软件工程Lab7

11510365 薛毅恒

Assignment 1

- Write a report on comparing build management tools for Java, including Gradle, Maven and Ant.

`Gradle` , `Maven` , `Ant` 这是三种不同的构建器，首先需要明确什么是java的构建。

> 在编程的过程中，与编程无关的管理工作包括下载依赖、编译源码、单元测试、项目部署等操作，除了小型项目中的手动操作外，在大型项目中需要通过构建工具帮助我们实现一系列 `项目管理` 、 `测试` 和 `部署` 等操作。

最开始，只有**Make**一种构建工具，后来发展为**GNU Make**, Java的自动构建工具主要是通过 `makefile` 文件进行，其中包括了自动构建的指令。从 `Ant` 到 `Maven` 再到 `Gradle` 是不断更新、进步的。

- **Ant**是2000年发布的，使用*Java*编写了核心，采用*XML*作为构建脚本，这样的好处是可以在任何环境下运行构建。缺点是*XML*定义构建脚本，导致脚本复杂、臃肿， `Ant` 自身也没有为项目构建提供指导，导致每个build的脚本都不一样，所以在不同项目中都要去熟悉脚本内容，没有提供在 `Ant` 环境中的依赖管理工具。

  > build.xml

- **Maven**团队也意识到了 `Ant` 的这些缺陷，在2004年发布这个版本，采用了标准的项目布局、统一的生命周期采用约定由于配置的思想，减少构建脚本需要编写的内容，使用人数联系紧密所以有了活跃的社区，方便找到合适的插件，也有强大的依赖管理工具， `Ant` 需要将所有task都列出来，而 `Maven` 可以依靠约定并提供现成的课调用的目标，可以从网络上自动下载依赖。缺点是使用*XML*作为构建脚本，采用了默认的结构和生命周期，限制过多，编写插件、扩展内容很麻烦。

  > pom.xml

- **Gradle**便结合了 `Ant` 和 `Maven` 的有点，程序员在使用过程中发现问题，不断更新。基于*Groovy*和*DSL(Domain Specific Languages)*，提供声明式的构建语言，采用标准的项目布局，但完全可配置、可修改，也有更多的插件，提供默认的构建生命周期，也可以自己定义任务，单独运行任务，定义任务间的依赖，强大的依赖管理工具，约定好、灵活性高，与 `Maven` 和 `Ant` 相结合、与Ant兼容，可以重用 `Ant` 的任务，多种实现插件的方式，强大的官方插件库，支持从 `Ant` 或者 `Maven` 的逐步迁移，也可以在多种平台上运行。

  > build.gradle

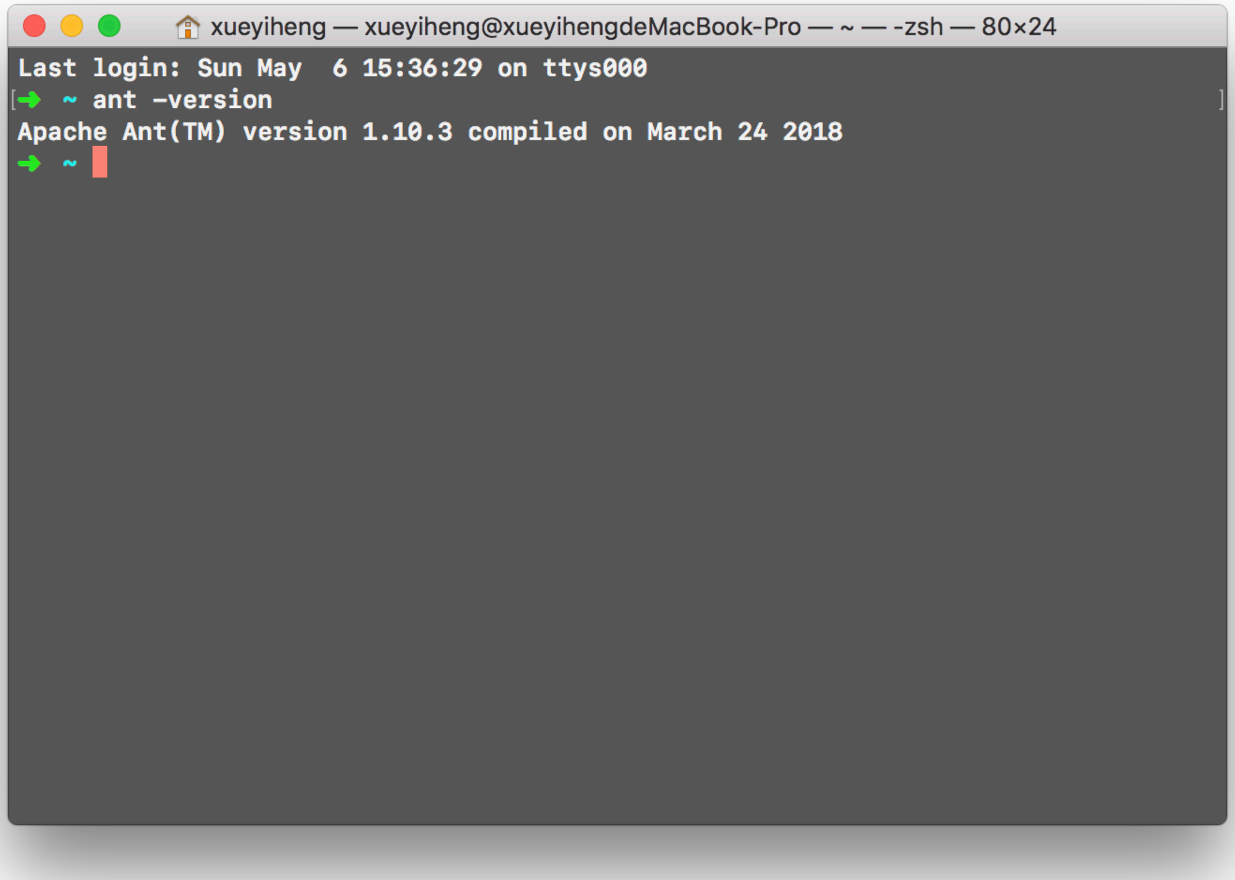
`Ant` 是最清晰直观的，*XML*文件中很详细、清晰的写了配置文件，可以理解它的作用，但是 `Ant` 文件很容易变得很复杂。 `Gradle` 有很多比合适的插件，就像是构建工具里的 `Python` 一样，不需要理解所有内容也可以使用的很顺利。

Assignment 2

- Complete this tutorials

  > 在 `gradle` 的一系列操作中，要在complete文件夹中进行，第6步一定要完成之后再进行最后一步的 `run` 操作， `build.gradle` 文件及build后的 `build` 文件夹的路径是 `gs-gradle/complete/`

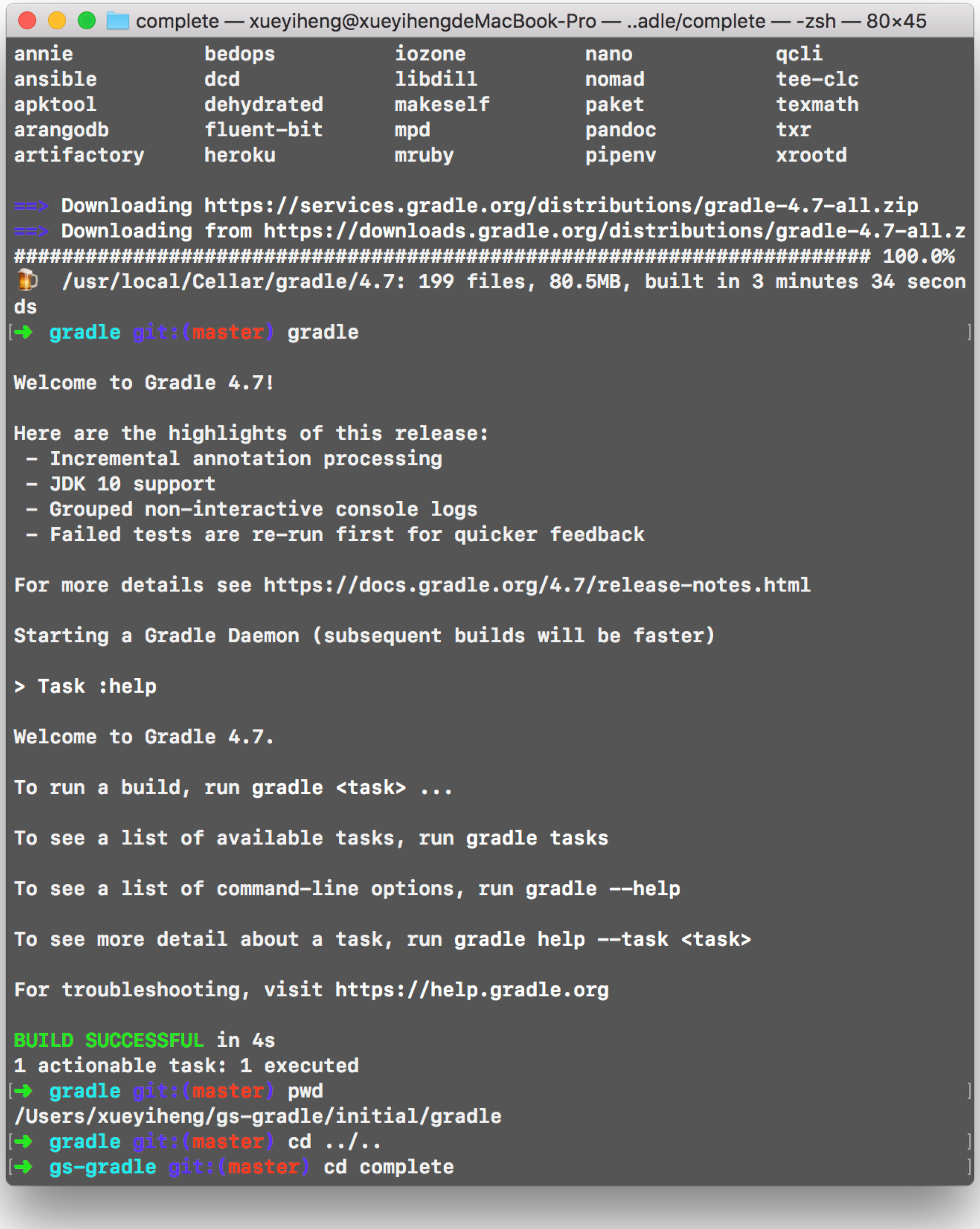
1. Install ant



xueyiheng — xueyiheng@xueyihengdeMacBook-Pro — ~ — -zsh — 80×24

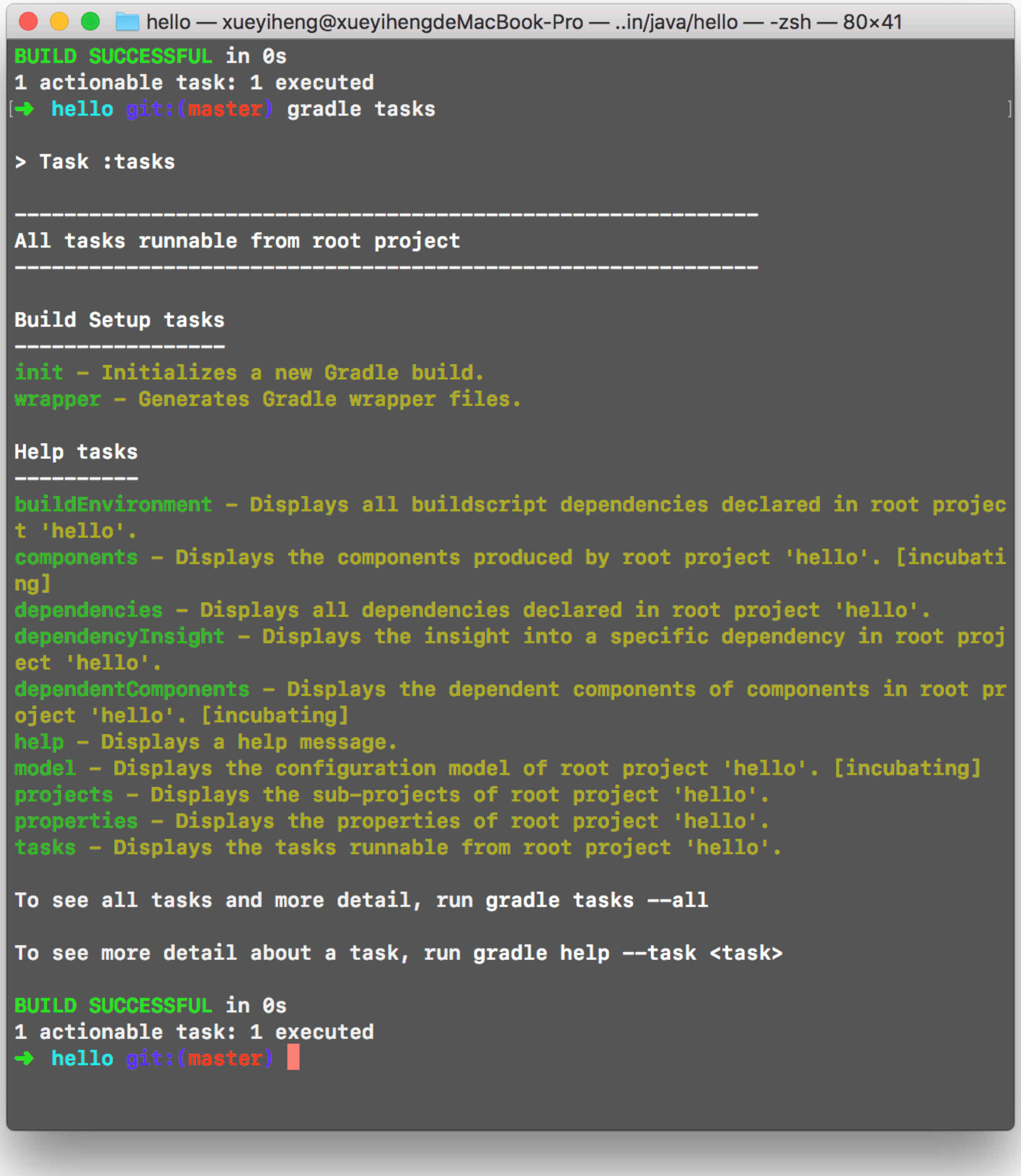
```
Last login: Sun May  6 15:36:29 on ttys000
➜  ~ ant -version
Apache Ant(TM) version 1.10.3 compiled on March 24 2018
➜  ~ 
```

2. Install gradle


complete — xueyiheng@xueyihengdeMacBook-Pro — ..adle/complete — -zsh — 80×45

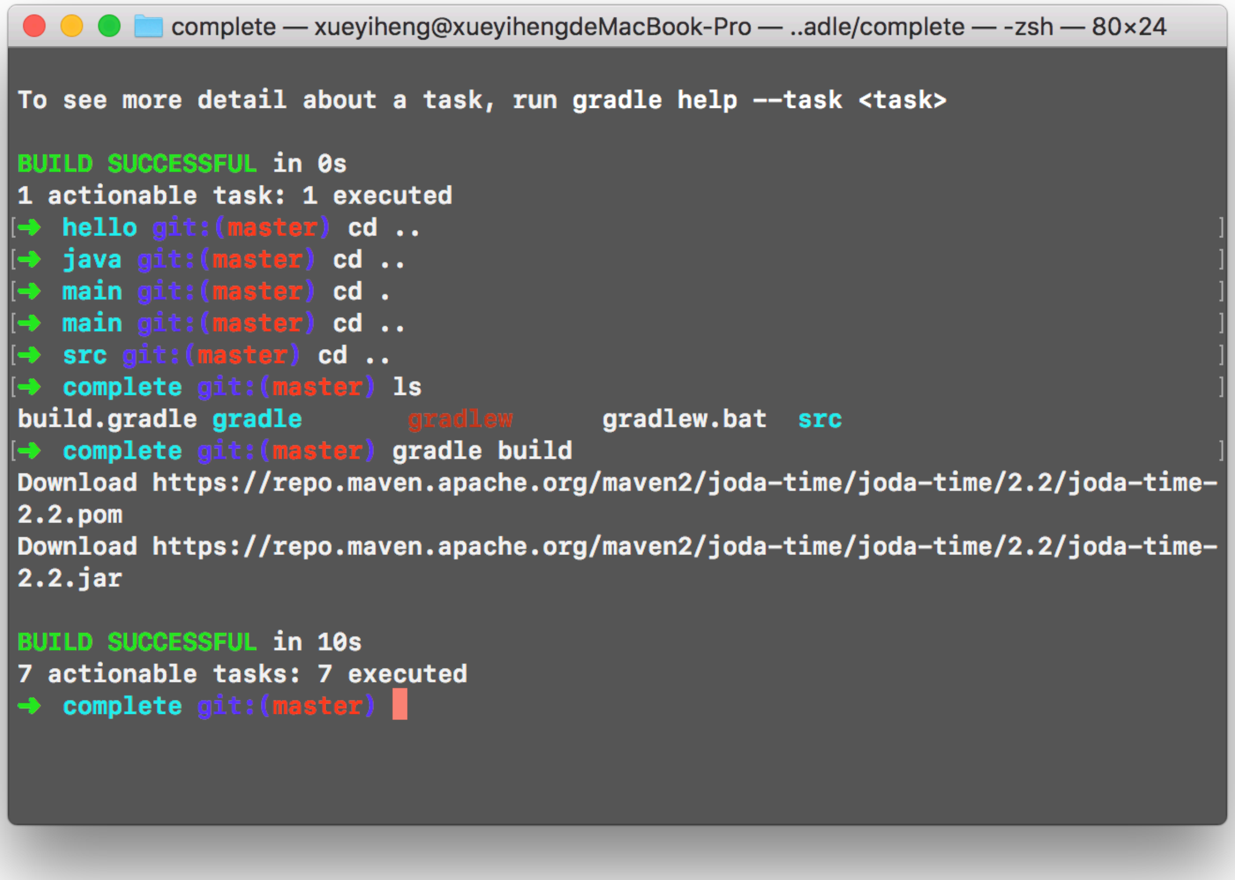
```
annie           bedops          iozone          nano            qcli
ansible         dcd             libdill         nomad           tee-clc
apktool         dehydrated      makeself        paket           texmath
arangodb        fluent-bit      mpd             pandoc          txr
artifactory     heroku          mruby           pipenv          xrootd

==> Downloading https://services.gradle.org/distributions/gradle-4.7-all.zip
==> Downloading from https://downloads.gradle.org/distributions/gradle-4.7-all.z
######################################################################## 100.0%
🍺  /usr/local/Cellar/gradle/4.7: 199 files, 80.5MB, built in 3 minutes 34 secon
ds
➜  gradle git:(master) gradle

Welcome to Gradle 4.7!

Here are the highlights of this release:
 - Incremental annotation processing
 - JDK 10 support
 - Grouped non-interactive console logs
 - Failed tests are re-run first for quicker feedback

For more details see https://docs.gradle.org/4.7/release-notes.html

Starting a Gradle Daemon (subsequent builds will be faster)

> Task :help

Welcome to Gradle 4.7.

To run a build, run gradle <task> ...

To see a list of available tasks, run gradle tasks

To see a list of command-line options, run gradle --help

To see more detail about a task, run gradle help --task <task>

For troubleshooting, visit https://help.gradle.org

BUILD SUCCESSFUL in 4s
1 actionable task: 1 executed
➜  gradle git:(master) pwd
/Users/xueyiheng/gs-gradle/initial/gradle
➜  gradle git:(master) cd ../..
➜  gs-gradle git:(master) cd complete
```

3. Gradle tasks


hello — xueyiheng@xueyihengdeMacBook-Pro — ..in/java/hello — -zsh — 80×41

```
BUILD SUCCESSFUL in 0s
1 actionable task: 1 executed
➜ hello git:(master) gradle tasks

> Task :tasks

------------------------------------------------------------
All tasks runnable from root project
------------------------------------------------------------

Build Setup tasks
-----------------
init - Initializes a new Gradle build.
wrapper - Generates Gradle wrapper files.

Help tasks
----------
buildEnvironment - Displays all buildscript dependencies declared in root projec
t 'hello'.
components - Displays the components produced by root project 'hello'. [incubati
ng]
dependencies - Displays all dependencies declared in root project 'hello'.
dependencyInsight - Displays the insight into a specific dependency in root proj
ect 'hello'.
dependentComponents - Displays the dependent components of components in root pr
oject 'hello'. [incubating]
help - Displays a help message.
model - Displays the configuration model of root project 'hello'. [incubating]
projects - Displays the sub-projects of root project 'hello'.
properties - Displays the properties of root project 'hello'.
tasks - Displays the tasks runnable from root project 'hello'.

To see all tasks and more detail, run gradle tasks --all

To see more detail about a task, run gradle help --task <task>

BUILD SUCCESSFUL in 0s
1 actionable task: 1 executed
➜ hello git:(master)
```

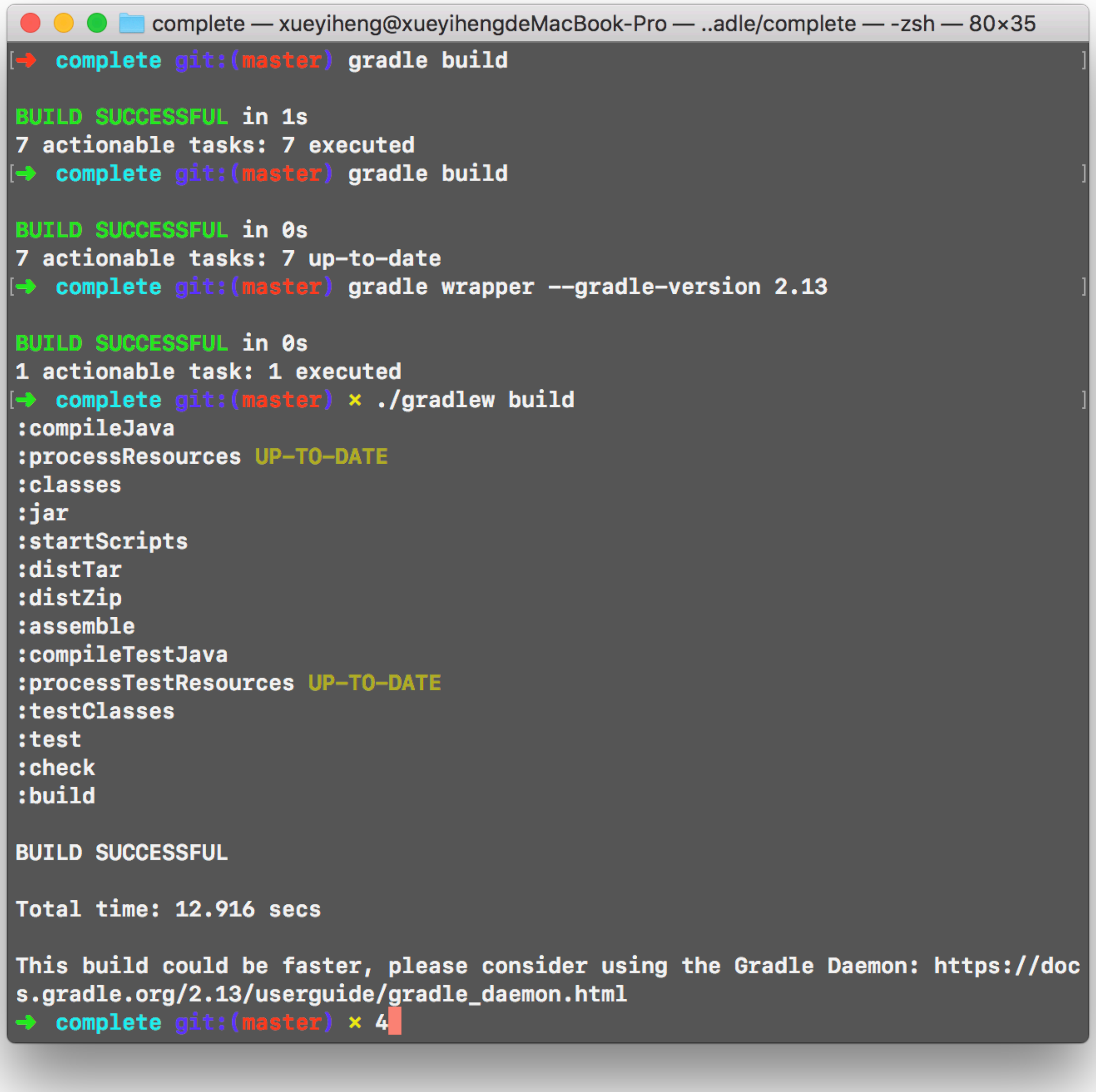
4. Gradle build



complete — xueyiheng@xueyihengdeMacBook-Pro — ..adle/complete — -zsh — 80×24

```
To see more detail about a task, run gradle help --task <task>

BUILD SUCCESSFUL in 0s
1 actionable task: 1 executed
➜ hello git:(master) cd ..
➜ java git:(master) cd ..
➜ main git:(master) cd .
➜ main git:(master) cd ..
➜ src git:(master) cd ..
➜ complete git:(master) ls
build.gradle gradle       gradlew      gradlew.bat  src
➜ complete git:(master) gradle build
Download https://repo.maven.apache.org/maven2/joda-time/joda-time/2.2/joda-time-
2.2.pom
Download https://repo.maven.apache.org/maven2/joda-time/joda-time/2.2/joda-time-
2.2.jar

BUILD SUCCESSFUL in 10s
7 actionable tasks: 7 executed
➜ complete git:(master)
```

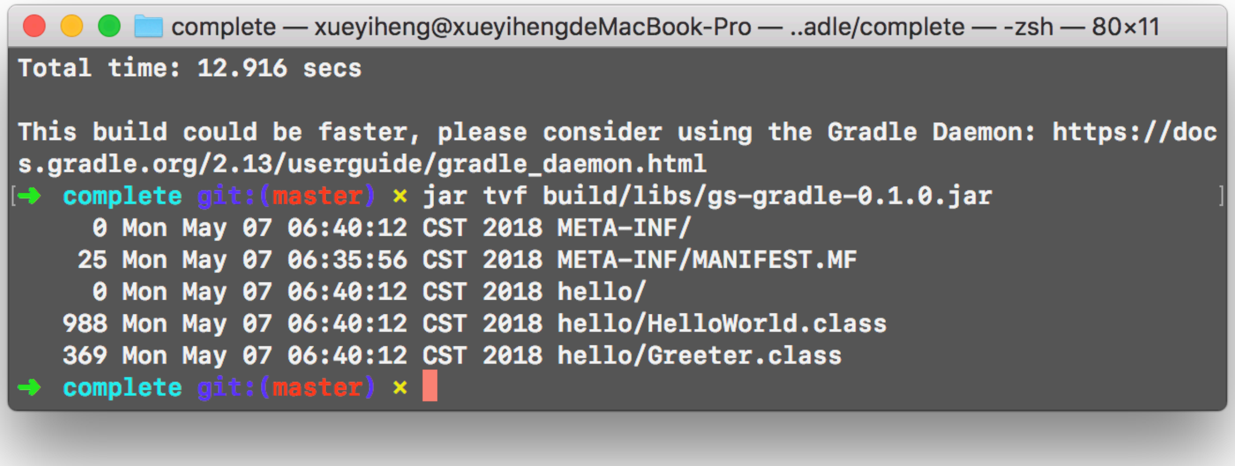
5. `./gradlew_build`



complete — xueyiheng@xueyihengdeMacBook-Pro — ..adle/complete — -zsh — 80×35

```
[➜  complete git:(master) gradle build                                         ]

BUILD SUCCESSFUL in 1s
7 actionable tasks: 7 executed
[➜  complete git:(master) gradle build                                         ]

BUILD SUCCESSFUL in 0s
7 actionable tasks: 7 up-to-date
[➜  complete git:(master) gradle wrapper --gradle-version 2.13                 ]

BUILD SUCCESSFUL in 0s
1 actionable task: 1 executed
[➜  complete git:(master) ✗ ./gradlew build                                    ]
:compileJava
:processResources UP-TO-DATE
:classes
:jar
:startScripts
:distTar
:distZip
:assemble
:compileTestJava
:processTestResources UP-TO-DATE
:testClasses
:test
:check
:build

BUILD SUCCESSFUL

Total time: 12.916 secs

This build could be faster, please consider using the Gradle Daemon: https://doc
s.gradle.org/2.13/userguide/gradle_daemon.html
➜  complete git:(master) ✗ 4
```

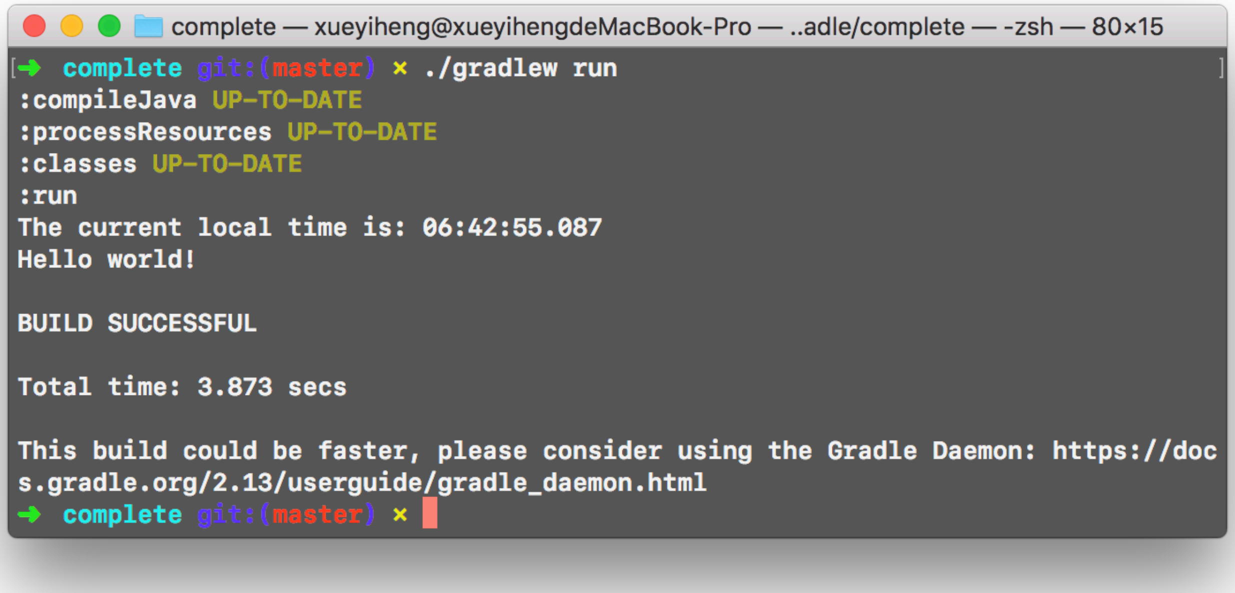
6. `jar tvf build/libs/gs-gradle-0.1.0.jar`



complete — xueyiheng@xueyihengdeMacBook-Pro — ..adle/complete — -zsh — 80×11

```
Total time: 12.916 secs

This build could be faster, please consider using the Gradle Daemon: https://doc
s.gradle.org/2.13/userguide/gradle_daemon.html
➜ complete git:(master) ✗ jar tvf build/libs/gs-gradle-0.1.0.jar
     0 Mon May 07 06:40:12 CST 2018 META-INF/
    25 Mon May 07 06:35:56 CST 2018 META-INF/MANIFEST.MF
     0 Mon May 07 06:40:12 CST 2018 hello/
   988 Mon May 07 06:40:12 CST 2018 hello/HelloWorld.class
   369 Mon May 07 06:40:12 CST 2018 hello/Greeter.class
➜ complete git:(master) ✗
```

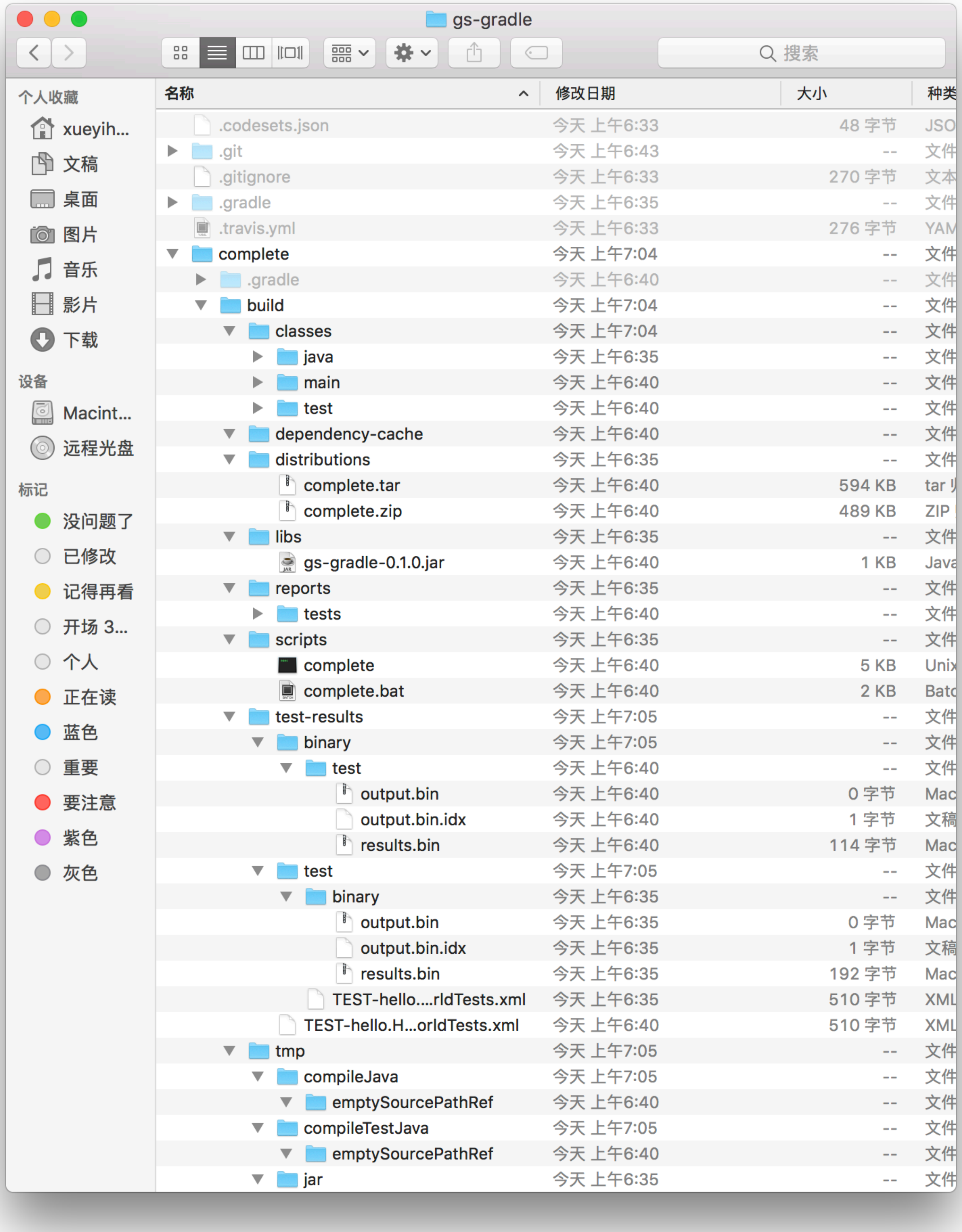
7. `./gradlew_run`



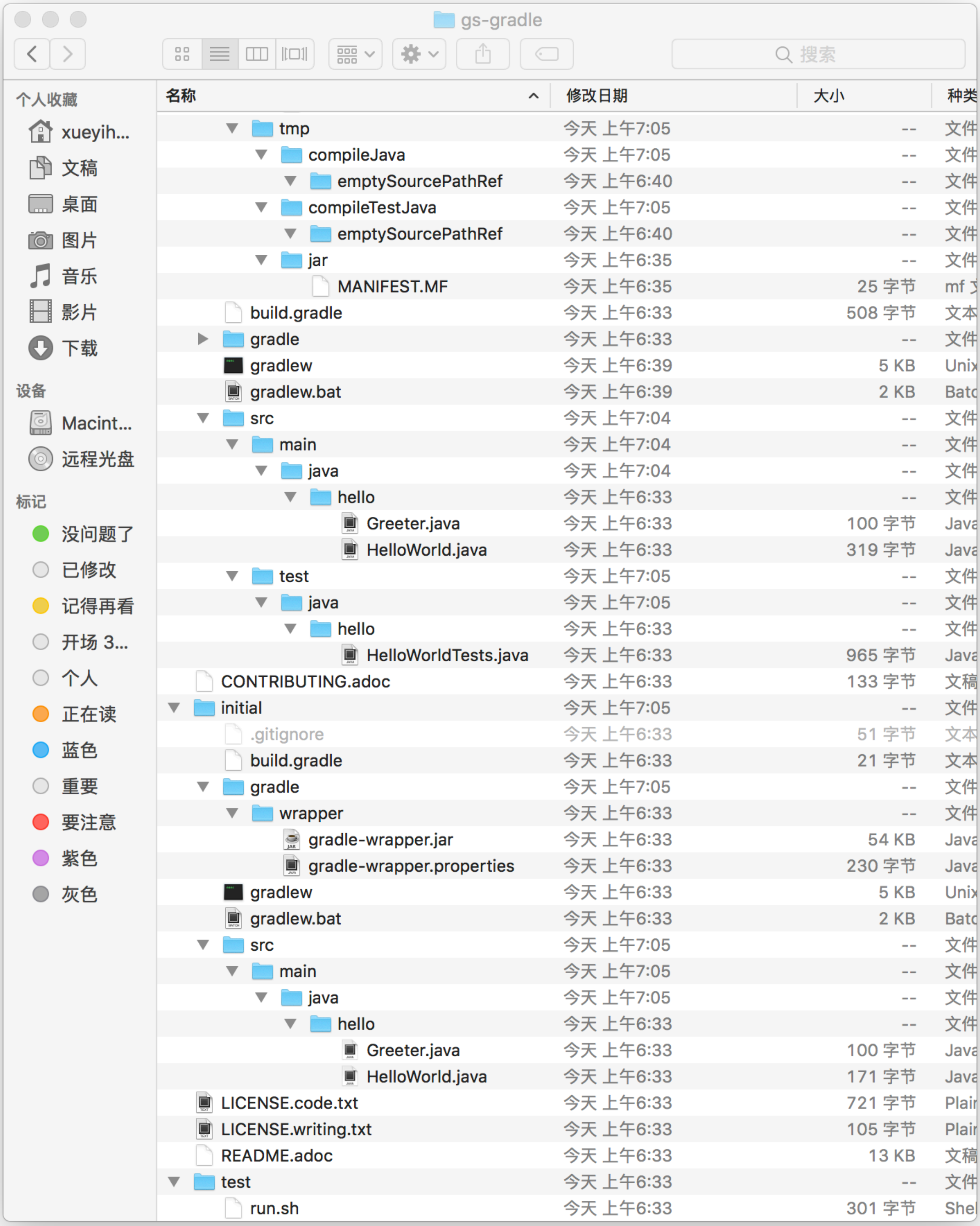
complete — xueyiheng@xueyihengdeMacBook-Pro — ..adle/complete — -zsh — 80×15

```
➜ complete git:(master) ✗ ./gradlew run
:compileJava UP-TO-DATE
:processResources UP-TO-DATE
:classes UP-TO-DATE
:run
The current local time is: 06:42:55.087
Hello world!

BUILD SUCCESSFUL

Total time: 3.873 secs

This build could be faster, please consider using the Gradle Daemon: https://doc
s.gradle.org/2.13/userguide/gradle_daemon.html
➜ complete git:(master) ✗
```

8. 文件夹内容



名称	修改日期	大小	种类
.codesets.json	今天 上午6:33	48 字节	JSO
▶ .git	今天 上午6:43	--	文件
.gitignore	今天 上午6:33	270 字节	文本
▶ .gradle	今天 上午6:35	--	文件
.travis.yml	今天 上午6:33	276 字节	YAM
▼ complete	今天 上午7:04	--	文件
▶ .gradle	今天 上午6:40	--	文件
▼ build	今天 上午7:04	--	文件
▼ classes	今天 上午7:04	--	文件
▶ java	今天 上午6:35	--	文件
▶ main	今天 上午6:40	--	文件
▶ test	今天 上午6:40	--	文件
▼ dependency-cache	今天 上午6:40	--	文件
▼ distributions	今天 上午6:35	--	文件
complete.tar	今天 上午6:40	594 KB	tar
complete.zip	今天 上午6:40	489 KB	ZIP
▼ libs	今天 上午6:35	--	文件
gs-gradle-0.1.0.jar	今天 上午6:40	1 KB	Java
▼ reports	今天 上午6:35	--	文件
▶ tests	今天 上午6:40	--	文件
▼ scripts	今天 上午6:35	--	文件
complete	今天 上午6:40	5 KB	Unix
complete.bat	今天 上午6:40	2 KB	Batc
▼ test-results	今天 上午7:05	--	文件
▼ binary	今天 上午7:05	--	文件
▼ test	今天 上午6:40	--	文件
output.bin	今天 上午6:40	0 字节	Mac
output.bin.idx	今天 上午6:40	1 字节	文稿
results.bin	今天 上午6:40	114 字节	Mac
▼ test	今天 上午7:05	--	文件
▼ binary	今天 上午6:35	--	文件
output.bin	今天 上午6:35	0 字节	Mac
output.bin.idx	今天 上午6:35	1 字节	文稿
results.bin	今天 上午6:35	192 字节	Mac
TEST-hello....rldTests.xml	今天 上午6:35	510 字节	XML
TEST-hello.H...orldTests.xml	今天 上午6:40	510 字节	XML
▼ tmp	今天 上午7:05	--	文件
▼ compileJava	今天 上午7:05	--	文件
▼ emptySourcePathRef	今天 上午6:40	--	文件
▼ compileTestJava	今天 上午7:05	--	文件
▼ emptySourcePathRef	今天 上午6:40	--	文件
▼ jar	今天 上午6:35	--	文件



名称	修改日期	大小	种类
▼ tmp	今天 上午7:05	--	文件
▼ compileJava	今天 上午7:05	--	文件
▼ emptySourcePathRef	今天 上午6:40	--	文件
▼ compileTestJava	今天 上午7:05	--	文件
▼ emptySourcePathRef	今天 上午6:40	--	文件
▼ jar	今天 上午6:35	--	文件
MANIFEST.MF	今天 上午6:35	25 字节	mf 文
build.gradle	今天 上午6:33	508 字节	文本
▶ gradle	今天 上午6:33	--	文件
gradlew	今天 上午6:39	5 KB	Unix
gradlew.bat	今天 上午6:39	2 KB	Batc
▼ src	今天 上午7:04	--	文件
▼ main	今天 上午7:04	--	文件
▼ java	今天 上午7:04	--	文件
▼ hello	今天 上午6:33	--	文件
Greeter.java	今天 上午6:33	100 字节	Java
HelloWorld.java	今天 上午6:33	319 字节	Java
▼ test	今天 上午7:05	--	文件
▼ java	今天 上午7:05	--	文件
▼ hello	今天 上午6:33	--	文件
HelloWorldTests.java	今天 上午6:33	965 字节	Java
CONTRIBUTING.adoc	今天 上午6:33	133 字节	文稿
▼ initial	今天 上午7:05	--	文件
.gitignore	今天 上午6:33	51 字节	文本
build.gradle	今天 上午6:33	21 字节	文本
▼ gradle	今天 上午7:05	--	文件
▼ wrapper	今天 上午6:33	--	文件
gradle-wrapper.jar	今天 上午6:33	54 KB	Java
gradle-wrapper.properties	今天 上午6:33	230 字节	Java
gradlew	今天 上午6:33	5 KB	Unix
gradlew.bat	今天 上午6:33	2 KB	Batc
▼ src	今天 上午7:05	--	文件
▼ main	今天 上午7:05	--	文件
▼ java	今天 上午7:05	--	文件
▼ hello	今天 上午6:33	--	文件
Greeter.java	今天 上午6:33	100 字节	Java
HelloWorld.java	今天 上午6:33	171 字节	Java
LICENSE.code.txt	今天 上午6:33	721 字节	Plair
LICENSE.writing.txt	今天 上午6:33	105 字节	Plair
README.adoc	今天 上午6:33	13 KB	文稿
▼ test	今天 上午6:33	--	文件
run.sh	今天 上午6:33	301 字节	Shel

Assignment 3

- Convert the Ant build.xml of the tool used in Lab 6 to Gradle, and try to make it works for the same usability.
- You **cannot** use any existed automatic tools to convert.

Markup

```
<project name="tarantula" default="compile" basedir=".">

<!-- patternset to decrease redundancy in the buildfile-->
   <patternset id="jars" >
      <include name="**/*.jar"/>
   </patternset>

<!-- Path attribute using the patternset-->
   <path id="jar.files">
      <fileset dir="lib">
         <patternset refid="jars"/>
      </fileset>
   </path>

<!-- Target to clean up -->
    <target name="clean" description="Clean">
        <delete dir="./bin"/>
    </target>

<!-- Target to initialize build -->
    <target name="init">
        <mkdir dir="./bin"/>
    </target>

<!-- Target to compile the project -->
    <target name="compile" depends="init" description="Compile">
        <javac includeantruntime="true" srcdir="src" destdir="bin" debug="yes" fork="no">
            <classpath refid="jar.files"/>
        </javac>
    </target>

<!-- Target to run the main-->
    <target name="run.tarantula" depends="compile">
        <java classname="tarantula.Main">
            <classpath>
               <path refid="jar.files"/>
               <pathelement location="bin"/>
            </classpath>
            <arg value="${arg0}"/>
            <arg value="${arg1}"/>
        </java>
    </target>
</project>
```

对**Ant**的 `Build.xml` 进行分析得到如下信息

- Project
    - 工程名称： `tarantula`
    - 基准目录： `.`
    - 默认运行的target： `compile` （当ant命令没有制定target时，会运行这个target）

- Patternset
    - id唯一标识： `jars` （其他元素通过refid指向该模式）
    - 内嵌要包含的模式： `**/*.jar`

- Path
    - id： `jar.files`
        - fileset
            - 文件集合的根目录： `lib`
            - 模式集合的refid： `jars`

- Target
    - clean
        - 要删除的目录： `./bin/`

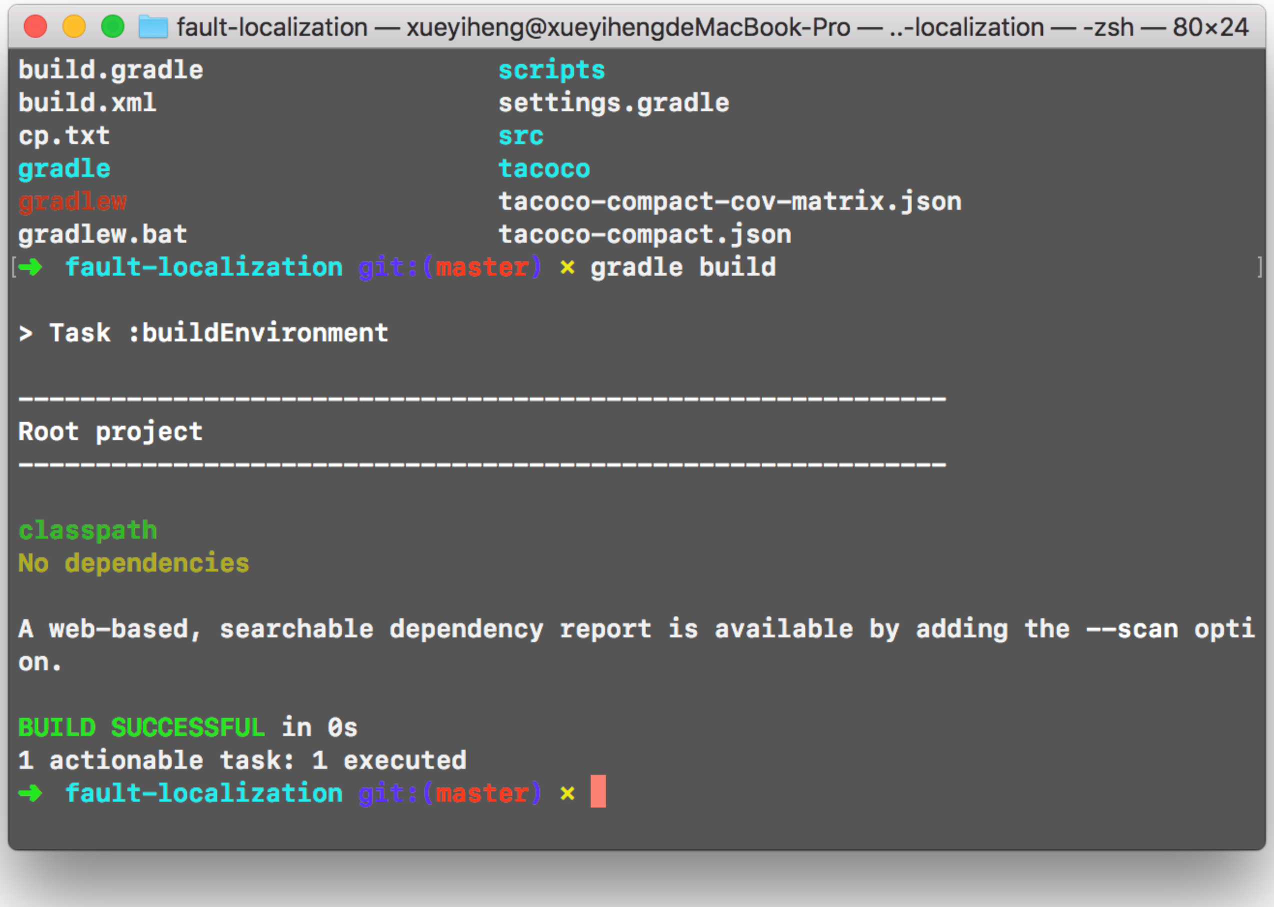
    - init

        - 要创建的目录： `./bin/`

    - compile
        - 依赖的target： `init`
        - 是否包含 `ant` 库路径： `True`
        - 源文件路径： `src`
        - 存放变异后class文件路径： `bin`
        - 编译时是否显示调试信息： `yes`
        - `javac` 的依赖库路径： `jar.files`

    - run.tarantula
        - 依赖的target： `compile`
        - java类名： `tarantula.Main`
        - classpath中类路径指向改id对应的元素： `jar.files`
        - 单个文件或目录： `bin`
        - `arg value` ： **NOTSURE**

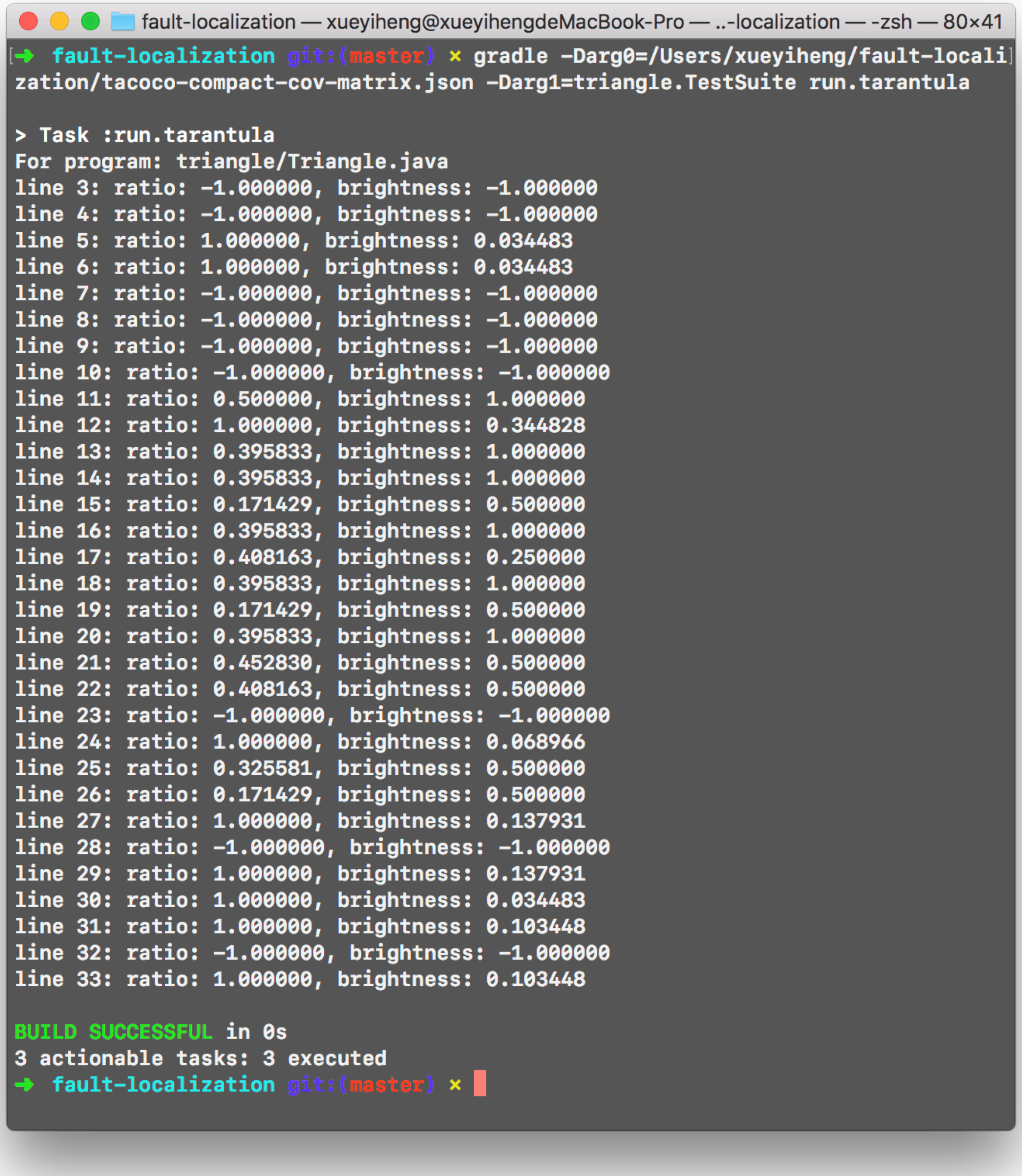
如下图所示通过 `gradle build`



fault-localization — xueyiheng@xueyihengdeMacBook-Pro — ..-localization — -zsh — 80×24

```
build.gradle                    scripts
build.xml                       settings.gradle
cp.txt                          src
gradle                          tacoco
gradlew                         tacoco-compact-cov-matrix.json
gradlew.bat                     tacoco-compact.json
➜  fault-localization git:(master) ✗ gradle build

> Task :buildEnvironment

------------------------------------------------------------
Root project
------------------------------------------------------------

classpath
No dependencies

A web-based, searchable dependency report is available by adding the --scan opti
on.

BUILD SUCCESSFUL in 0s
1 actionable task: 1 executed
➜  fault-localization git:(master) ✗ 
```

`gradle -Darg0=/Users/xueyiheng/fault-localization/tacoco-compact-cov-matrix.json -Darg1=triangle.TestSuite run.tarantula` 运行的出和 `ant` 的 `build.xml` 相同的结果，如下图所示。


fault-localization — xueyiheng@xueyihengdeMacBook-Pro — ..-localization — -zsh — 80×41

```
➜  fault-localization git:(master) ✗ gradle -Darg0=/Users/xueyiheng/fault-locali
zation/tacoco-compact-cov-matrix.json -Darg1=triangle.TestSuite run.tarantula

> Task :run.tarantula
For program: triangle/Triangle.java
line 3: ratio: -1.000000, brightness: -1.000000
line 4: ratio: -1.000000, brightness: -1.000000
line 5: ratio: 1.000000, brightness: 0.034483
line 6: ratio: 1.000000, brightness: 0.034483
line 7: ratio: -1.000000, brightness: -1.000000
line 8: ratio: -1.000000, brightness: -1.000000
line 9: ratio: -1.000000, brightness: -1.000000
line 10: ratio: -1.000000, brightness: -1.000000
line 11: ratio: 0.500000, brightness: 1.000000
line 12: ratio: 1.000000, brightness: 0.344828
line 13: ratio: 0.395833, brightness: 1.000000
line 14: ratio: 0.395833, brightness: 1.000000
line 15: ratio: 0.171429, brightness: 0.500000
line 16: ratio: 0.395833, brightness: 1.000000
line 17: ratio: 0.408163, brightness: 0.250000
line 18: ratio: 0.395833, brightness: 1.000000
line 19: ratio: 0.171429, brightness: 0.500000
line 20: ratio: 0.395833, brightness: 1.000000
line 21: ratio: 0.452830, brightness: 0.500000
line 22: ratio: 0.408163, brightness: 0.500000
line 23: ratio: -1.000000, brightness: -1.000000
line 24: ratio: 1.000000, brightness: 0.068966
line 25: ratio: 0.325581, brightness: 0.500000
line 26: ratio: 0.171429, brightness: 0.500000
line 27: ratio: 1.000000, brightness: 0.137931
line 28: ratio: -1.000000, brightness: -1.000000
line 29: ratio: 1.000000, brightness: 0.137931
line 30: ratio: 1.000000, brightness: 0.034483
line 31: ratio: 1.000000, brightness: 0.103448
line 32: ratio: -1.000000, brightness: -1.000000
line 33: ratio: 1.000000, brightness: 0.103448

BUILD SUCCESSFUL in 0s
3 actionable tasks: 3 executed
➜  fault-localization git:(master) ✗ 
```

Assignment 4

- Deploy your project in Assignment 1 on Travis-Ci
- 未完成