

CS 351
Design of Large Programs
Submitting your Work

Brooke Chenoweth

University of New Mexico

Spring 2024

JDK and IntelliJ on CS machines

- JDK 21 with JavaFX located at `/usr/lib/jvm/zulu-fx-21-amd64/bin`
It is also configured to be the default java linked from `/usr/bin`, so both `java` and `javac` should already be in your path. (Unless you previously messed around with your environment variables.)
- IntelliJ is installed at `/opt/idea-IC-231.8770.65/bin`
The startup script is linked from `/usr/local/bin/idea.sh`, so that also should be in your path by default.

Submitting In Canvas

- Each programming assignment must be submitted in Canvas as a link to a Lobogit repository.
- For group work, the group will submit a link to a single shared repository.
- Suggestion: once you have set up the repository, go ahead and submit the link to Canvas immediately.
- If you have submitted your repository link before the deadline, I'm willing to let you take a day or two extra to polish your work. (Fix typos, proofread your readme, small bug fixes)

Lobogit GitLab Server

- UNM has a GitLab server at `https://lobogit.unm.edu`
- You log on with your UNM credentials.

GitLab repository

- Repository must be *private*
- Add instructors as collaborators so we can see your work.
- Brooke's username: bchenoweth
- Nicholas' username: nlivingsto

Project Structure

Inside your Project Repository

- README.md
- .gitignore
- src/
- doc/
- resources/ – If using any resource files
- descrip_name.jar
 - descrip is meaningful descriptive string (version number, etc.)
 - name is your username, or for group projects, each member's username, separated with underscores
 - There will be a jar for each incrementally demonstrated version.

.gitignore file

- In general, auto-generated files should not be included in your repository and so should be ignored.
 - Exception: the required jar file(s) submitted for the assignment
- In particular, make sure out (or bin) directory with the .class files is ignored.
- Example .gitignore contents:

```
out  
.idea  
*.iml
```

Jar file

- Must be executable JAR with all resources needed to run.
- Must include the source used to build the program (This is not the default when building a JAR, so you need to explicitly configure to include the java files)
- Jar files must be built with JDK 21.

README.md

Each assignment includes a README file in Markdown format that explains:

- How to use the program (which class is entry point, command line arguments, etc...)
- Any testing and/or debugging tools/options in the project and how to use them.
- For group projects: Who did which parts of the program
- Any known issues (bugs, unfinished features).

doc directory

- Includes all documentation other than the README and JavaDoc (we don't need generated Javadoc pages, just what is in the source code)
- Must include a design document in PDF format.
 - First page/slide is object diagram.
 - Description of objects follows diagram on next page/slide.