

Homework 4:

Mutation Based Testing

Software Testing 2021

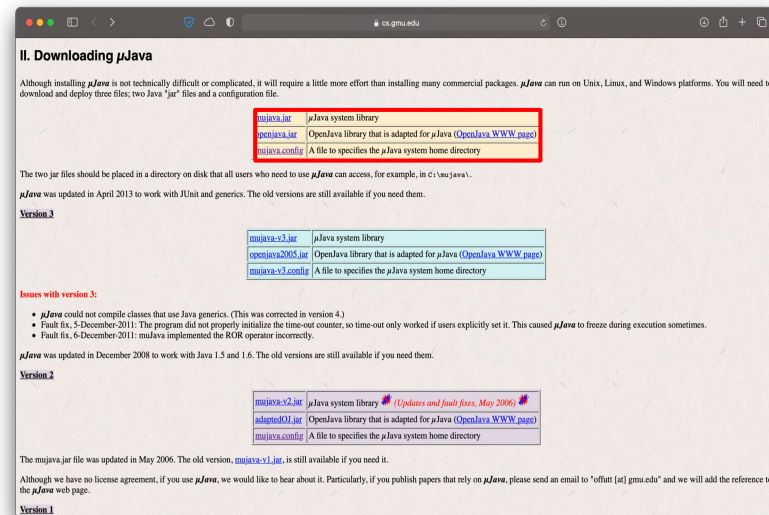
2021/05/27

muJava

muJava

→ Download **mujava.jar**, **openjava.jar** and **mujava.config**

◆ From: <https://cs.gmu.edu/~offutt/mujava/>



The screenshot shows a web browser window displaying the "II. Downloading μ Java" page. The page contains instructions for downloading and installing μ Java. A red box highlights the download links for Version 3: [mujava.jar](#) (Java system library), [openjava.jar](#) (OpenJava library adapted for μ Java), and [mujava.config](#) (file specifying the μ Java system home directory). Below this, the text states that the two jar files should be placed in a directory accessible to all users. Further down, another table lists download links for Version 2: [mujava-v2.jar](#), [openjava2005.jar](#), and [mujava-v2.config](#). The page also includes a section for "Issues with version 3:" listing several bugs and their fixes. At the bottom, there is a section for "Version 1" which is mostly obscured.

II. Downloading μ Java

Although installing μ Java is not technically difficult or complicated, it will require a little more effort than installing many commercial packages. μ Java can run on Unix, Linux, and Windows platforms. You will need to download and deploy three files; two Java "jar" files and a configuration file.

mujava.jar	Java system library
openjava.jar	OpenJava library that is adapted for μ Java (OpenJava WWW page)
mujava.config	A file that specifies the μ Java system home directory

The two jar files should be placed in a directory on disk that all users who need to use μ Java can access, for example, in `C:\muJava\`.

μ Java was updated in April 2013 to work with JUnit and generics. The old versions are still available if you need them.

Version 3

mujava-v3.jar	Java system library
openjava2005.jar	OpenJava library that is adapted for μ Java (OpenJava WWW page)
mujava-v3.config	A file that specifies the μ Java system home directory

Issues with version 3:

- μ Java could not compile classes that use Java generics. (This was corrected in version 4.)
- Fault fix, 5-December-2011: The program did not properly initialize the time-out counter, so time-out only worked if users explicitly set it. This caused μ Java to freeze during execution sometimes.
- Fault fix, 6-December-2011: μ Java implemented the ROR operator incorrectly.

μ Java was updated in December 2008 to work with Java 1.5 and 1.6. The old versions are still available if you need them.

Version 2

mujava-v2.jar	Java system library (Updates and fault fixes, May 2006)
adaptedOJ.jar	OpenJava library that is adapted for μ Java (OpenJava WWW page)
mujava.config	A file that specifies the μ Java system home directory

The mujava.jar file was updated in May 2006. The old version, [mujava-v1.jar](#), is still available if you need it.

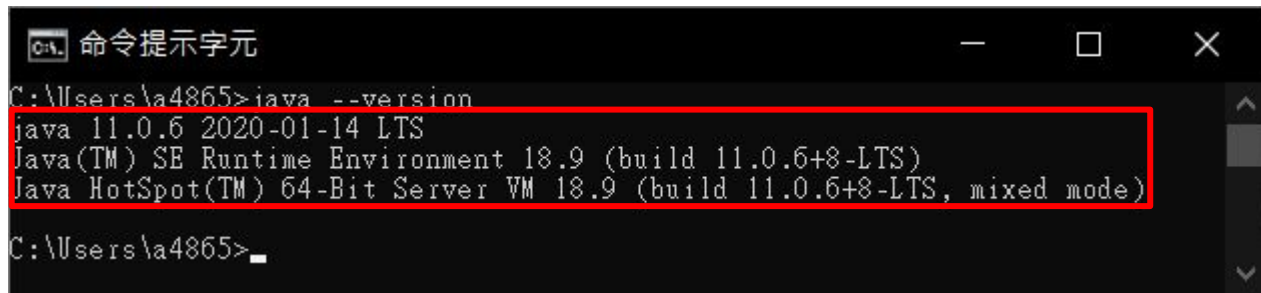
Although we have no license agreement, if you use μ Java, we would like to hear about it. Particularly, if you publish papers that rely on μ Java, please send an email to "offutt[at]gmu.edu" and we will add the reference to the μ Java web page.

Version 1

Installation - Windows 10

→ Step 1 :

- ◆ check jdk version (at least 1.8)
 - **\$ java --version**

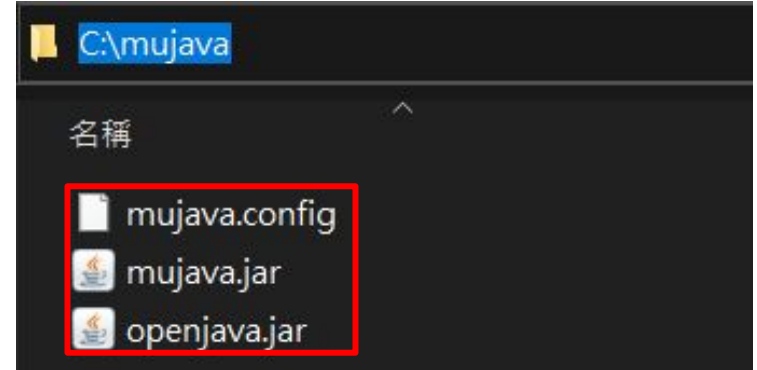


```
命令提示字元
C:\Users\A4865>java --version
java 11.0.6 2020-01-14 LTS
Java(TM) SE Runtime Environment 18.9 (build 11.0.6+8-LTS)
Java HotSpot(TM) 64-Bit Server VM 18.9 (build 11.0.6+8-LTS, mixed mode)
C:\Users\A4865>
```

Installation - Windows 10

→ Step 2 :

- ◆ Put the file in here :
 - C:\mujava

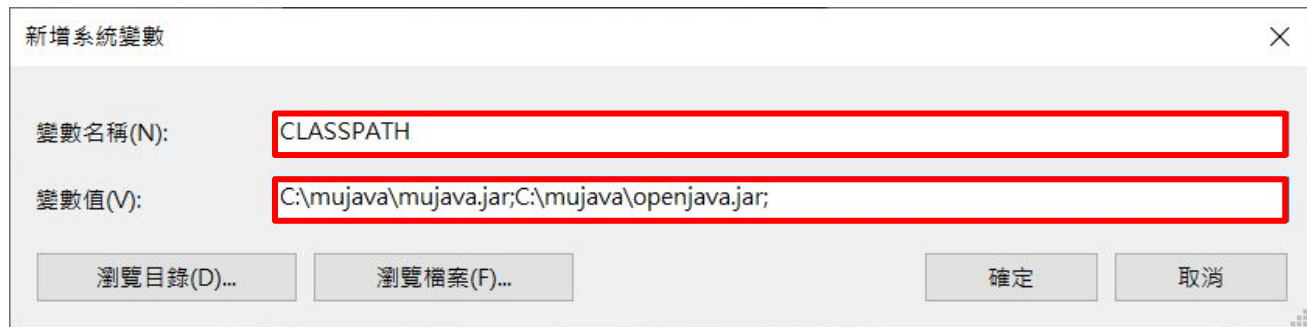


Installation - Windows 10

→ Step 3 :

◆ Set CLASSPATH

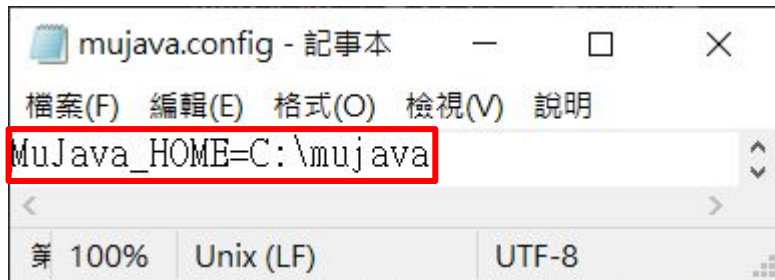
- 本機 → 內容 → 進階系統設定 → 進階 → 環境變數 → 系統變數 → 新增
 - 變數名稱: CLASSPATH
 - 變數值: C:\mujava\mujava.jar;C:\mujava\openjava.jar;



Installation - Windows 10

→ Step 4 :

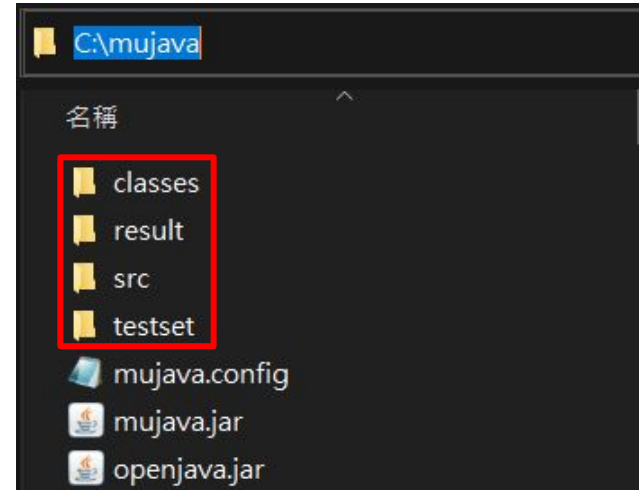
- ◆ Modify the content of ***mujava.config***
 - MuJava_HOME=C:\mujava



Installation - Windows 10

→ Step 5 :

- ◆ Create these four folders

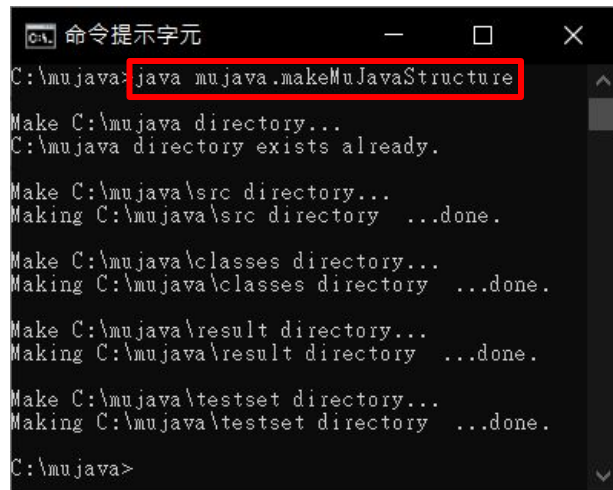


Installation - Windows 10

→ Step 5 :

◆ Or, can use cmd create

- **\$ java mujava.makeMuJavaStructure**



```
命令提示字元
C:\mujava> java mujava.makeMuJavaStructure
Make C:\mujava directory...
C:\mujava directory exists already.
Make C:\mujava\src directory...
Making C:\mujava\src directory ...done.
Make C:\mujava\classes directory...
Making C:\mujava\classes directory ...done.
Make C:\mujava\result directory...
Making C:\mujava\result directory ...done.
Make C:\mujava\testset directory...
Making C:\mujava\testset directory ...done.
C:\mujava>
```

Installation - Ubuntu 20.04

→ Steps

- ◆ install openjdk-11
 - `sudo apt-get install openjdk-11-jdk`
- ◆ export CLASSPATH
 - `export CLASSPATH=/DIR/mujava.jar:/DIR/openjava.jar`
- ◆ set mujava.config
 - `MuJava_HOME=/DIR/`

```
tl455047@tl455047-X556UR ▣ ~/st_hw4 ▣ java --version
openjdk 11.0.11 2021-04-20
OpenJDK Runtime Environment (build 11.0.11+9-Ubuntu-0ubuntu2.20.04)
OpenJDK 64-Bit Server VM (build 11.0.11+9-Ubuntu-0ubuntu2.20.04, mixed mode, sharing)
tl455047@tl455047-X556UR ▣ ~/st_hw4 ▣ echo $CLASSPATH
/home/tl455047/st_hw4/mujava.jar:/home/tl455047/st_hw4/openjava.jar
tl455047@tl455047-X556UR ▣ ~/st_hw4 ▣ cat mujava.config
MuJava_HOME=/home/tl455047/st_hw4
```

Installation - macOS 11.4

→ Steps

- ◆ `$ export CLASSPATH=$CLASSPATH:/Users/chtsai/muJava/mujava.jar:/Users/chtsai/muJava/openjava.jar`
 - e.g. `/Users/chtsai/muJava`
- ◆ `$ java --version`
 - `openjdk 11.0.9 2020-10-20`
 - `OpenJDK Runtime Environment (build 11.0.9+11)`
 - `OpenJDK 64-Bit Server VM (build 11.0.9+11, mixed mode)`



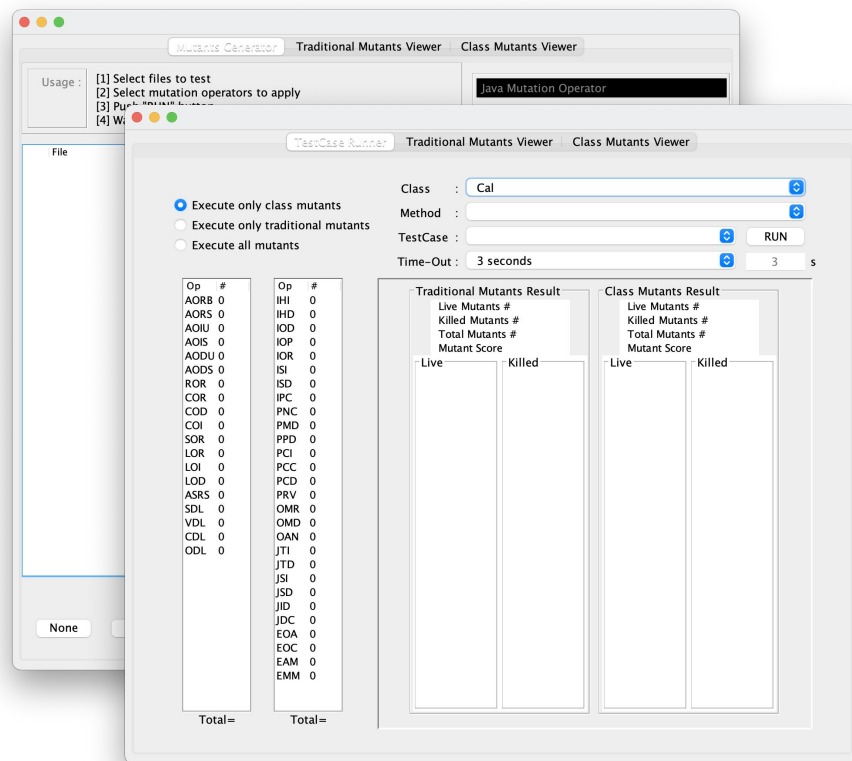
Homework

→ Installation TL;DR

- Get Java version
- Export CLASSPATH
 - jar files
- Set muJava configuration
 - specify muJava home directory

→ Run muJava

- `$ java mujava.gui.GenMutantsMain`
- `$ java mujava.gui.RunTestMain`



Hint

1. Junit 4 only

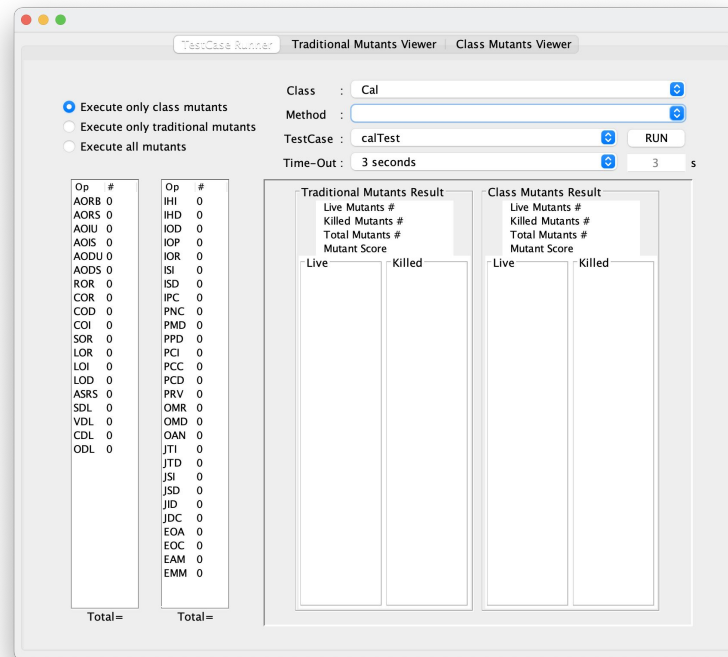
- a. export CLASSPATH
 - i. [junit-4.12.jar](#)
 - ii. [hamcrest-core-1.3.jar](#)

2. Method is blank

- a. choose **“class”** list first

3. TestCase is blank

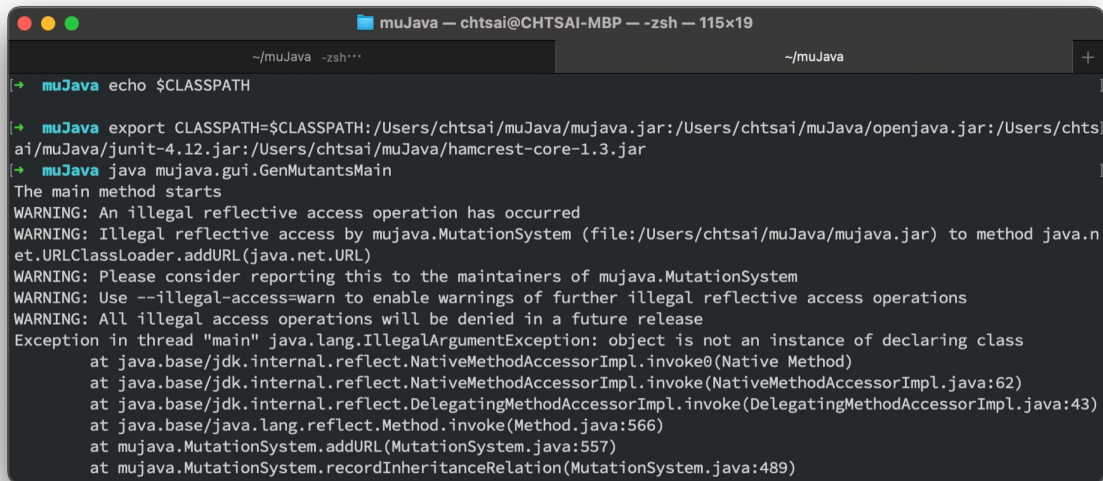
- a. write test cases (CalTest.java) and compile
 - i. `javac CalTest.java ../src/Cal.java`
- b. location
 - i. `*/muJava/session1/testset/CalTest.class`



Hint

4. After GenMutantsMain

- Avoid score is always 100%
- Prevent the warning
- `cp */muJava/session1/src/Cal.class */muJava/session1/classes`**



```
muJava — chtsai@CHTSAI-MBP — -zsh — 115x19
~/muJava -zsh***
[+] muJava echo $CLASSPATH
[+] muJava export CLASSPATH=$CLASSPATH:/Users/chtsai/muJava/mujava.jar:/Users/chtsai/muJava/openjava.jar:/Users/chtsai/muJava/junit-4.12.jar:/Users/chtsai/muJava/hamcrest-core-1.3.jar
[+] muJava java mujava.gui.GenMutantsMain
The main method starts
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by mujava.MutationSystem (file:/Users/chtsai/muJava/mujava.jar) to method java.net.URLClassLoader.addURL(java.net.URL)
WARNING: Please consider reporting this to the maintainers of mujava.MutationSystem
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
Exception in thread "main" java.lang.IllegalArgumentException: object is not an instance of declaring class
    at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
    at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.base/java.lang.reflect.Method.invoke(Method.java:566)
    at mujava.MutationSystem.addURL(MutationSystem.java:557)
    at mujava.MutationSystem.recordInheritanceRelation(MutationSystem.java:489)
```