

Assignment 2 requires you to write a program in Java. To complete the assignment you will need to (1) install a Java compiler, (2) install a Java integrated development environment (IDE), (3) complete the assignment by writing the Java code, and (4) turn in the assignment electronically. In addition to these steps, this document provides (5) a troubleshooting guide for IntelliJ projects for this course.

1. Install a Java compiler.

We will be using Java 11 from Adoptium, which supplies a free Java Development Kit (JDK) with an open-source license. Click on the following link.

<https://adoptium.net/temurin/releases/>

Select your operating system, Package Type JDK, and Version 11. Follow the usual procedure for your platform – Mac, Windows, or Linux – to install the JDK.

2. Install a Java Integrated Development Environment (IDE).

We will use an IDE from JetBrains called IntelliJ IDEA. Here is a link to the JetBrains IntelliJ IDEA page.

<https://www.jetbrains.com/idea/>

Click the Downloads button to get to the downloads page for your platform. The page shows two versions of IntelliJ IDEA – Ultimate for enterprise development and Community for free open-source. Select the Community version, download it, and install it on your machine. The app will be named "IntelliJ IDEA CE".

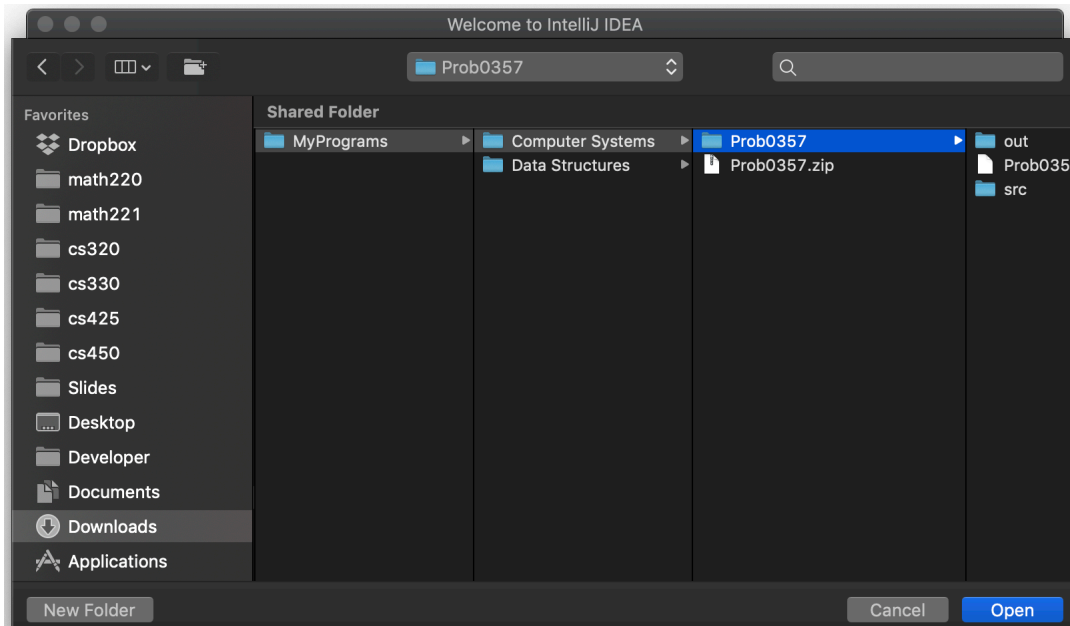
NOTE: There is a third version of IntelliJ IDEA for education that you might encounter on the JetBrains site. Do *not* install the edu version.

3. Complete Assignment 2.

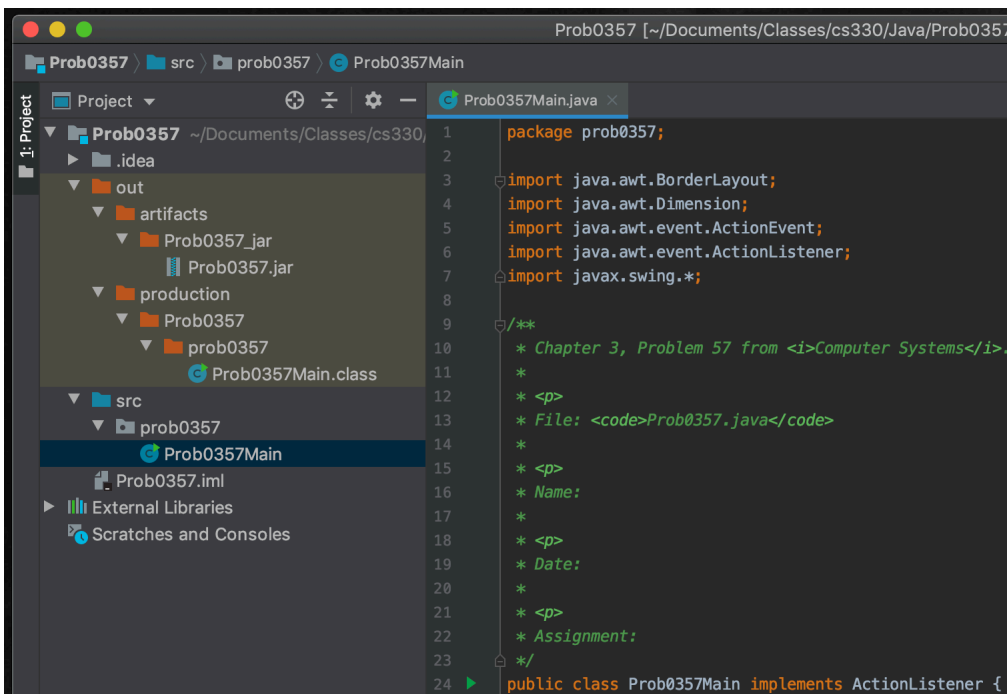
To write the Java program with IntelliJ:

- a. Download the file Prob0357.zip from the course web site.
Move the .zip file to the directory where you will keep your Java programs for this course.
Uncompress the .zip file.
- b. Open IntelliJ. From the splash screen or from the File menu select Open Project.
Navigate to the uncompressed Prob0357 directory and click Open.

(continued on next page)



- c. In the project pane on the left side, double click Prob0357 → src → Prob0357Main to open the source file to edit. In the documentation section at the top, fill in your name, date, and the assignment, which is required on all homework submissions.

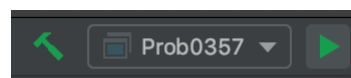


- d. Scroll to the bottom, and replace the code

```
// Problem for the student
System.out.printf("Print the next ten binary numbers here.");
```

with the code to complete the assignment.

- e. To run the program click the green triangle run icon.



4. Turn in Assignment 2 electronically.
- a. The standard distribution format for Java programs is the Java Archive (JAR) file. The IntelliJ projects you download from the course web site are configured to automatically create a `.jar` file when you run the program. For Assignment 2, you must navigate in the Finder (Mac) or File Explorer (Windows) to

Prob0357 → out → artifacts → Prob0357_jar → Prob0357.jar

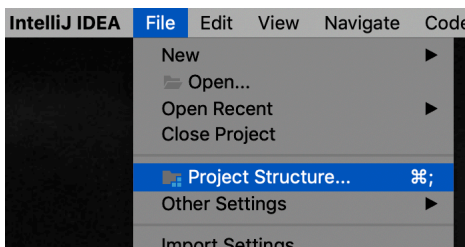


Duplicate the file.

- b. Rename the duplicated file according to the specifications from your instructor.
- For Pepperdine students, rename it to `xxProb0357.jar` where `xx` is your assigned two-digit number. For example, if your two-digit number is 99 you would rename it `99Prob0357.jar`. Hand in this file as an attachment in Sakai Courses.
5. Troubleshoot IntelliJ projects for this course.

Occasionally when you download and unzip an IntelliJ project for this course the project settings are not preserved. When that happens you must manually set them. This section lists the project settings for Prob0357 of Assignment 2. All the Java projects for this course have the same settings except for the specific name Prob0357.

- a. From the File menu select File → Project Structure...



In the Project category, there are four fields.

The Project name should be Prob0357.

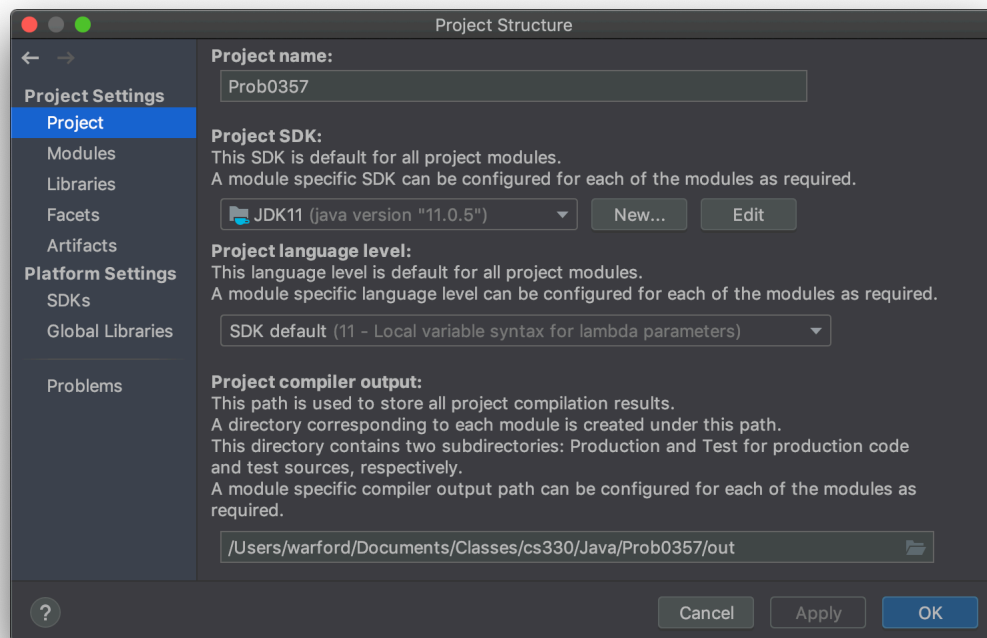
The Project SDK should be JDK11. If you cannot set it to JDK11 you may need to first set the name in part d. below.

The Project language level should be the default and contain (11 - Local variable syntax for lambda parameters).

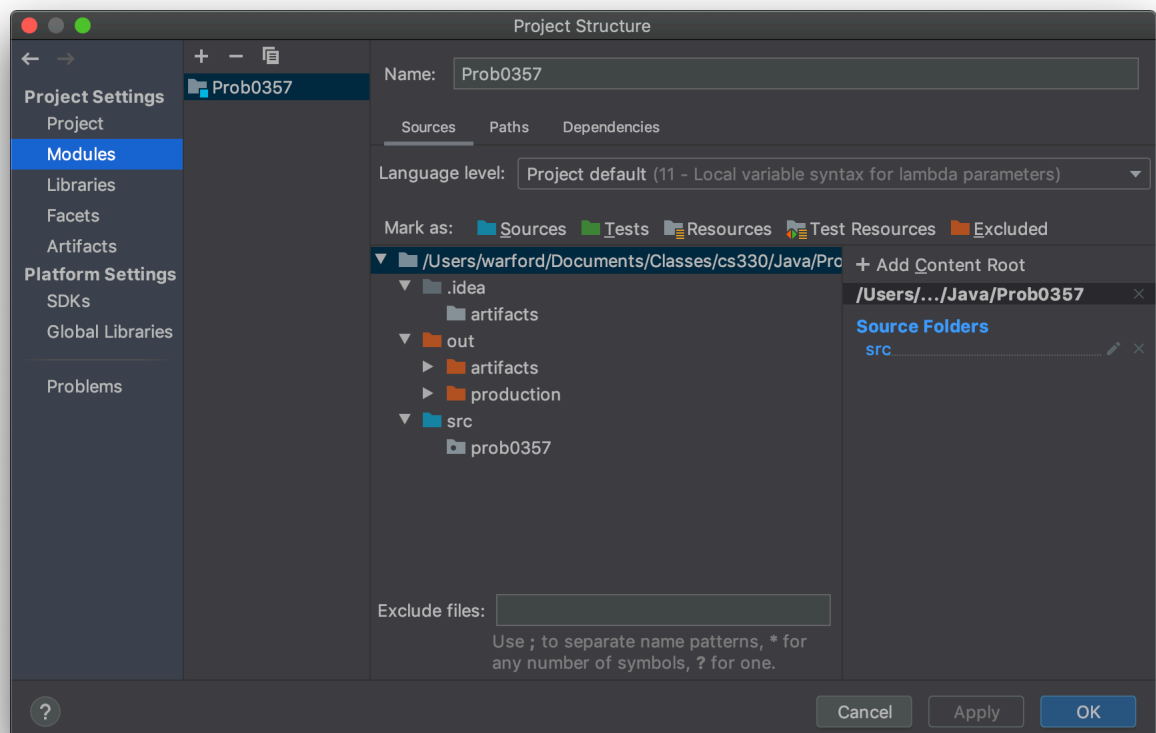
The Project compiler output should be the path to the out folder in your project. That is, it should end with `/Prob0357/out` and begin with the path to the folder where you put the project.

If any of these settings are not correct change them and click the Apply button.

(continued on next page)



- b. In the Modules category, there should be one module named Prob0357.



(continued on next page)

Under the Sources tab, the Language level should be the default and contain (11 - Local variable syntax for lambda parameters).

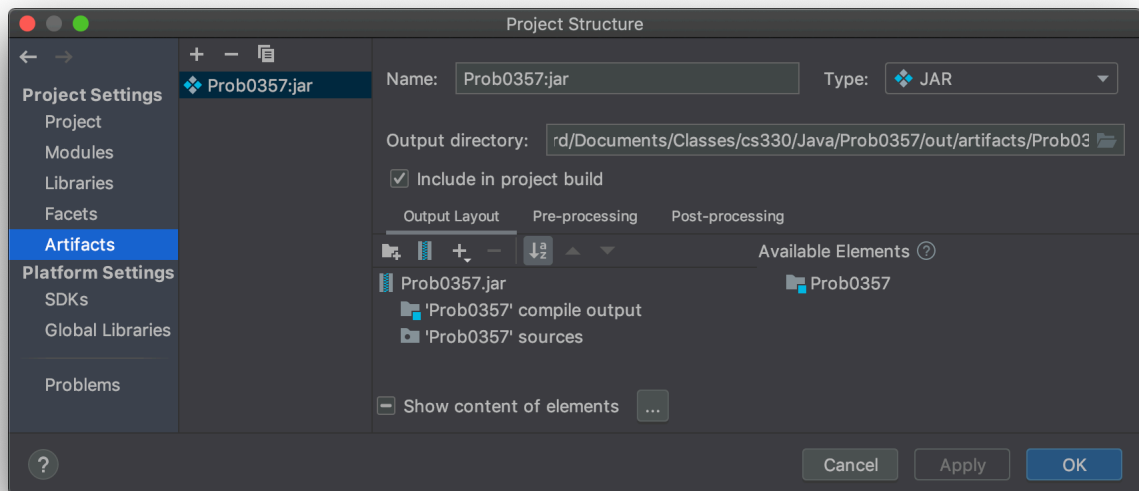
The `src` folder must be color-coded to be marked as Sources. If it is not color-coded as a source, click on + Add Content Root to add it as a Source Folder.

- c. In the Artifacts category, there should be one artifact named `Prob0357.jar` of type JAR.

The output directory should begin with the path to your project folder and end with `Prob0357/out/artifacts/Prob0357.jar`.

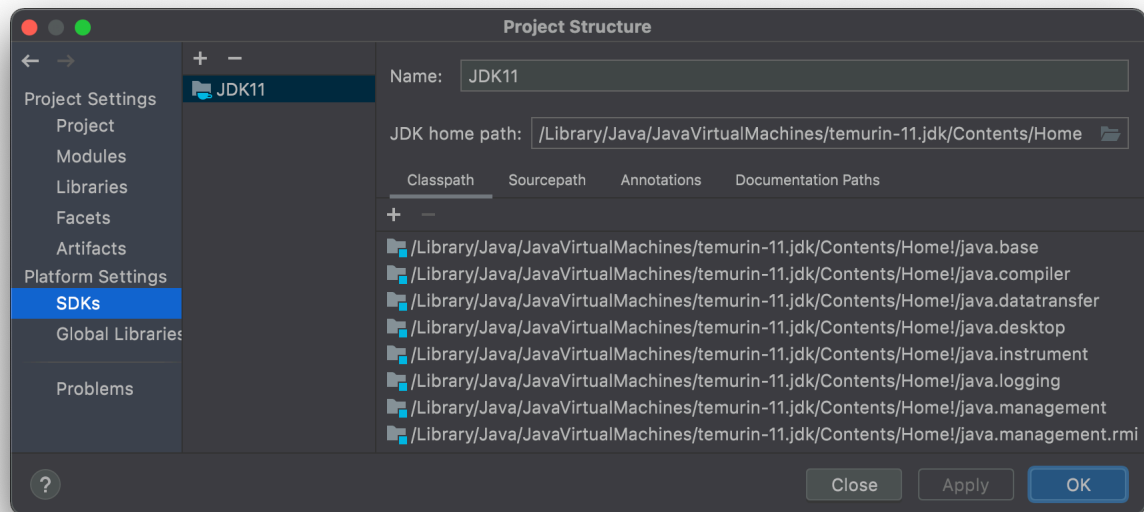
Be sure that the item Include in project build is checked so that the JAR file will be created automatically each time you build your project.

Under the Output Layout tab there must be an item named 'Prob0357' sources. If not, click the + button to add the directory contents of the `src` folder. Otherwise, your source code will not be included in the JAR file and your homework cannot be graded.



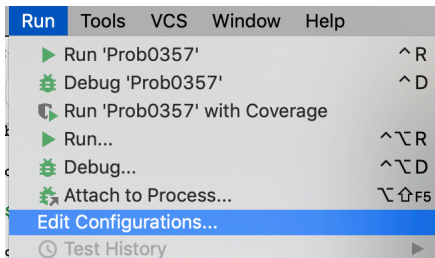
(continued on next page)

- d. In the SDKs (software development kit) category, there should be one SDK named JDK11. The JDK home path should be the path to where the Adoptium temurin release was installed.



(continued on next page)

- e. From the Run menu select Run → Edit Configurations... .



In the left pane of the Run/Debug Configurations window there should be an Application named Prob0357. If not, click the + button and select Application to add a new configuration.

The name of the configuration should be Prob0357.

Under Build and run, the tool kit should be java 11 JDK11.

The Main class should be `prob0357.Prob0357Main`. You will be given a warning if you have not designated `src` as a source folder in the Modules category as described in part b. above.

