

Lab Manuals for  
***Software Construction***

**实验环境与提交指南**



School of Computer Science and Technology

Harbin Institute of Technology

Spring 2019

**目录**

[1 实验发布 1](#_Toc507220262)

[2 实验环境 1](#_Toc507220263)

[3 提交方式 1](#_Toc507220264)

[3.1 注册个人GitHub账号 1](#_Toc507220265)

[3.2 在GitHub Classroom中获取实验的private repositories 2](#_Toc507220266)

[3.3 本地Eclipse开发、Git管理、提交至GitHub仓库 6](#_Toc507220267)

[4 评分方式 6](#_Toc507220268)

# 实验发布

所有实验均在CMS上发布，请关注“作业/实验”区域所发布的信息（实验手册、提交截止日期）：

https://cms.hit.edu.cn/course/view.php?id=343

# 实验环境

1. 请阅读<http://web.mit.edu/6.031/www/fa18/getting-started/>，按该页面列出的指南，在本地机器安装相应的开发环境（JDK、Eclipse、Git）并熟练掌握它们的配置过程。
2. 可以从<http://web.mit.edu/6.031/www/sp17/getting-started/eclipse-faq/>获取更多的Eclipse帮助。
3. 阅读<http://web.mit.edu/6.031/www/fa18/classes/02-basic-java/>了解Java的基本编程特性。
4. 关于Git的学习手册：<https://git-scm.com/book/en/v2>（英文版）、<https://git-scm.com/book/zh/v2>（中文版）。可使用<https://www.shiyanlou.com/courses/4>提供的在线实验环境进行Git练习。
5. 阅读<http://web.mit.edu/6.005/www/fa16/psets/ps0/#unit_testing>，了解单元测试和JUnit工具。
6. 阅读<https://github.com/junit-team/junit4/wiki/Download-and-Install>，并在自己的Eclipse IDE中安装配置JUnit。
7. 阅读<https://github.com/junit-team/junit4/wiki/Getting-started>，了解如何使用JUnit为Java程序编写测试代码并执行测试。

# 提交方式

## 注册个人GitHub账号

在GitHub上注册个人账号，user name请使用自己的学号（例如11703005XX，共10位数，不要遗漏了年级17前面的“1”）。如果之前已有GitHub账号，请修改user name为自己的学号，或者重新注册新账号。

若不遵循该规则，GitHub上为你创建的private repositories将无法被TA发现，因而无法得到评判和分数（TA通过10位学号查询你的仓库）。

## 在GitHub Classroom中获取实验的private repositories

GitHub Classroom是GitHub为学校提供的教学环境，教师在这里布置作业，为每个学生创建私有仓库，学生在仓库里提交代码，仓库对教师和TA可见。



在实验手册中获取每次实验的GitHub Classroom URL地址，例如：

https://classroom.github.com/a/z99eb2os

访问该地址，用个人GitHub账号登录，同意授权GitHub Classroom。第一次授权之后，后续实验就无需再授权。

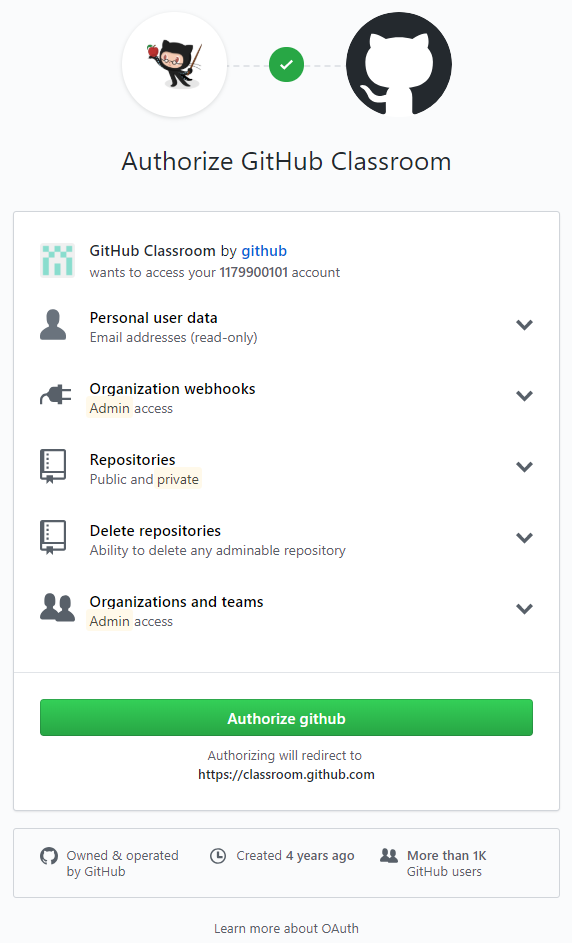


图1

进入之后，出现以下界面，请选择你的学号，建立该学号与你的GitHub账号的关联。请不要使用Skip，若在列表中找不到你的学号，请第一时间与教师联系。第一次绑定学号之后，后续实验无需再次绑定。

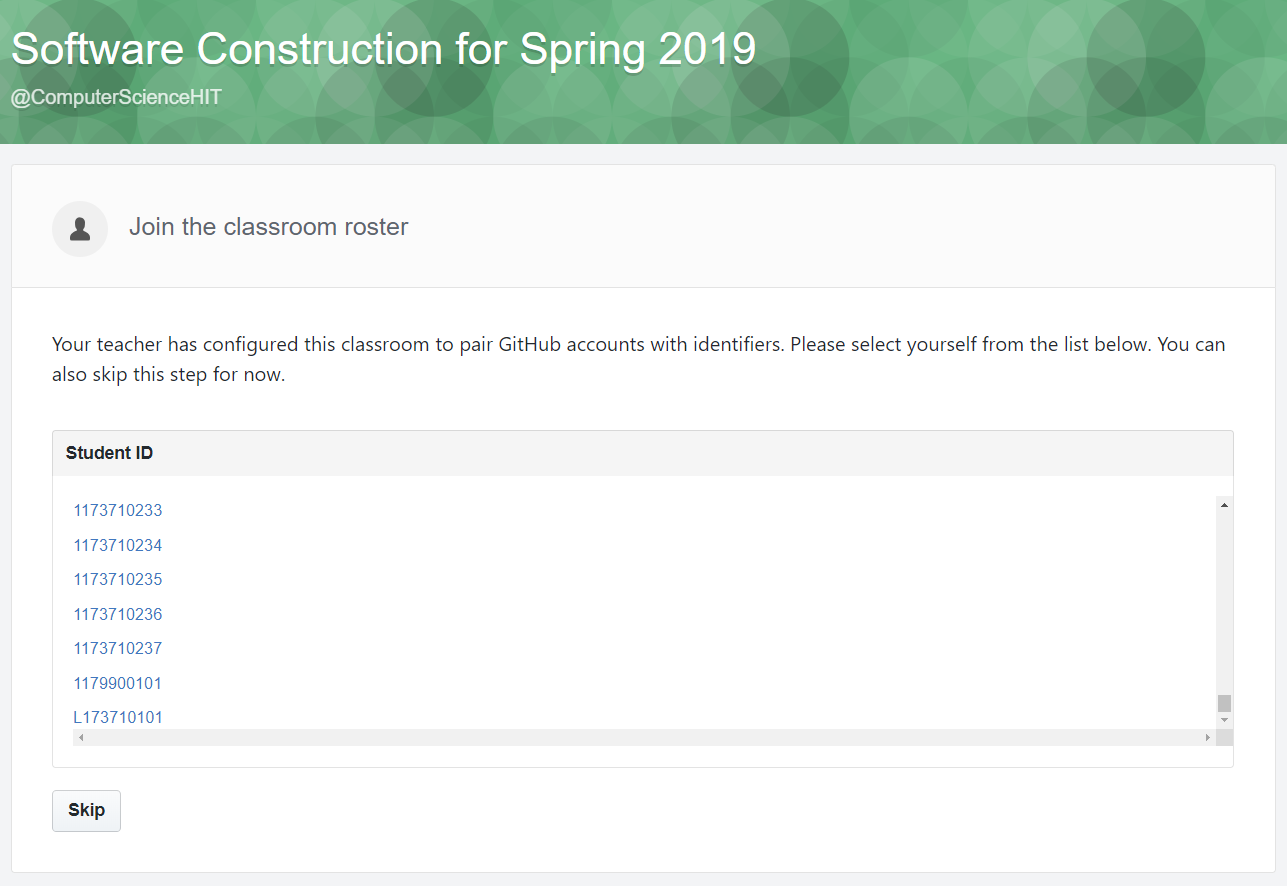


图2

之后，进入图3界面，点击“Accept this assignment”，GitHub classroom服务为你自动创建了用于提交此实验的Git repository（仓库名：Labx-1179900101，x为实验序号1~6，后面为你的学号）。进入之后，即可获得该仓库的信息（见图4）。

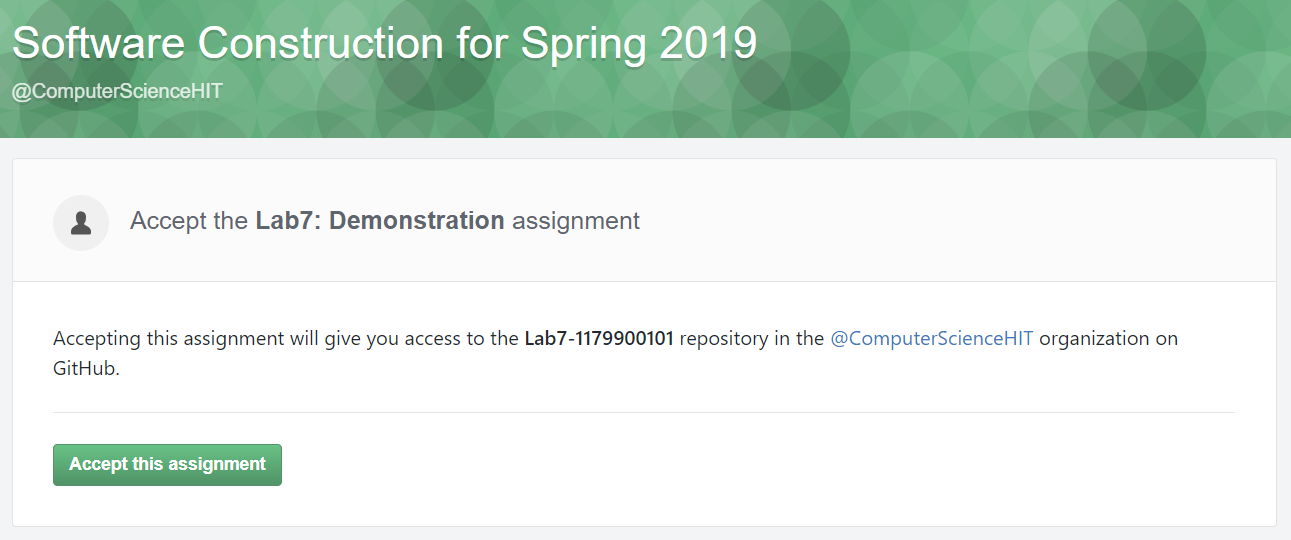


图3

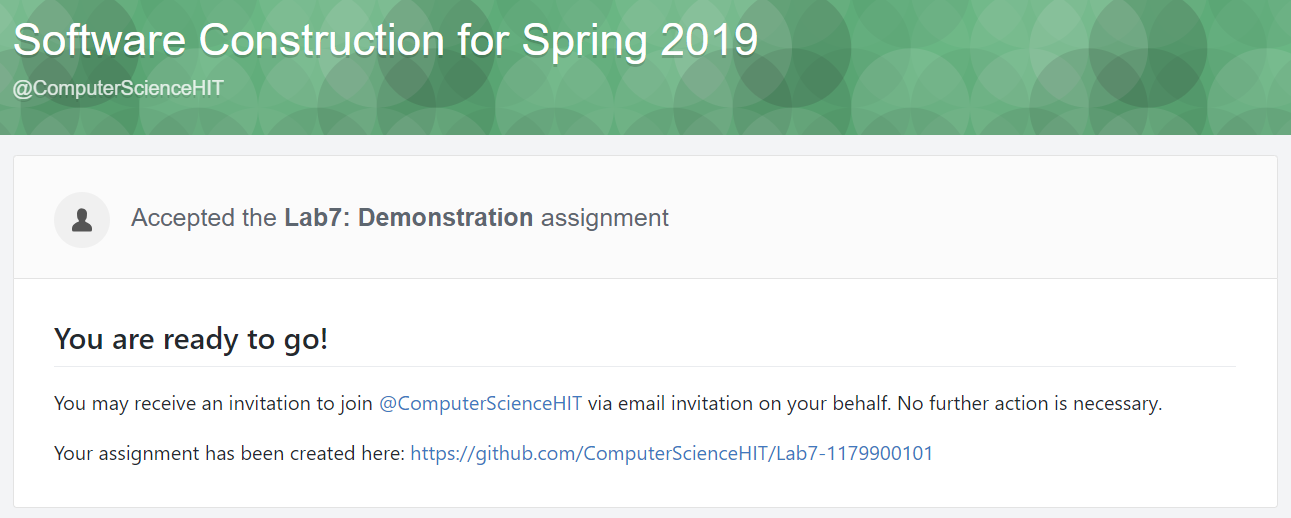


图4

之后，你就可以进入该仓库。最初仓库是空的。

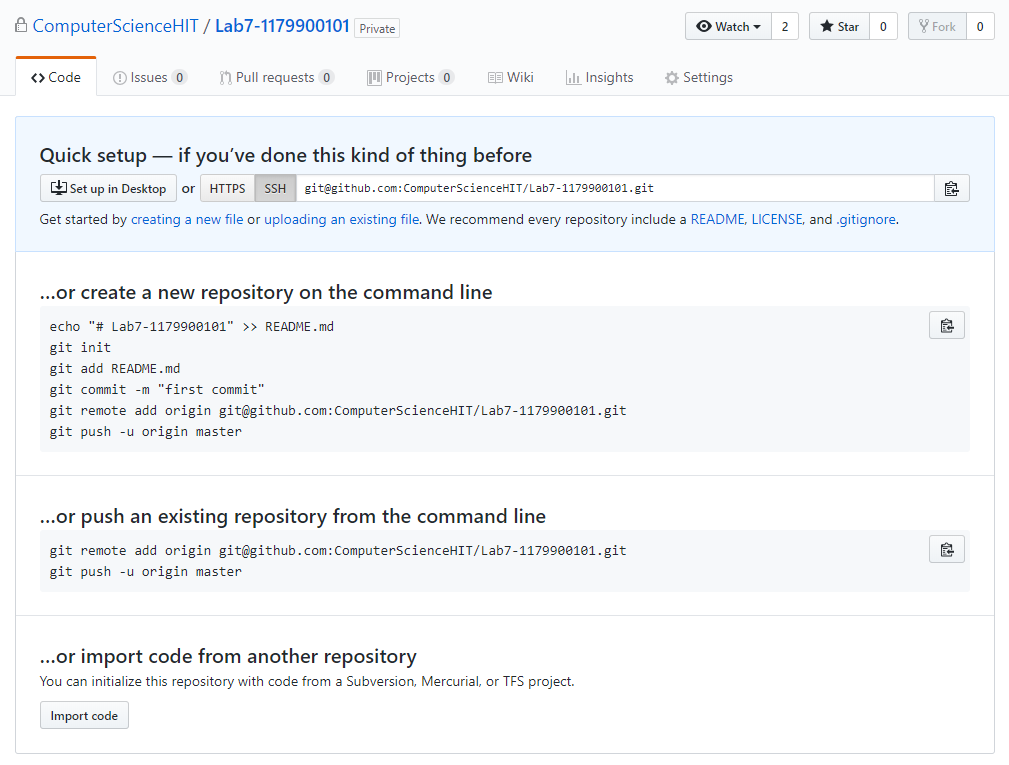


图5

你对该仓库有完全的控制权，与自己在GitHub上创建的仓库完全一样。你可以在本次Git进行远程提交，本地Git连接该仓库可使用https或SSH的方式，具体配置请自行探索。

注意：该仓库是private的，即仅对自己和教师可见。请坚决不要将各实验的代码发布到public的GitHub仓库中。如果因为将自己的代码发布到public仓库导致被其他人抄袭借鉴，那么双方均无成绩。

## 本地Eclipse开发、Git管理、提交至GitHub仓库

按照实验要求，在本地机器使用Eclipse、Git进行开发，本地Git仓库保存所有提交历史。在截止日期前推送至上述GitHub仓库。

各实验的仓库目录结构如下：

Labx-1179900101 根目录

src java文件子目录，内部可以包含子目录

test JUnit测试程序子目录，内部可以包含子目录

lib 程序所使用的所有外部库文件，内部不分子目录

doc 实验报告

… 你自己设置的其他目录

注意：

1. 请使用JDK 8及以上（JDK 9、JDK10或JDK11）。
2. 请遵循该目录规范，若不满足，TA评分使用的持续集成工具无法构建你的实验。
3. 在Eclipse中通过配置加入项目的各种外部库（通常是.jar形式），本地构建时一般是通过classpath引入到项目中，但在提交到GitHub仓库之前，请将代码中所有用到的jar文件拷贝至项目的lib目录，否则在GitHub上构建时无法成功。
4. 在提交至GitHub仓库前，请将实验代码从Eclipse环境脱离开来，建议你自行使用JDK、Ant (<http://ant.apache.org>)、Maven (http://maven.apache. org)、Gradle (<https://gradle.org>)等工具进行build，或者在提交至GitHub仓库之后使用Travis-CI (<https://travis-ci.org>)进行在线build。如果因为缺少某些库文件导致你的程序无法运行，TA不再为其评分。
5. 提交到GitHub仓库时，构建程序会读取master分支。所以如果你在其他分支上开发，提交前请务必将最新修改合并至master分支。

# 评分方式

在各实验的截止日期之后，教师可能会使用某些工具对你的代码进行build、执行JUnit测试用例、使用诸如CheckStyle和FindBugs等静态代码检查工具对代码规范进行检查，再通过人工测试的方式，对你的实验结果进行打分。

此外，TA阅读doc目录下的实验报告，对其进行人工打分。

注意：实验截止日期（周日夜里23:55）之后，GitHub仓库将不再允许提交。