站间只有一条最短路径。



返回考试概览 / 试卷评阅：2016级“数据结构与程序设计基础”期中考试

学生视图 题目视图

2016级“数据结构与程序设计基础”期中考试 (B卷)

1439. 正文排版 (两端对齐)

2016级“数据结构与程序设计基础”期中考试 (A卷)

====按题目分类浏览====

统计分析							功能操作
最高分	最低分	平均分	标准差	正确率	平均代码行	平均完成时间(分钟)	重新评判
10.0	0.0	3.86	3.71	76/133	53	54	
学号	姓名	最后提交时间	完成时间(分钟)	分数	修订得分		

支持考试、作业的大数据分析与度量，能够自动度量题目的正确率、代码行（编程题）、平均完成时间等指标，帮助教师客观量化题目难度与特征，如上图所示。

批量导出至Excel 进入“设置 / 题型偏好”，可以隐藏不需要的题型。

编程题							
#	学号	姓名	提交题目	正确题目	代码行	提交次数	答题时间(分)
1			20	20	961	107	1190
2			20	20	784	151	877
3			20	20	788	197	1507
4			20	20	962	164	1637
5			20	20	896	69	963
6			20	19	937	100	661
7			20	20	1039	133	1289
8			20	20	1125	45	418
9			20	20	1037	190	1266
10			20	20	1242	100	1095
11			20	20	1056	138	1292
12			20	20	1069	75	1057

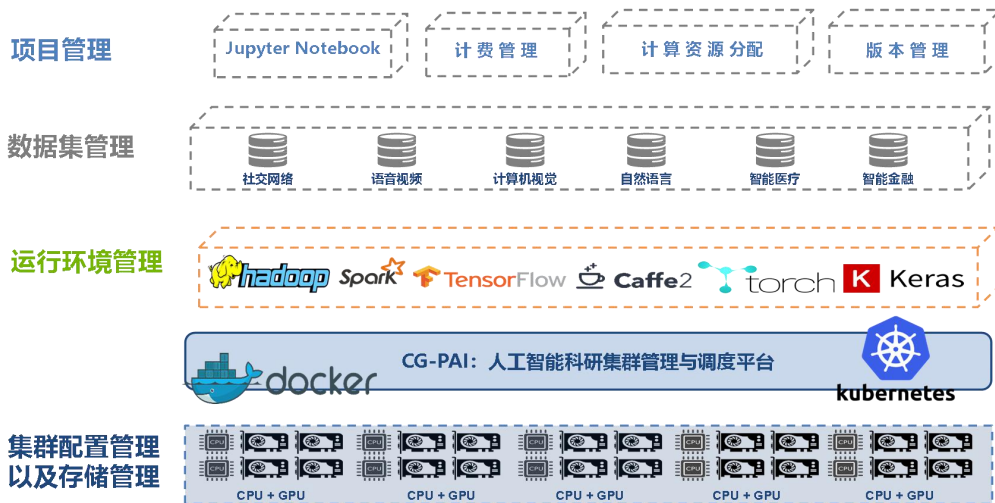
支持统计学生的作业活动分析和考试活动分析，统计每个学生的提交题目数量、正确题目数量、编写代码总行数、提交次数、答题时间等信息，如上图所示。同时，也统计了每个学生的答题明细数据，每个学生在完成每道题目时，平台记录的信息包括当前题目的题型、难易度、知识点、作业 ID、是否通过、完成时间、提交次数、首次通过时间、首次通过时累计提交次数、优化次数、有效优化次数、代码行、圈复杂度、函数度量、首次提交时间、最后提交时间等信息，如下图所示。以上所有统计分析数据可导出 csv 文件，基于离线数据可进行二次分析。

批量导出至CSV(UTF-8编码) 统计数据定义

#	学号	姓名	题目ID	题型	难度	知识点	作业ID	是否通过	完成时间	提交次数	首次AC	首次AC时间	首次AC提交次数	优化次数	有效优化次数	代码行	复杂度	函数数量	首次提交时间	最后提交时间
1			64664	编程题	0		401	1	5693	2	1	0	1	1	1	17	1.00	1	2018-10-07 23:17:10	2018-10-07 23:17:14
2			64663	编程题	0		401	1	4441	10	0	2008	9	1	1	15	1.00	1	2018-10-07 21:08:38	2018-10-07 21:42:12
3			69246	编程题	0		442	1	993	3	0	935	2	1	1	15	7.00	1	2018-10-13 14:35:41	2018-10-13 14:51:20
4			69247	编程题	0		442	1	975	3	0	18	2	1	1	33	20.00	1	2018-10-13 15:05:43	2018-10-13 15:07:41
5			68865	编程题	0		514	1	4005	1	1	0	1	0	0	39	10.00	1	2018-10-25 22:39:12	2018-10-25 22:39:12
6			68864	编程题	0		514	1	15	2	0	89419	2	0	0	34	9.00	1	2018-10-26 12:54:23	2018-10-27 13:44:42
7			81389	编程题	0		570	1	2534	7	1	1	1	6	6	28	5.00	1	2018-10-31 22:27:56	2018-11-01 15:32:13

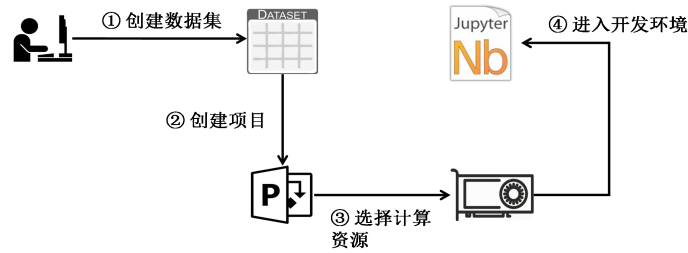
8 人工智能科研平台：CG-PAI

科研平台介绍



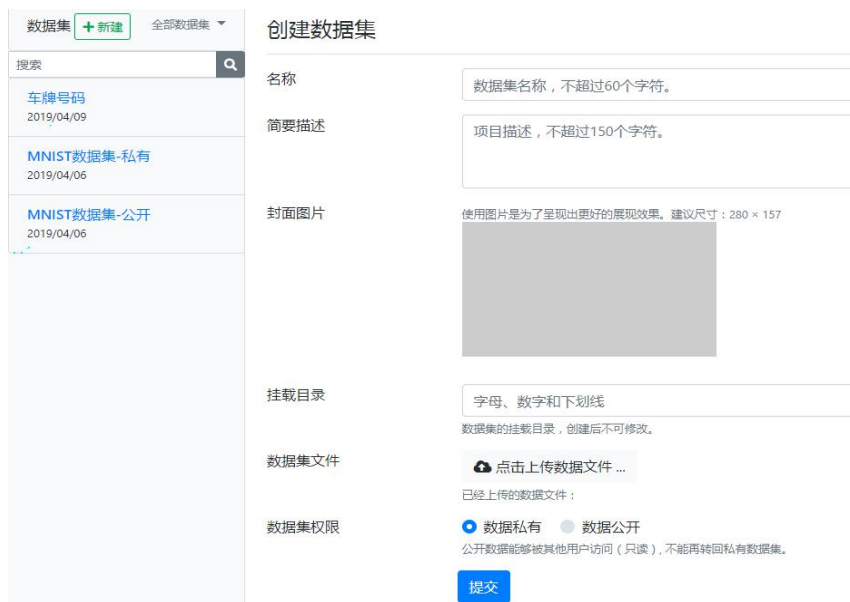
GPU + k8s + Docker + DL Tools+Jupyter

CG-PAI 是 CG 平台为科研用户设计的深度学习与新一代人工智能科研平台。CG-PAI 提供了数据集管理、项目管理、集群配置管理、存储管理、运行环境管理以及项目管理功能，如上图所示。在用户层，用户可通过 CG-PAI 平台管理数据集、管理科研项目、分配计算资源、对计算资源进行计费。为了方便用户开展科研活动，CG-PAI 提供了 Jupyter 笔记交互方式。通过 Jupyter，用户可交互式的对数据进行分析、挖掘和研究。CG-PAI 内置多种科研数据集和科研案例，集成各类深度学习和机器学习框架，并为用户提供了多种交互方式。在资源层，通过 k8s 负责集群配置管理、集群计算资源的管理与调度以及运行环境的管理。利用 Docker 技术将所有计算资源 Docker 化，通过 k8s 编排、管理、调度、监控所有 Docker 资源。在资源层，用户可将不同的深度学习框架以 Docker 镜像的方式进行管理。在集群层，CG-PAI 平台对硬件资源进行定义，将多台异构计算节点、管理节点、交换机、存储节点组成一套完整的高可用、高可靠、高性能集群。



CG-PAI 的最大优势是易学、易用。用户可以在 1 分钟内学会科研平台的使用，每个项目可在 1 分钟内创建完毕并进入开发环境。如上图所示，在 CG-PAI 上，用户只需要简单地通过以下 4 步，即可创建完毕科研项目，并进入开发环境中开展科研代码的编写、调试和运行。

- (1) 创建数据集，可将自己科研项目的数据集上传到集群的存储系统上，数据集分为公有数据集和私有数据集两种类型。



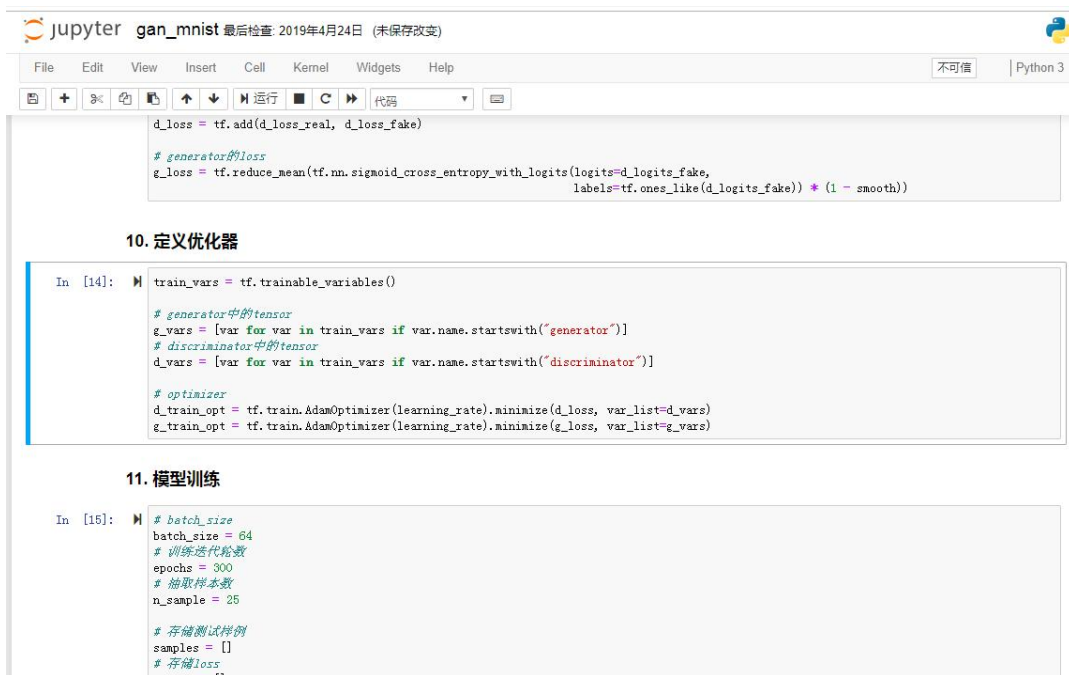
- (2) 创建项目，选择运行环境，关联数据集。选择运行环境时，可选择需要的深度学习框架。



(3) 运行项目，选择需要的计算资源，包括 CPU 核数、内存大小、GPU 卡数等，如下图所示。



(4) 进入开发环境，打开 ipynb 笔记文档，开始编写、调试和运行科研代码，如下图所示。



在科研平台的开发环境中，用户可保存当前科研项目的版本。科研平台每个科研用户的计算资源使用情况进行计费，计费的信息包括总时间、CPU 时间（CPU 核数*分钟）、内存时间（GB*分钟）、GPU 时间（GPU 卡*分钟）。在管理端，可配置每个科研用户的资源配额；支持集群配置管理，用户可动态增加或者更换集群节点；支持集群存储配置管理，可添加或者删除科研平台的共享存储系统，当前支持 NFS、GlusterFS 并行文件系统，科研数据与运行环境镜像都存放于这些存储系统上。系统支持多个共享存储目录，且支持自动在多个存储系统间进行容量均衡。

CG-PAI 支持运行环境管理，支持 Jupyter 官方 Docker 镜像和以此为基础构建的 Docker 镜像。用户可添加新的 Jupyter Docker 镜像，添加时可指定 Docker ID、Docker 标签以及 Docker 描述，可勾选是否依赖于 GPU。在增加计算节点后，支持一键分发镜像。对于现有的镜像，用户可上传 Docker 镜像文件更新现有镜像。上传后，可自动重新分发镜像。支持环境配置重载，如果存储临时故障并修复，或在存储设备上直接修改了镜像配置，可一键重新加载环境配置。

系统目前只支持Jupyter官方Docker镜像和以此为基础构建的Docker镜像。

DockerID	Docker标签(tag)	镜像描述	镜像文件包	镜像分发状态	删除	编辑
jupyter-datascience-notebook	jupyter/datascience-notebook:latest	includes libraries for data analysis from the Julia, Python, and R communities.	更新Docker镜像 选择文件 未选择任何文件 下载Docker镜像	100% 查看日志		
jupyter-tensorflow-notebook	jupyter/tensorflow-notebook:latest	popular Python deep learning libraries.	更新Docker镜像 选择文件 未选择任何文件 下载Docker镜像	100% 查看日志		
jupyter-r-notebook	jupyter/r-notebook:latest	R语言常用的开发包	更新Docker镜像 选择文件 未选择任何文件 下载Docker镜像	100% 查看日志		

教学科研一体化

可将 CG 教学平台中采集的数据批量导出后，上传到科研平台，基于科研平台提供的计算资源、开发环境，对导出的教学实验过程性数据进行分析，实现教学科研一体化的目标。

9 平台质量

高并发

- 单台服务器支撑 20000 人同时在线进行程序评测考试。
- 单台服务器支撑 300 人同时开展虚拟桌面实验。
- 单台服务器支撑 300 人同时开展 Jupyter 笔记实验。
- 通过简单配置 VNC 代理，10 台服务器可支撑 3000 人同时开展虚拟桌面实验。
- 虚拟桌面实验环境、Jupyter 笔记实验环境、Docker 实验环境、通用评测环境都支持分布式自动化部署，只需录入新服务器的 IP 地址和 root 密码，即可实现实验环境的弹性扩展，可支撑数千规模的高并发量。

高性能

- 程序评测采用多核并行评测，实时反馈评测结果，无需排队。
- 本地部署后，虚拟桌面打开时间在 1 秒以内。
- 在千兆网的校园网内，并发打开 1000 个实验桌面总耗时在 3 秒内。
- Jupyter 笔记实验环境打开时间在 5 秒内。

高可靠

- 程序评测采用沙箱机制。
- 数据库密码 2 次 MD5 单向加密。
- 教师账号临时秘钥加密登录，防暴力破解。

- 后台定期运行自检程序，发现问题反馈给管理员和教师。
- 自动修复断电引起的数据库表状态不一致问题。
- 多重手段保护数据安全性，支持一键手工备份、异存储介质备份、远程备份、双机备份，可同时开启多种备份机制。

自动备份数据 系统每周自动备份数据，并保留最近一个月的备份数据。

关闭自动备份

备份路径 确定

若同时备份多处，路径之间逗号隔开。备份数据默认存放在系统安装路径下。

备份数据列表:

- 备份时间：2019-06-09 04:03，下载备份文件：[CGImage_20190609.tgz](#) (1251 MB)
- 备份时间：2019-06-16 04:03，下载备份文件：[CGImage_20190616.tgz](#) (1250 MB)
- 备份时间：2019-06-23 04:40，下载备份文件：[CGImage_20190623.tgz](#) (1259 MB)
- 备份时间：2019-06-30 04:04，下载备份文件：[CGImage_20190630.tgz](#) (1259 MB)

自动备份帮助

- 每周执行一次全备份，执行全备份预防系统数据被误删或者设备故障造成的损失。
- 建议把数据备份到不同的磁盘设备上，即使服务器故障，也不会丢失系统数据。推荐配置方法：
 - 如果有条件，可以挂载一个磁盘阵列，备份路径设置到该磁盘阵列上。
 - 可以在服务器上插一块移动硬盘(或优盘)，把备份路径设置到该移动硬盘上。
 - 可以在服务器上挂载一块独立的磁盘，备份路径设置到该磁盘设备上。
- 高级用法：存储路径可以设置多个（逗号隔开），每个路径对应不同的存储设备，数据可靠性更高。

数据同步 系统每晚将数据自动(增量)同步至其它服务器。

同步机器IP

该服务器能够从CG Web服务器 ssh 登录，通单只需要配内网IP。

用户名

在备份机器上以该用户保存同步文件。

用户密码

同步方式 远程备份 双机同步

若选双机备份，同步服务器必须安装CG系统。

备份路径

确定 关闭数据同步

同步日志:
.....

数据同步帮助

- 每天晚上将主服务器的数据增量同步到其它服务器。如果遇到特殊情况，可以快速恢复服务。
- 同步服务器上的数据与主服务器保持一致。
- 同步服务器与主服务器必须安装rsync。

高扩展

- 虚拟桌面实验环境及实验资源可扩展。
- Jupyter 笔记实验环境及实验资源可扩展。
- 自动评测机及自动化评测题目可扩展。
- Docker 实验环境及 Docker 实验资源可扩展。
- 课程资源（作业、考试、MOOC 等）可扩展。
- 硬件服务器可扩展。
- 学生并发人数可扩展。

易维护

- 服务器重启后，CG 平台服务自启。
- CG 服务自启后自检，常规问题自动修复。
- 支持一键恢复，上传备份数据，一键系统恢复。
- 支持一键升级，上传升级包后，一键系统升级。
- 支持详细操作日志下载及异常环境与异常操作报警。

10 平台生态建设

集体智慧

教师可在 **CourseGrading** 高校用户交流群中反馈新的需求，**CG** 开发团队和已部署的用户会根据需求的通用性，讨论和决定是否在后续版本中实现该需求。当前 **CG** 的更新频率为 1 个月到 1.5 个月，因此通过讨论的需求，通常会在 1 到 1.5 个月内体现在下个版本中。

CG 最近升级内容可参阅：<http://www.educg.net/download.html>

师资培训

提供多种师资培训形式：

- 提供 **CG** 平台使用的现场培训。
- 提供针对相关课程教学与实验的师资培训。
- 教师可参加新工科联盟的师资培训课程，目前已开设的课程包括：操作系统、组成原理、软件工程、数据结构等。

视频与讲义

免费赠送覆盖各专业课程的视频教程和 PPT 讲义。视频教程和 PPT 讲义可导入到 **CG** 平台的 MOOC 模块，既可用于日常教学，也可由学生自学。

11 演示与试用

本手册所介绍的内容可在 **CG** 平台在线演示系统中进行体验与验证。在线演示系统开放了 **CG** 平台的所有功能，由于实验资源的可复制性，每门课程中仅放入了部分题库资源和实验资源。**CG** 平台提供无限期免费试用，提供本地安装部署试用服务。在线演示系统访问入口：<http://educg.net/demo.html>

可随时咨询 **CG** 平台的技术与产品问题，联系方式：李超 18600162252（手机、微信） 794288684（QQ）

CG 平台高校用户交流群：



12 参考文献

- [1] 赵长海, 晏海华, 金茂忠. 基于编译优化和反汇编的程序相似性检测方法[J]. 北京航空航天大学学报, 2008, 34(6).
- [2] 赵长海, 晏海华, 贾宝龙, 等. 并行程序设计课程学生作业的自动评判方法[J]. 计算机教育, No.170(14):73-77.
- [3] 王雷, 高超, 沃天宇. 面向实验过程的操作系统实验集成环境[J]. 计算机教育(05).
- [4] 高小鹏. 计算机专业系统能力培养的技术途径[J]. 中国大学教学, 2014(8).
- [5] 高小鹏, 龙翔. 一体化教学实验体系初探[J]. 计算机教育, 2009(19):43-46.
- [6] Zhao C , Yan H , Song C . A System for Automatic Assessment and Plagiarism Detection of Student Programs[C]// The International Conference on E-Business and E-Government, ICEE 2010, 7-9 May 2010, Guangzhou, China, Proceedings. IEEE, 2010.
- [7] A gallery of interesting Jupyter Notebooks, <https://github.com/jupyter/jupyter/wiki/A-gallery-of-interesting-Jupyter-Notebooks>, [EB/OL]
- [8] Deeplearning models, <https://github.com/rasbt/deeplearning-models> [EB/OL]
- [9] Data science ipython notebooks, <https://github.com/donnemartin/data-science-ipython-notebooks> [EB/OL]
- [10] Python data science handbook, <https://github.com/jakevdp/PythonDataScienceHandbook> [EB/OL]
- [11] Apache spark notebook, <https://github.com/spark-notebook/spark-notebook/tree/master/notebooks> [EB/OL]
- [12] Numerical linear algebra, <https://github.com/fastai/numerical-linear-algebra> [EB/OL]
- [13] Linear algebra notes, <https://github.com/zlotus/notes-linear-algebra> [EB/OL]
- [14] Machine learning, <https://github.com/udacity/machine-learning> [EB/OL]
- [15] Python machine learning book, <https://github.com/rasbt/python-machine-learning-book> [EB/OL]
- [16] Homemade machine learning, <https://github.com/trekhleb/homemade-machine-learning> [EB/OL]
- [17] MIT deep learning, <https://github.com/lexfridman/mit-deep-learning> [EB/OL]
- [18] Deep learning with python notebooks, <https://github.com/fchollet/deep-learning-with-python-notebooks> [EB/OL]
- [19] Deep learning zero to all, <https://github.com/hunkim/DeepLearningZeroToAll> [EB/OL]
- [20] Deep learning, <https://github.com/udacity/deep-learning> [EB/OL]
- [21] Deep reinforcement learning, <https://github.com/AppliedDataSciencePartners/DeepReinforcementLearning> [EB/OL]
- [22] Deep learning keras tensorflow, <https://github.com/leriomaggio/deep-learning-keras-tensorflow> [EB/OL]
- [23] Deep learning pytorch, <https://github.com/udacity/deep-learning-v2-pytorch> [EB/OL]
- [24] GAN notebooks, <https://github.com/tjwei/GANotebooks> [EB/OL]
- [25] Autoencoders, <https://github.com/nathanhubens/Autoencoders> [EB/OL]
- [26] Industry machine learning, <https://github.com/firmai/industry-machine-learning> [EB/OL]
- [27] 2019《操作系统课程设计》师资培训通知, <http://educg.net/os.html> [EB/OL]
- [28] 2019 计算机组成 CPU 工程化方法师资培训通知, <http://educg.net/computer.html> [EB/OL]

用户

北京航空航天大学
上海大学计算中心
上海交通大学
中国建设银行总行金融科技部
北京交通大学计算机与信息技术学院
北方工业大学
国防科学技术大学
江南大学
武汉大学
北京科技大学
浙江传媒大学
中国石油大学
北京信息科技大学
武汉理工大学
火箭军工程大学
中国矿业大学
江南大学
华中农业大学
齐鲁工业大学
华北电力大学
青岛科技大学
杭州电子科技大学
大连民族大学
山东科技大学
济南大学
重庆邮电大学
哈尔滨工程大学
西南民族大学
沈阳航空航天大学
华南理工大学
电子科技大学
长春理工大学
哈尔滨理工大学
重庆交通大学
东北大学
合肥工业大学
山东大学
中国人民解放军陆军炮兵防空兵学院
中国海洋大学
南京理工大学
湖南大学
.....

联系我们

系统官网: www.educg.net
技术研发: 北京航空航天大学计算机学院
投资运营: 郑州云海科技有限公司
教学在线: course.cjudge.net
竞赛在线: acmoj.com
商务联络: (北京) 19910509699
(郑州) 18530023918
0371-56003691
QQ:737133234
业务合作: yunhaikj@aliyun.com
公司地址: 郑州郑东新区和谐路东连邦大厦
713 室

版权所有 © 郑州云海科技有限公司 2018。保留一切权利。

非经郑州云海科技有限公司书面同意, 任何单位和个人不得擅自摘抄、复制本手册内容的部分或全部, 并不得以任何形式传播。