

阅读提问（《代码大全2》）

1、

第四章中提到了Ada编程语言，是现代计算机语言的成功代表。但是我却很少听说，在学校也没有相应的介绍课程，总结网友的评论，Ada的编译器最初需要美国国防部的认证，是标准化的，所以缺少商业化的竞争，各公司没法靠卖编译器赚钱，所以就没人推广，但是，我觉得作为一个安全性很高的语言，也是具有一定的学术研究价值的。此外，书中提到的cobol、fortran和Visual Basic也很少听人提起。因此，我好奇为什么没有介绍它们的选修课程？

2、

如何理解深入一种语言编程和在一种语言上编程的区别，书中说编程时不要受到语言特性的限制，而是先确定自己的思想，而后通过语言特性去实现它。我的理解就是，对于一些好的编程方法，如果那个编程语言并没有现成的函数，就自己去实现它，或者对编程语言添加自己设定的约束。但是在实际编程时，如何评判这种做法的利弊，或者如何设定合理的约束？

3、

在开发者测试章节中，书中提到先行测试，即提前设计好测试用例，然后经过修改和完善，让产品代码满足测试用例的需求，相比于编码之后编写测试用例，这种做法能够缩短缺陷-侦测-调试-修正这一周期。但是在我看来，如果事先设计好的测试用例就考虑不周，或随着需求的改变导致某些测试用例不再合法，强行让项目满足这些“错误”测试用例的需求也会让开发者走弯路，毕竟测试时，首先考虑的是代码的问题，而不是测试用例的问。而且随着项目的开展，开发者对项目会有更加明确的认识，在编程时也会考虑到可能出现的错误，或许在编码之后编写测试用例是一种更好的选择？

4、

书中提到，细心的程序员不到万不得已，是不会用宏来代替子程序的。但是存在即合理，宏和子程序都有着自己的优缺点，子程序模块化，节省内存，可被多次调用，编程效率高，而宏调用参数传送简单，执行效率高。如果能够按照语言的规则合理使用宏调用，可以带来很多方便。

5、

关于用户调研的方法，目前最常见的就是网络上的用户调查问卷，提到了有全开放性问题、二项选择题、多项选择题，顺位选择题等等。我们预期全开放性的问题会让用户畅所欲言，但是实际情况却是用户往往会留空，或者只是输入“暂无”，毕竟回答这样的问题会比单纯选择更累。我认为还可以补充一种更加有效方式：在设计调查问卷的时候，就应该多考虑一些可能的情况罗列出来，最后一项为“其他”，如果用户真的有我们未考虑到的情况可以在此处添加。

调研源代码版本管理软件

GitHub

github能够免费建立私有仓库，能够同时支持最多3个协作者，便于独立开发和小团队开发。尽管它的功能比较完善，但是在国内访问时并不稳定，需要加速器来维持较好的使用体验，这也催生了gitee（国内的Git托管服务）。

GitLab

相比于github，gitlab具有wiki和issue跟踪功能，并且很好的集成了CI/CD，企业一般有自己企业专属的gitlab，ruby（一般专业课）的课程网站就是一个gitlab，体验相比github稍好。

Bitbucket

5人以内的小团队使用可以使用无限量的免费存储库，不限容量。集成 Jira 工具，通过集成的错误跟踪组件，Jira 自动更新有关检测到的问题的信息。速度快、权限管控灵活、自定义域名、支持 wiki，但是使用群体不如前面两种，系统不够稳定。

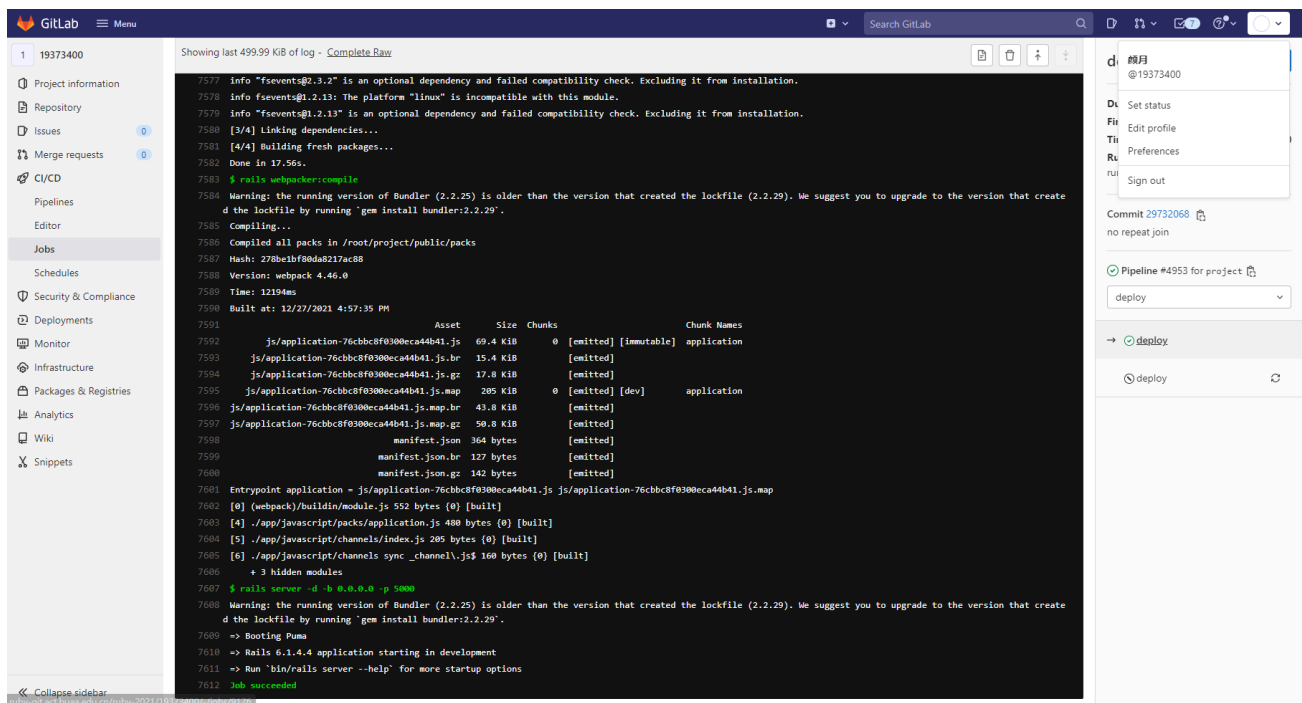
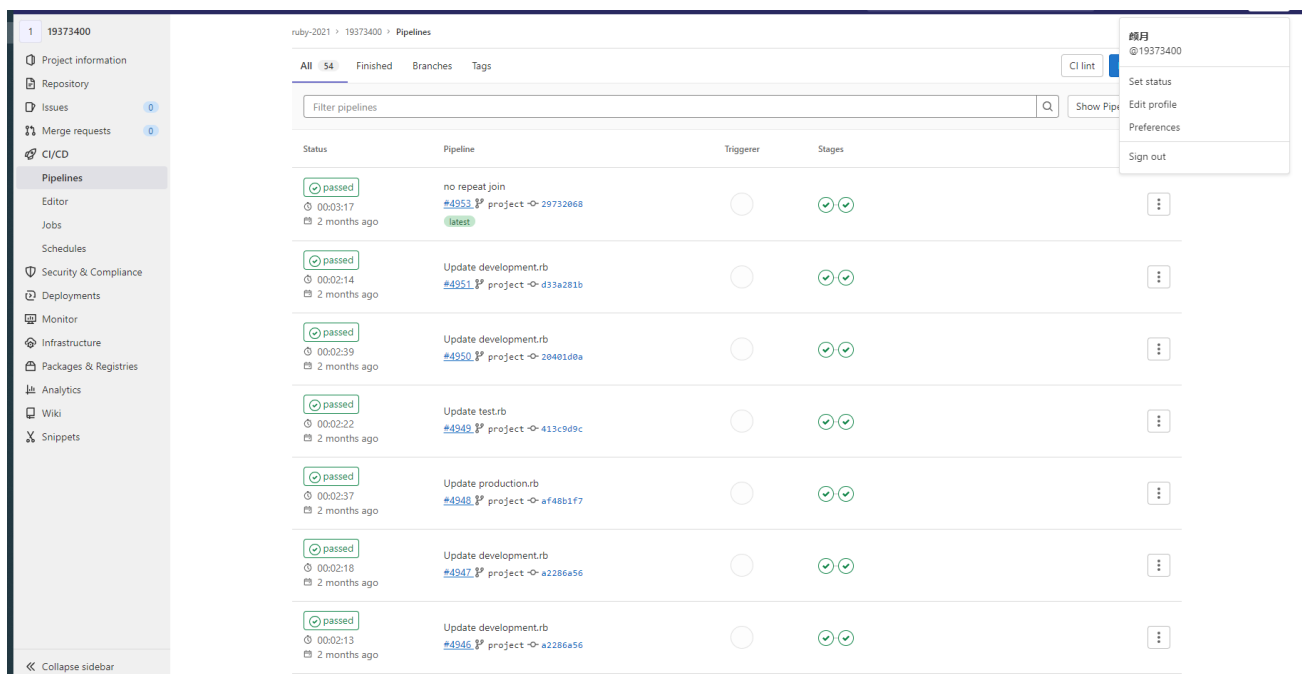
调研持续集成/部署工具

1、ruby课程项目

一个类似于论坛的网站，代码仓库在ruby课程组的gitlab中，无法访问

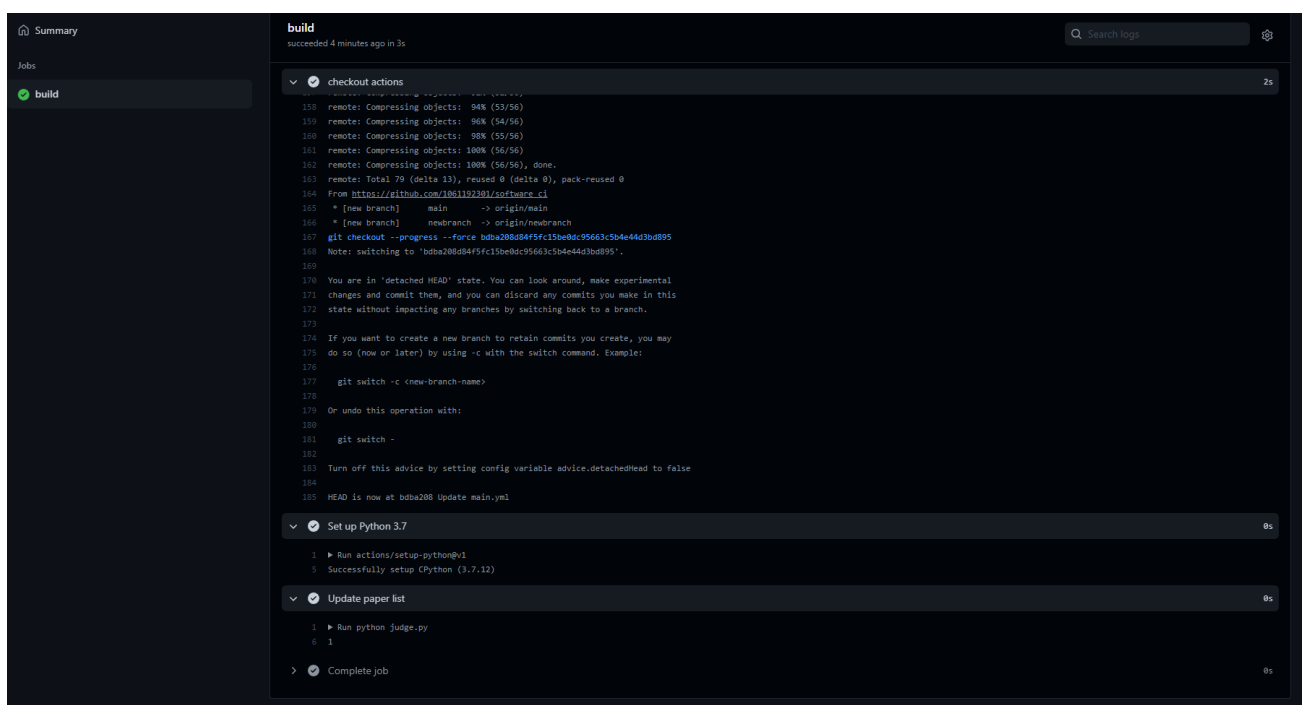
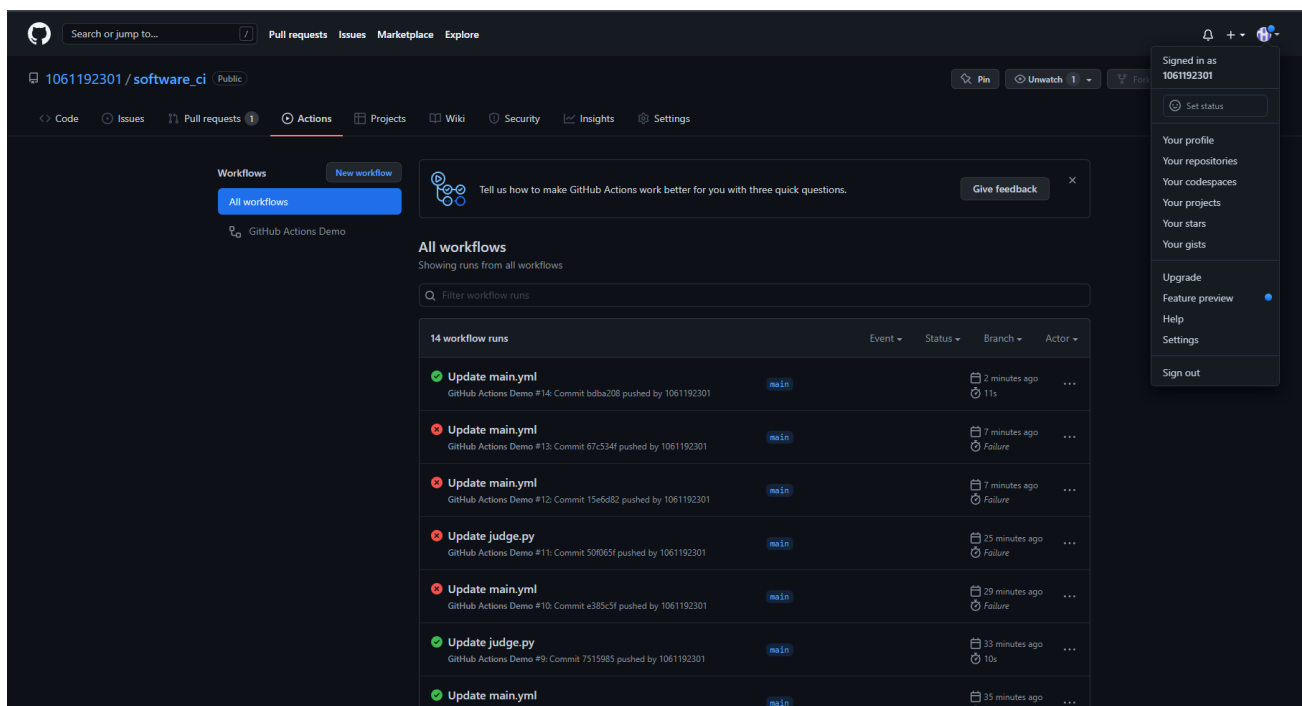
访问网址（校园网访问）：http://ruby-git.act.buaa.edu.cn:15324/users/sign_in

deploy截图



2、使用github的action运行python文件

代码仓库: https://github.com/1061192301/software_ci



CI/CD特点及对比

持续集成强调开发人员提交了新代码之后，立刻自动的进行构建、（单元）测试。这样以后，能够保证集成后的项目依然能够正常运行。CD实现了源代码跟踪、构建、测试、指标采集，版本管理等工作流程自动化、高效、可重复，可以大大减少工作量，提高效率。通过CI/CD可以加快迭代过程以满足需求变更、保证技术正确、缩短产品交付流程。

相比于Gitlab action，githubCI/CD支持费第三方的嵌入式CD，有自己的生态系统（<https://about.gitlab.com/partners/>），内置Kubernetes部署和监控管理，具有安全仪表盘、容器注册表等，具有自托管、安全协作的优点。

而Gitlab action通过使管道成为即插即用系统来带来其开源社区的强大功能，可以在其中引入其他人可能已经构建的已经可用的"操作"，一定程度上避免了重复性任务。