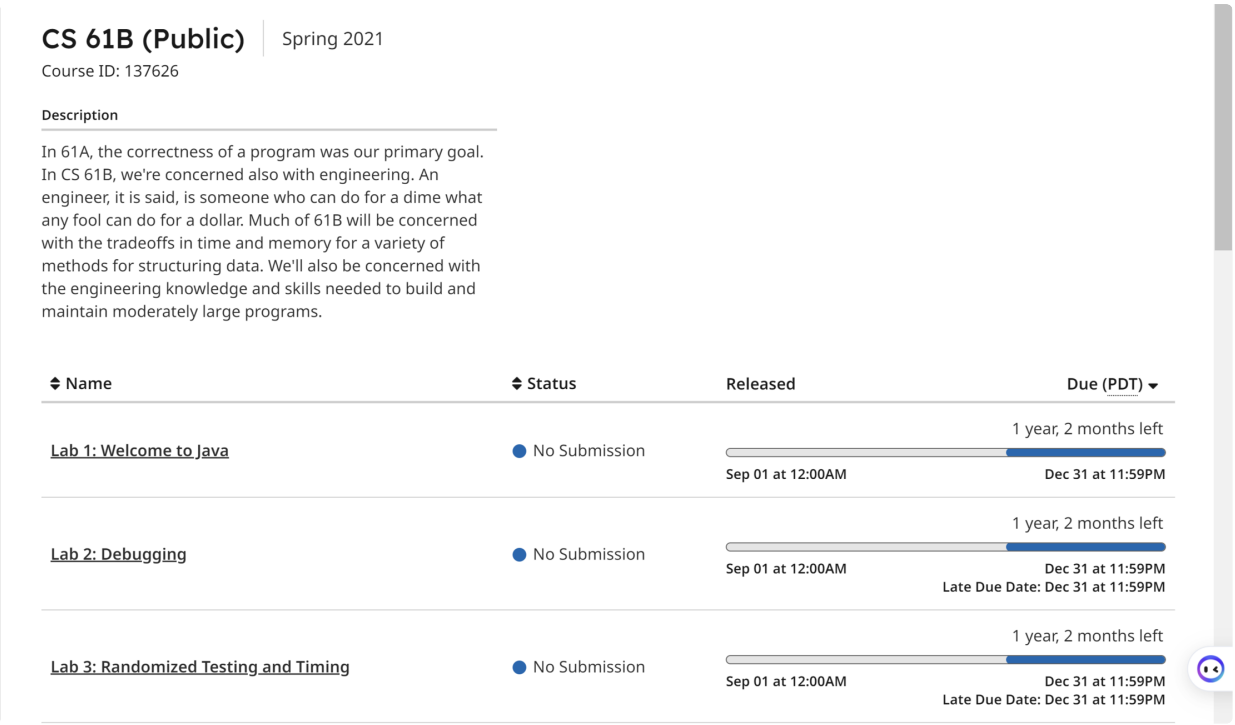
环境配置细节

step1

注册 gradescore，2021spring 邀请码：P5WVGW



CS 61B (Public) | Spring 2021

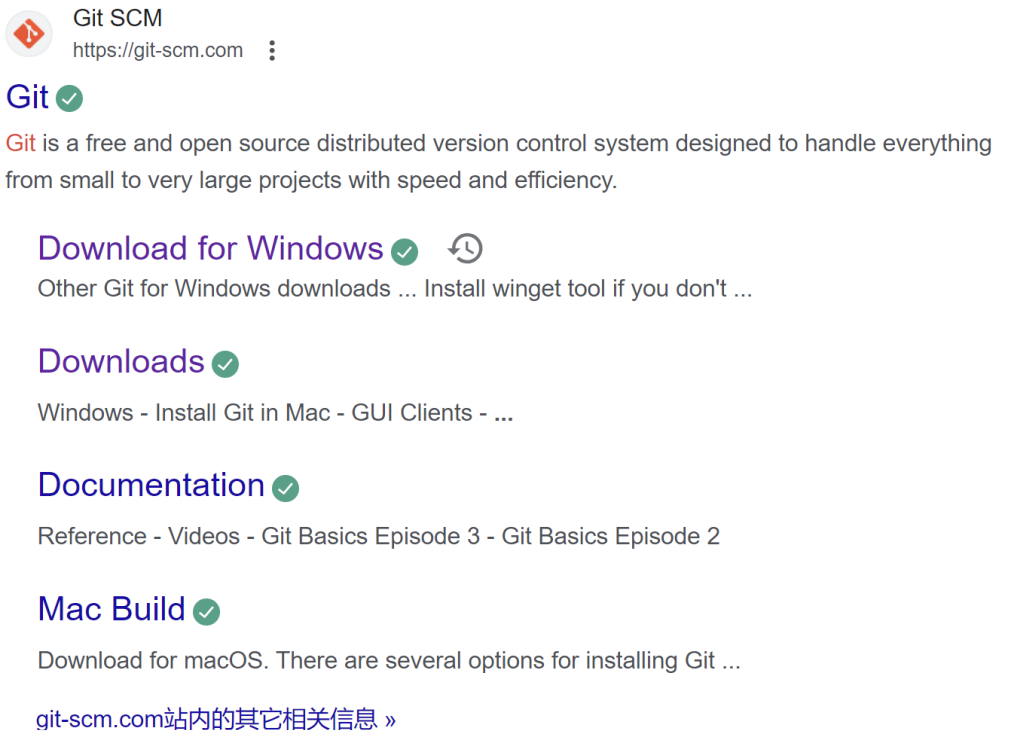
Course ID: 137626

Description

In 61A, the correctness of a program was our primary goal. In CS 61B, we're concerned also with engineering. An engineer, it is said, is someone who can do for a dime what any fool can do for a dollar. Much of 61B will be concerned with the tradeoffs in time and memory for a variety of methods for structuring data. We'll also be concerned with the engineering knowledge and skills needed to build and maintain moderately large programs.

Name	Status	Released	Due (PDT)
Lab 1: Welcome to Java	No Submission	Sep 01 at 12:00AM	1 year, 2 months left Dec 31 at 11:59PM
Lab 2: Debugging	No Submission	Sep 01 at 12:00AM	1 year, 2 months left Dec 31 at 11:59PM Late Due Date: Dec 31 at 11:59PM
Lab 3: Randomized Testing and Timing	No Submission	Sep 01 at 12:00AM	1 year, 2 months left Dec 31 at 11:59PM Late Due Date: Dec 31 at 11:59PM

step2 官网下载 git



Git SCM

https://git-scm.com

Git

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Download for Windows

Other Git for Windows downloads ... Install winget tool if you don't ...

Downloads

Windows - Install Git in Mac - GUI Clients - ...

Documentation

Reference - Videos - Git Basics Episode 3 - Git Basics Episode 2

Mac Build

Download for macOS. There are several options for installing Git ...

git-scm.com站内的其它相关信息 »

download 最新版本

Download for Windows

Click here to download the latest (2.42.0) 64-bit version of Git for Windows. This is the most

recent maintained build. It was released **about 2 months ago**, on 2023-08-30.

Other Git for Windows downloads

Standalone Installer
32-bit Git for Windows Setup.

64-bit Git for Windows Setup.

Portable ("thumbdrive edition")
32-bit Git for Windows Portable.

64-bit Git for Windows Portable.

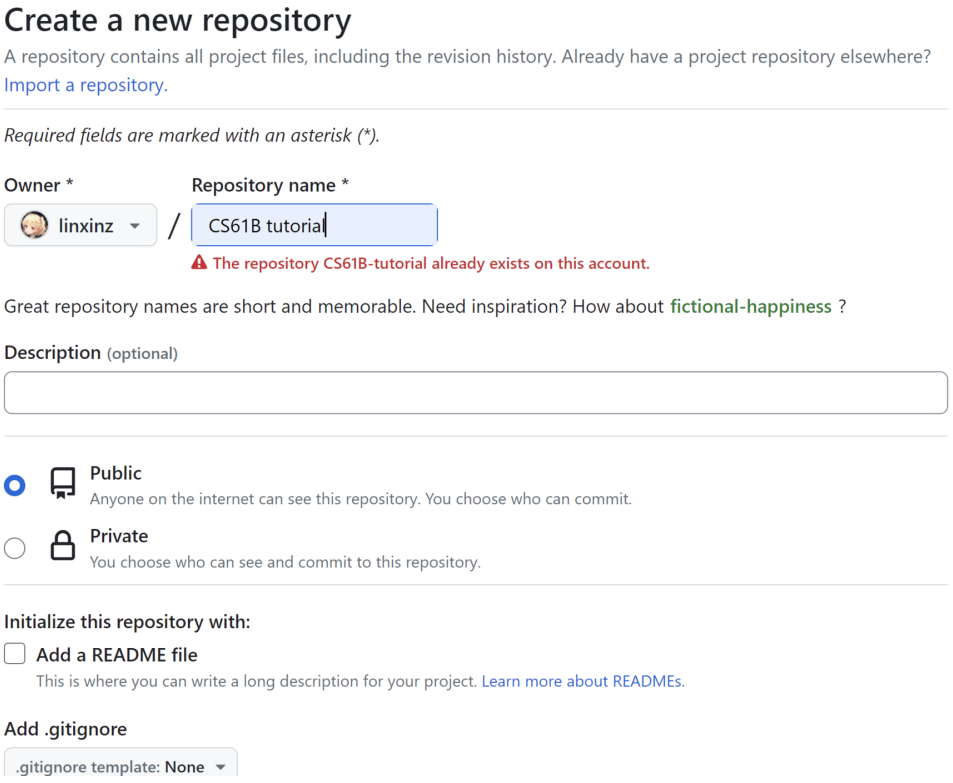
Using winget tool
Install winget tool if you don't already have it, then type this command in command prompt or Powershell.

下载后全选默认就行了。

安装完成后，打开 git bash 设置使用者的用户名和邮箱

```Git
git config --global user.name "Your Name"https://github.com/Berkele
git config --global user.email "email@example.com"
```

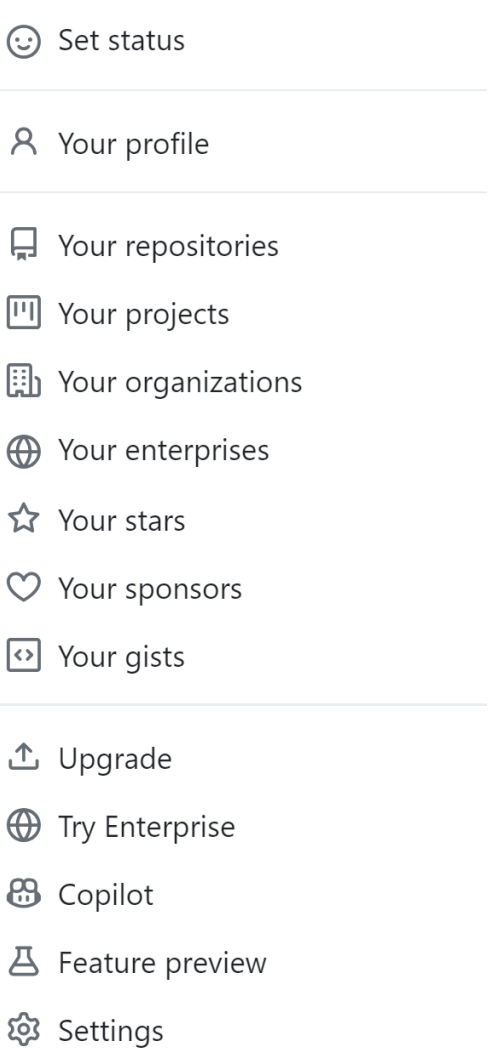
step3 GitHub 上新建一个仓库

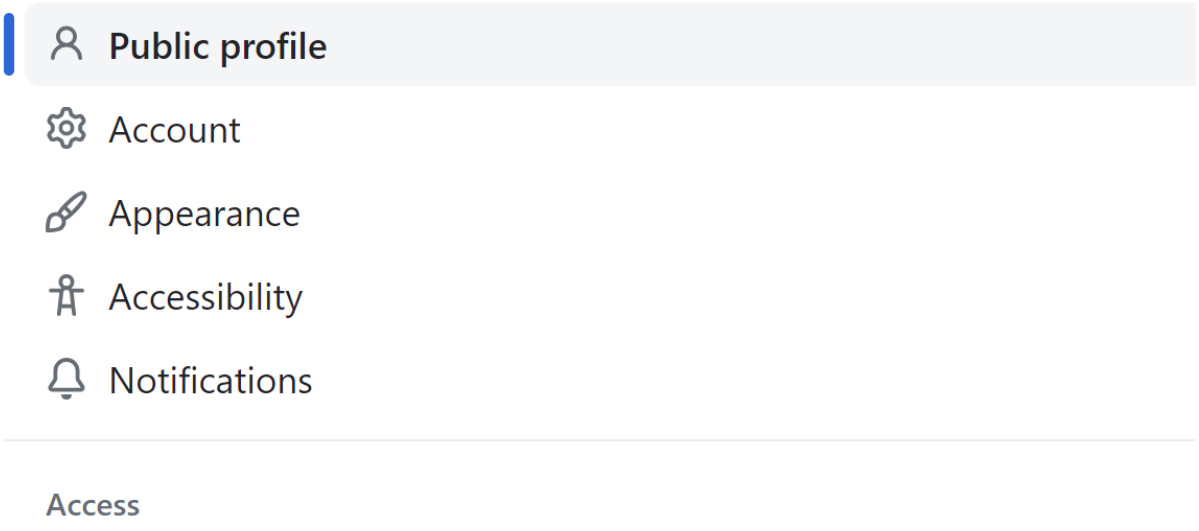


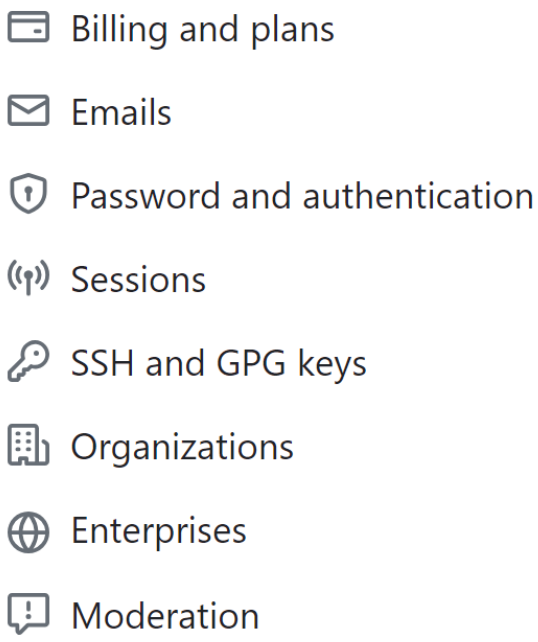
Choose which files not to track from a list of templates. Learn more about ignoring files.

step4 关联 git 和 GitHub

首先打开 GitHub setting







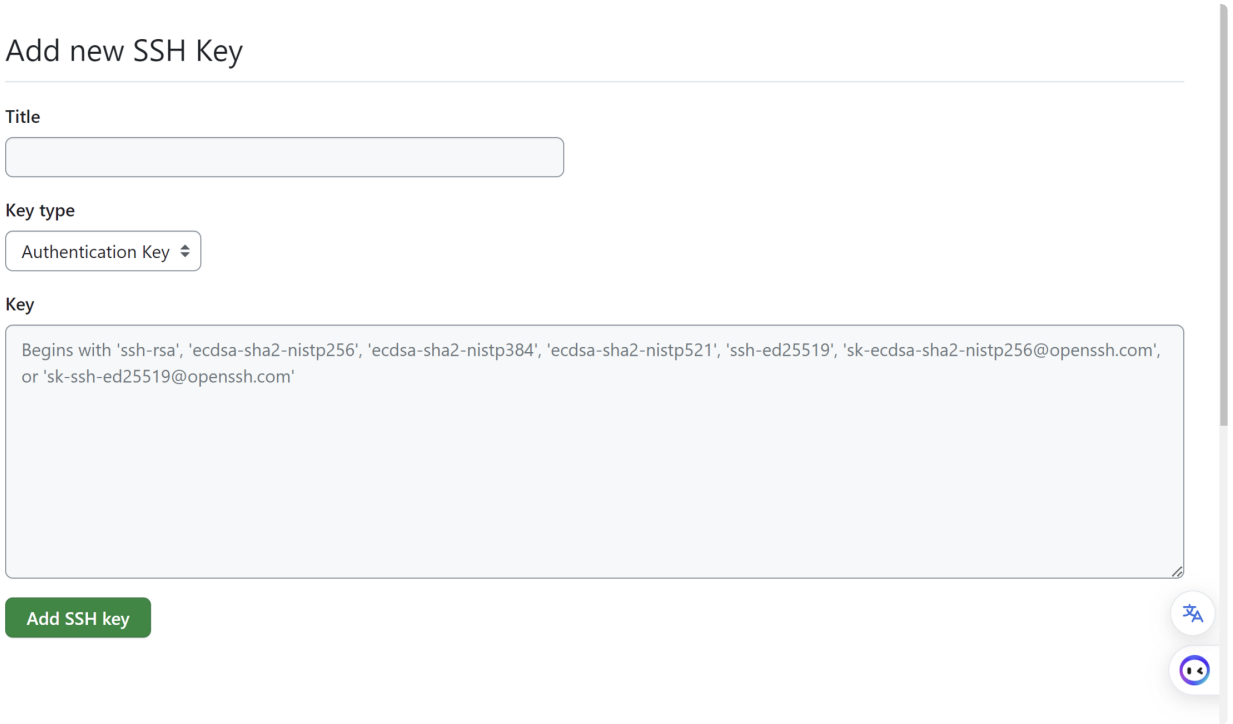
点击 SSH AND GPG authentication

New SSH KEY

然后打开终端，输入

```Git
ssh-keygen -t rsa -C "xxx@xxx.com"
```

然后在指定文件夹找到 id_rsa.pub 文件，记事本打开，复制里面的内容到刚刚的 ssh 设置界面



如果找不到指定文件的话，先打开终端

```
纯文本
cd C:\Users\你的用户名

//创建一个.ssh目录
mkdir .ssh

//切换到.ssh目录
cd .ssh


//再次运行
ssh-keygen -t rsa -C "邮箱" -f id_rsa
```

然后在指定目录下就可以找到 id_rsa.pub 文件了，将里面的内容复制到 add ssh key 里面，就可以连接 git 与 GitHub 了。

Step5 将仓库克隆到本地

在你建的本地仓库的目录下右键打开 git bash

```
纯文本
git clone 新建的仓库的url
```

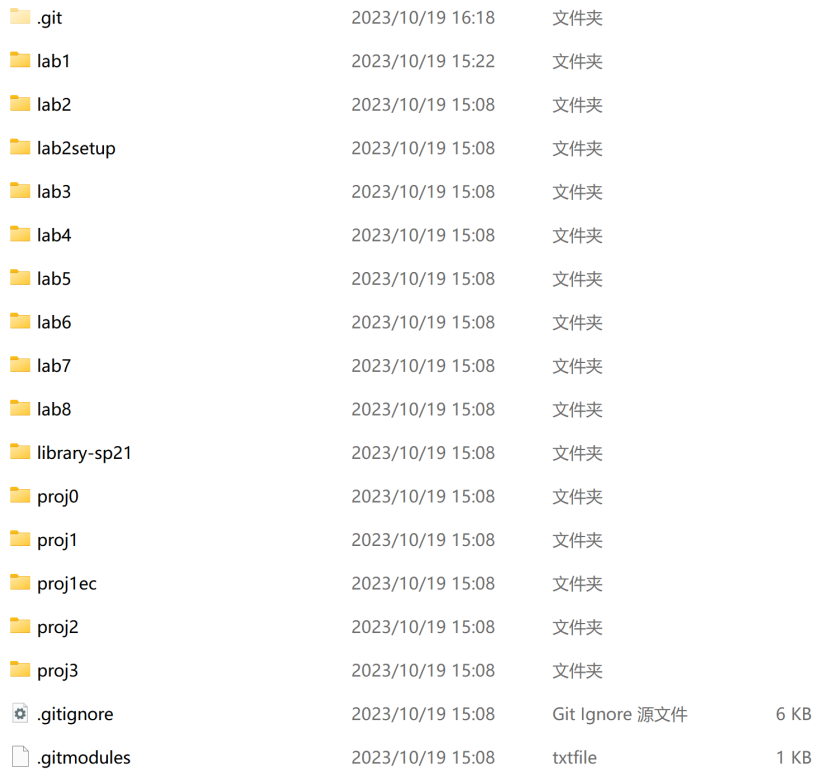
Step6 添加官方的远程仓库

```
纯文本
git remote add skeleton https://github.com/Berkeley-CS61B/skeleton-s
```

Step7 本地仓库拉取远程仓库的数据

```
纯文本
git pull skeleton master
```

现在本地就存储了我们实验需要的代码了



在 intellij 中打开项目就能运行指定的项目了

Step8 作业提交

假设你已经完成了指定的 lab 或者 project，现在需要提交

打开 git bash，输入

纯文本

```
git status
//查看当前的状态
```

可以看到红色部分是刚刚修改的代码

执行：

纯文本

```
git add lab1/*
```

纯文本

```
git commit -m"一些备注信息"
```

纯文本

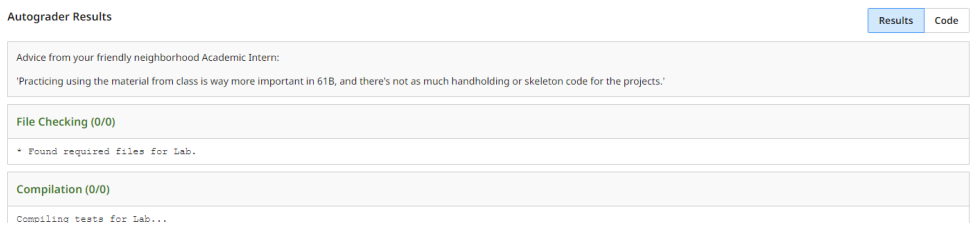
```
git push origin maste
```

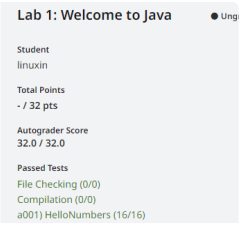
```
86157@laptop-linxin MINGW64 /d/1SWUfile/self-study/CS61B (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   lab1/Collatz.java

no changes added to commit (use "git add" and/or "git commit -a")

86157@laptop-linxin MINGW64 /d/1SWUfile/self-study/CS61B (master)
$ git add lab1/*
The following paths are ignored by one of your .gitignore files:
lab1/lab1.iml
lab1/out
hint: Use -f if you really want to add them.
hint: Turn this message off by running
hint: "git config advice.addIgnoredFile false"

86157@laptop-linxin MINGW64 /d/1SWUfile/self-study/CS61B (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   lab1/Collatz.java


86157@laptop-linxin MINGW64 /d/1SWUfile/self-study/CS61B (master)
$ git commit -m"第一次提交"
[master 4aa64af] 第一次提交
 1 file changed, 1 insertion(+), 1 deletion(-)

86157@laptop-linxin MINGW64 /d/1SWUfile/self-study/CS61B (master)
$
```

lab1	第一次提交	3 hours ago
lab2	Fixed relative path for pom.xml	2 years ago
lab2setup	Fixed relative path for pom.xml	2 years ago
lab3	Fixed typo in SLList in Lab 3	3 years ago
lab4	Added Lab 4 Starter Files	3 years ago
lab5	Lab 5 Starter Files	2 years ago
lab6	Added real test for in lab 6 (ungraded test)	2 years ago
lab7	Lab 7 Starter Code	2 years ago
lab8	Lab 8 Skeleton Code	2 years ago
library-sp21 @ c427536	add library-sp21 submodule	2 years ago
proj0	added javalib for proj0	2 years ago
proj1	Project 1: shortened assert statement	3 years ago

打开 GitHub 我们创建的库，可以看到 lab1 已经成功更新

然后打开 gradescope 提交即可





success.

a001) HelloNumbers (16/16)

```
Your HelloNumbers output: 0 1 3 6 10 15 21 28 36 45
Expected output:          0 1 3 6 10 15 21 28 36 45
```

b001) Collatz (16/16)

```
Your Collatz output: 5 16 8 4 2 1
Expected output:     5 16 8 4 2 1
```

b001) Collatz (16/16)

大功告成！