**自动化测试流程**

1.登录GitLab平台(http://10.2.28.170/users/sign\_in)

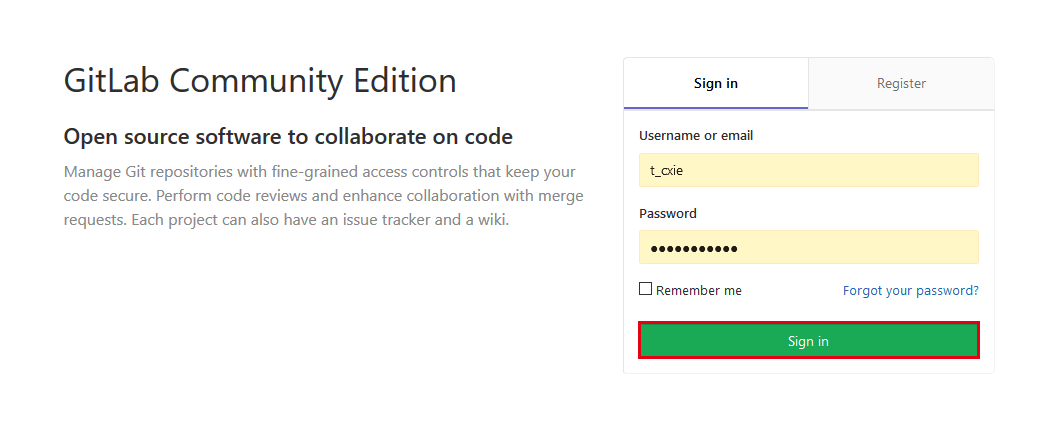


图1 登录平台

2. 进入软工平台

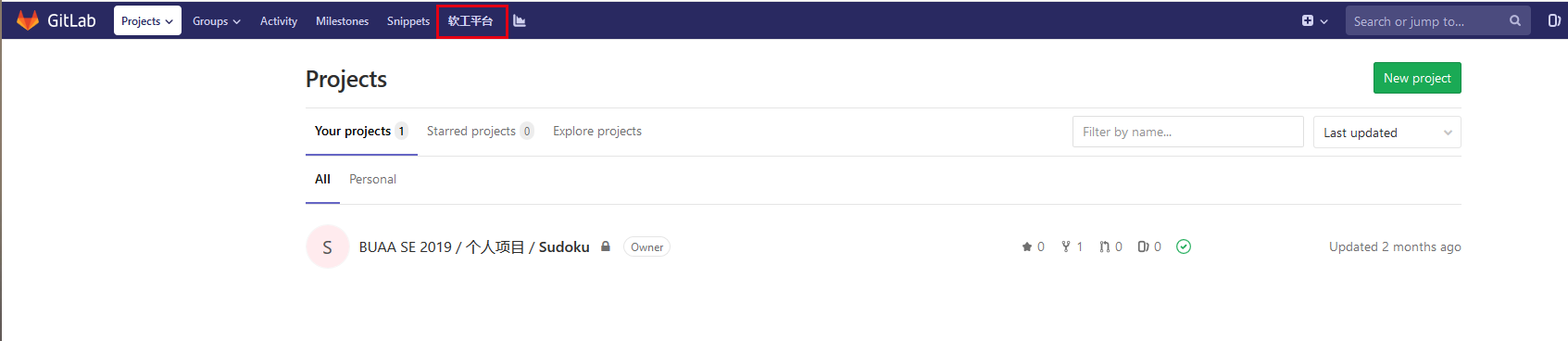


图2 进入软工平台

3.以教师身份登录

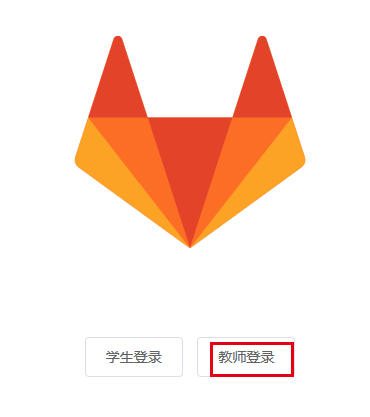


图3 教师登录

4.如果已经存在班级，则点击班级进入，如果还没有，则点击创建班级。（以BUAA SE 2019班级为例）

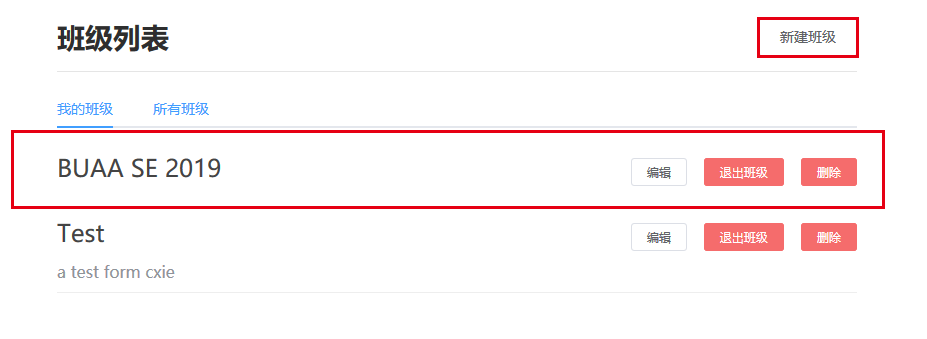


图4 创建班级

5.在添加项目时以个人项目为例，进入个人项目，点击创建个人项目，开始创建,（以创建项目Sudoku为例）。项目中需要包含有关单元测试的相关文件才可进行单元测试。此外，项目中还需要在.gitlab-ci.yml文件中配置CI任务即可实现自定义的自动测试。编译服务器上提供了编译和单元测试的工具，如果没有相应项目的单元测试工具，助教可以通过GitLab部署针对个人项目或结对项目作业开发的自动测试程序。



图5 创建个人项目

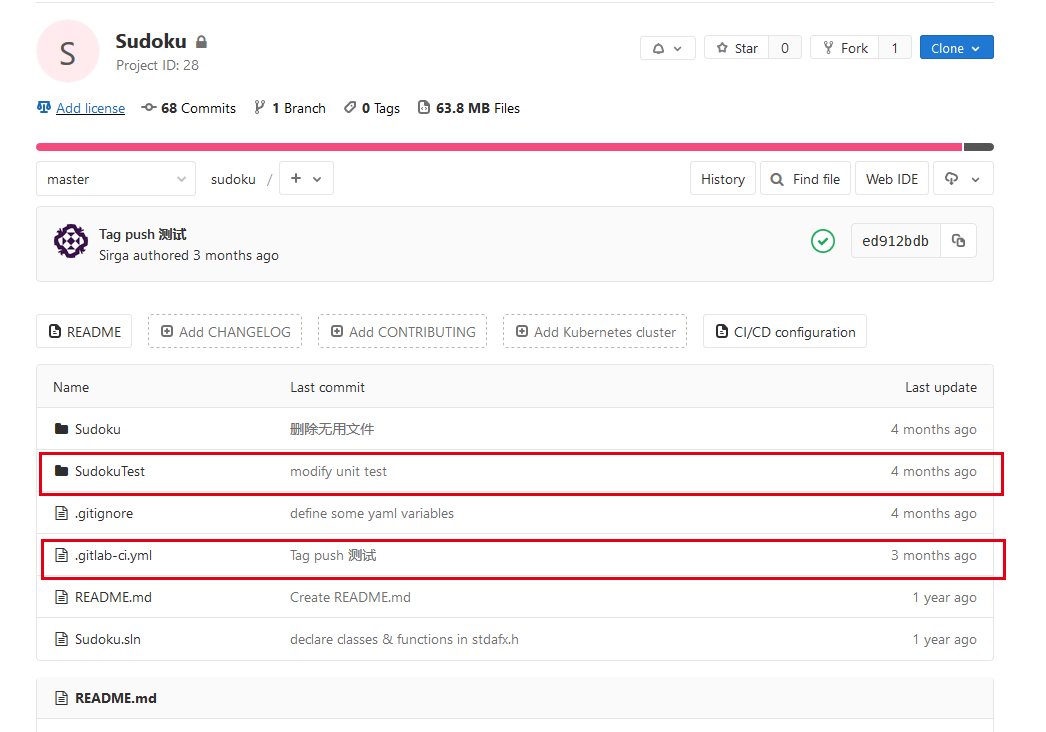


图6 项目中需包含单元测试文件和 .gitlab-ci.yml文件

6.创建好的个人项目，学生可通过fork将项目复制在学生自己的GitLab仓库。如果学生完成项目并提交到GatLab后，教师可点击自动测试进入自动测试页面。



图7 进入自动测试

7.进入自动测试页面后，点击Start CI，更新测试结果，完成自动测试。

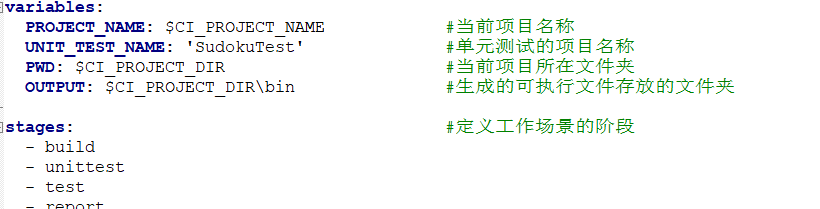


图8 Start CI 更新测试结果

8.项目中.gitlab-ci.yml配置文件的书写说明(以项目Sudoku为例)：

配置文件主要三部分组成

i.初始化阶段：先定义配置中所需要的变量，包括当前项目名称，项目所在文件夹以及CI工作场景的各个阶段等。

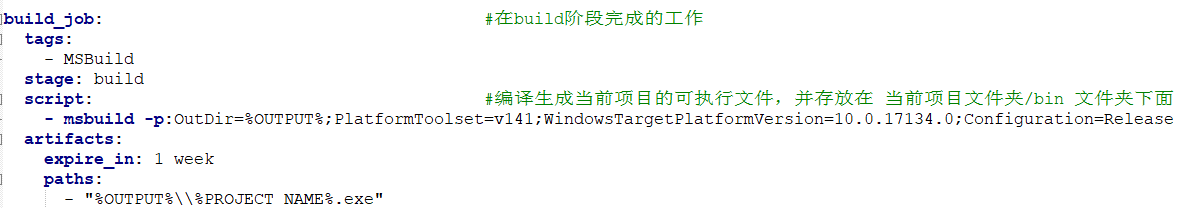


ii.定义作业开始执行之后以及执行之后的操作(本例中没有执行之后的操作)



iii.设置每个阶段的动作

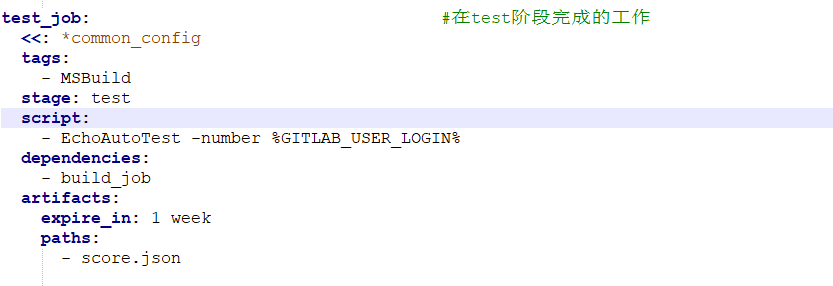
a. build阶段



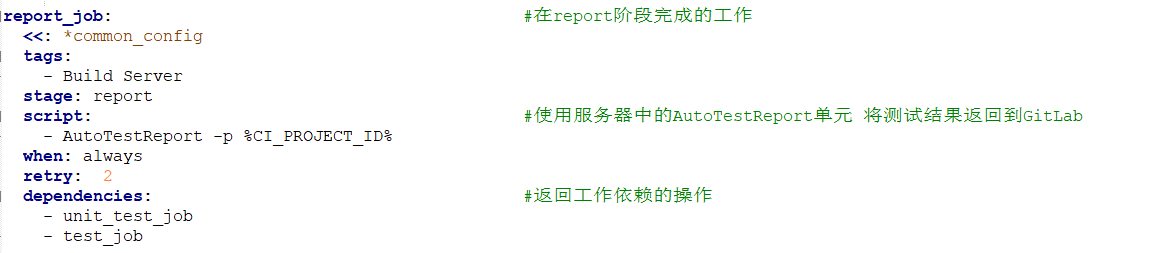
b. unittest阶段



c. test阶段



d. report阶段



注：

1.教师/助教可以通过修改服务器上的编译和单元测试的工具并上传到GitLab来修改测试用例和评测标准。

2.在配置文件中的test\_job 动作中，如果服务器中没有对应项目的测试文件，需要手动将测试程序添加到服务器。在服务器中部署测试项目的过程见附录1

3. 教师提供的个人项目的需要在项目配置中采用Release x64，并且学生运行测试也应在Release x64环境下。

4.处理fopen等函数时的安全性问题时， 需要在Release配置下的所有平台(包括x64 win32)都添加\_CRT\_SECURE\_NO\_WARNINGS;，学生可通过附录2中的测试情况，对项目进行相应的修改。

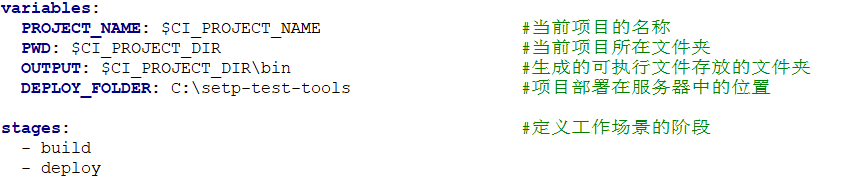
5.学生查看单元测试过程具体步骤见附录2

附录1 在服务器中部署测试项目(以部署SudoTestAutoTest为例)

将自动测试项目SudoTestAutoTest项目上传到GitLab并配置相应的 .gitlab-ci.yml文件用来将项目生成的可执行文件（如 SudukuAutoTest.exe）部署到 C:\setp-test-tools，即可在自动化测试中使用该命令（SudokuAutoTest），.gitlab-ci.yml 配置文件的书写说明如下：

配置文件主要三部分组成:

i.初始化阶段：先定义配置中所需要的变量，包括当前项目名称，项目所在文件夹以及CI工作场景的各个阶段等。



ii.定义作业开始执行之后以及执行之后的操作(本例中没有执行之后的操作)

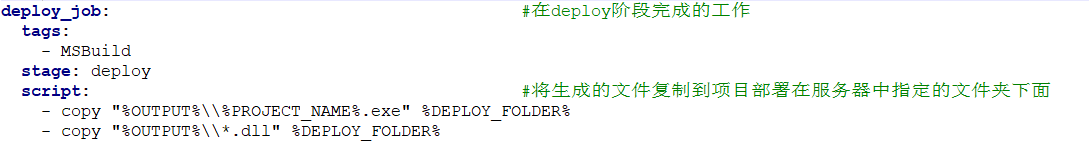


iii.设置每个阶段的动作:

a. build阶段



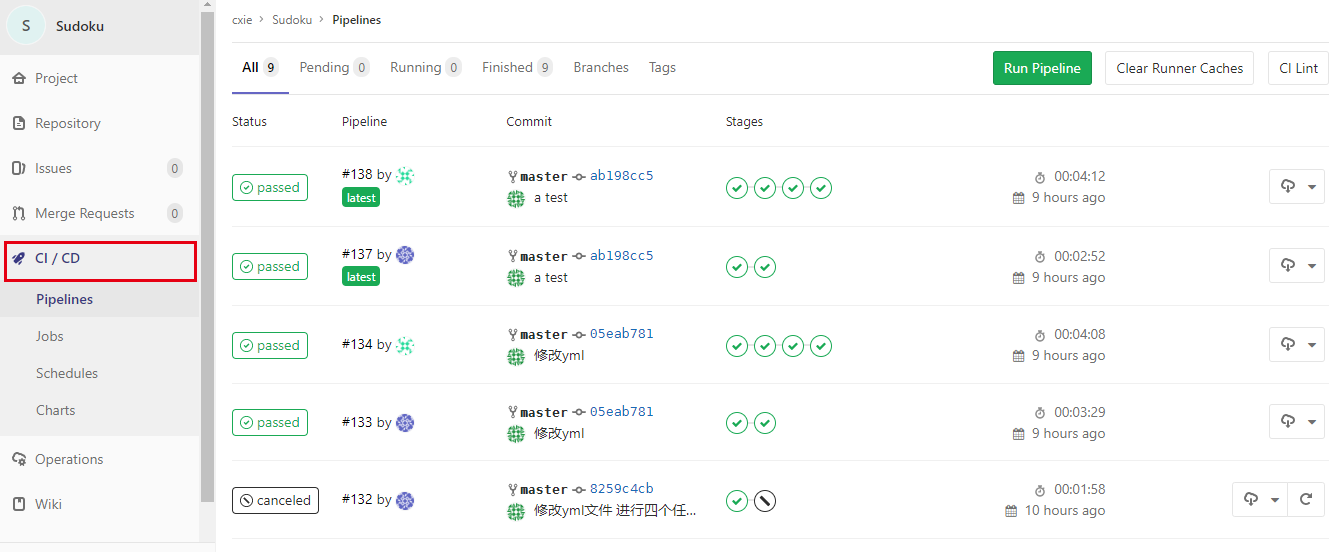
b. deploy阶段



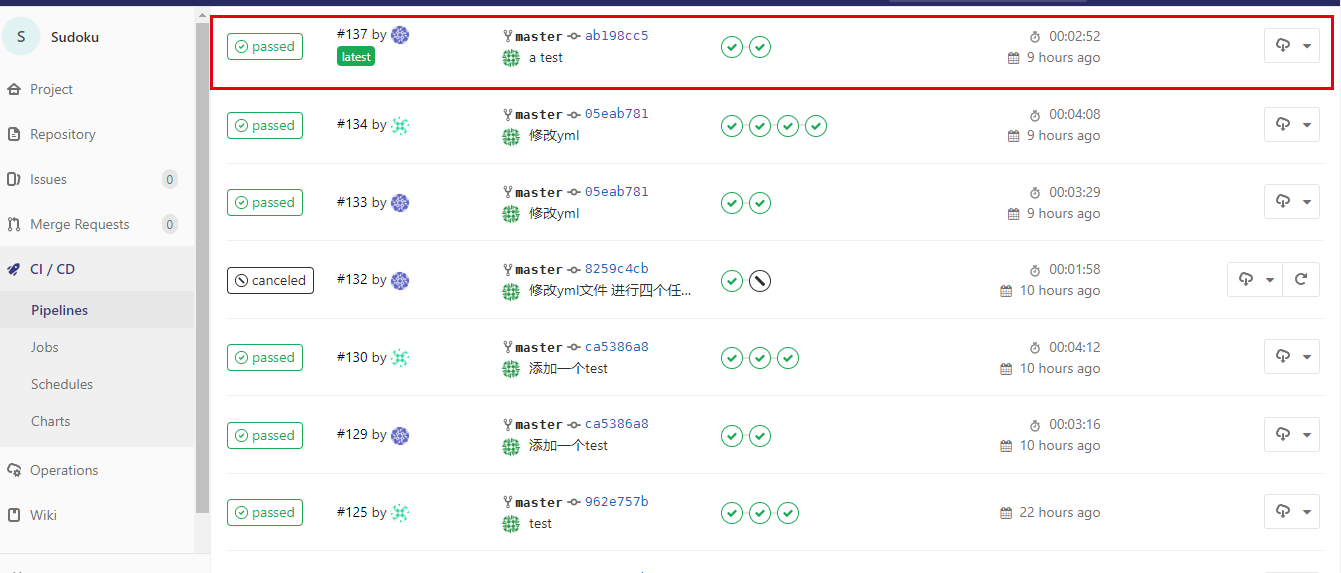
附录2 学生查看单元测试过程(以查看Sudoku项目为例)

学生通过GitLab提交项目之后，可以通过以下方式查看自己的单元测试、正确性测试、鲁棒性测试结果。

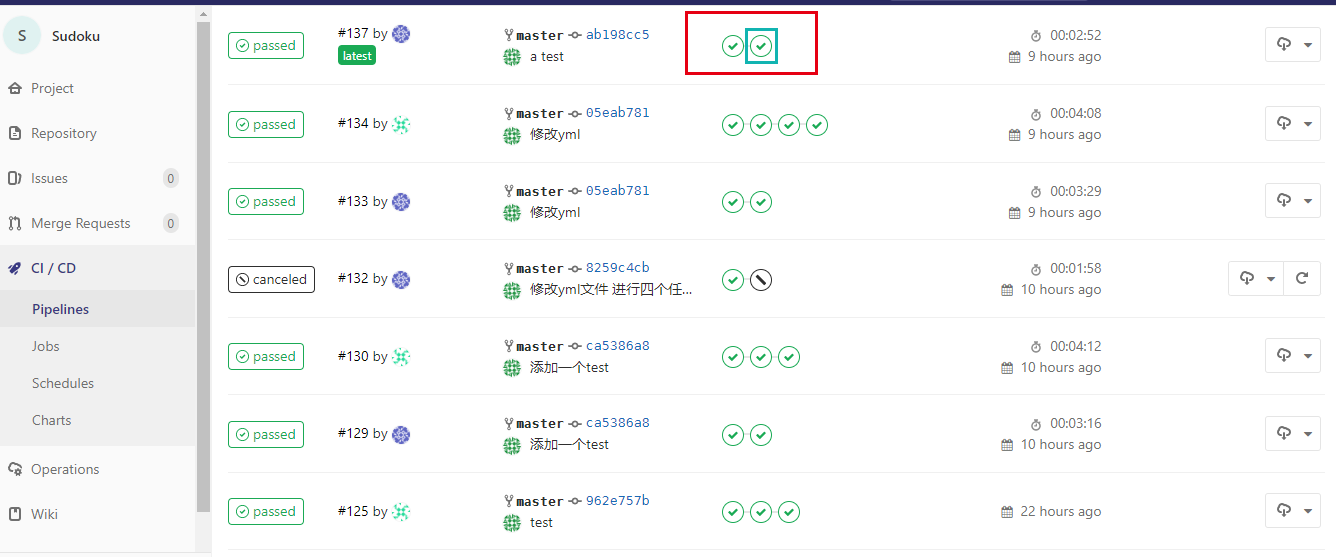
1.提交项目后，点击左侧的CI/CD 菜单选型，进入CI/CD页面



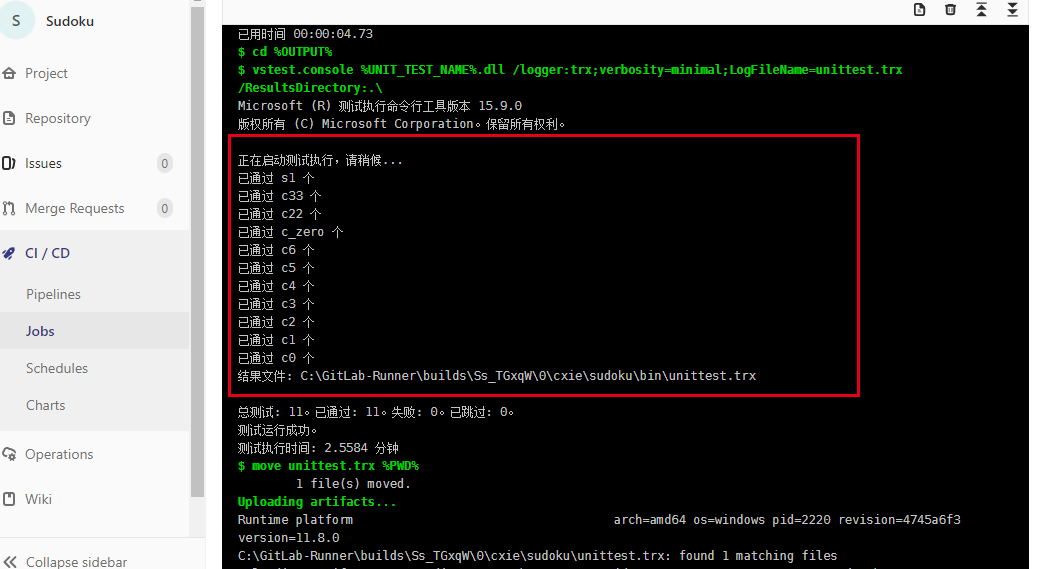
2.项目push之后，在Pipelines会出现一个任务



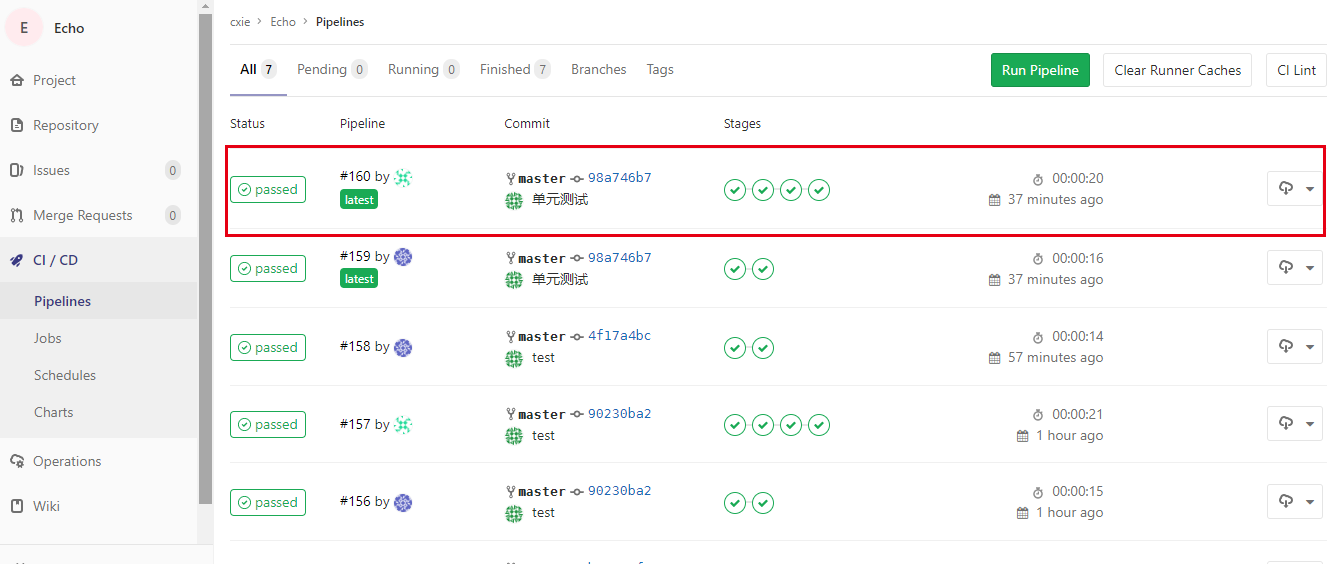
3.点击对应的任务，进入提交后单元测试页面，其中红框中的第二个任务即为单元测试任务



4.进入后，即可查看可识别的单元测试通过的情况



5.当教师执行Start CI之后(不要点Cancel CI，否则学生端任务就会消失)，学生端出现一个具有四个阶段的流水线任务。其中，第二个阶段为单元测试，第三个阶段为正确性测试和鲁棒性测试，第四个阶段为向GitLab提交测试报告。学生可点击进入相应的阶段查看结果



6.项目正确性和鲁棒性测试结果

