


## 1 Introduction

The programming environment for the course can be setup using the given pre-configured virtual machine image (see Section 2) or natively (see Section 3).

## 2 Virtual Machine Setup

For the virtual machine (VM) to work properly, you will need a 64-bit, virtualization-enabled machine with at least 8GB of RAM and 30GB of free disk space.

### 2.1 Steps

1. Download Virtual Box for your operating system:
  - Ubuntu Linux (Jammy) [↗](#)
  - Mac OS X (Intel) [↗](#)
  - Mac OS X (ARM) [↗](#)
  - Windows [↗](#)
2. Download Virtual Box Extension Pack [↗](#).
3. Install VirtualBox you downloaded in step 1.
4. Launch VirtualBox, go to *File* → *Preferences* → *Extensions* and add the Virtual Box Extension Pack you downloaded in step 2.
5. Download the pre-configured VM [↗](#).
6. Go to *File* → *Import Appliance*. Choose the file (`ElementaryOS.ova`) you downloaded in step 5.
7. Select *ElementaryOS*, go to *Settings* → *USB*, and select *Enable USB Controller* → *USB 3.0 (xHCI) Controller*.
8. To start the VM, select *ElementaryOS* and click *Start*. Log in as user *Student* using the password *enigma*.
9. To stop the machine, click  and select *Shut Down...*

### 2.2 FAQ

Q1. What do I do if I receive a “The Installation Failed” error on Mac OS X when I try to install VirtualBox?

*Answer:* It’s a permission issue. Checkout this article [↗](#).

Q2. What do I do if I receive a “corrupted image error” when I try to import `ElementaryOS.ova`?

*Answer:* Compare the size of the `ElementaryOS.ova` file you downloaded (or alternatively its MD5 hash) with the expected value listed on the Resources page of the course website and make sure it matches exactly.

Q3. What do I do if I receive an “E\_Invalidarg error” when I try to import `ElementaryOS.ova`?

*Answer:* That occurs when you have insufficient space on your hard drive. Delete unnecessary files to create more space and then try importing the appliance.

Q4. What do I do if I receive a “virtualization error” when starting the VM?

*Answer:* Reboot your computer and as it starts hit the appropriate key [↗](#) to get into the BIOS setup. Then find the virtualization option under one of the menus (usually system or something similar) and enable it. Save your changes and continue booting.

Q5. How do I get back in if I get locked out of the VM?

*Answer:* Use the password *enigma*.

Q6. How do I run the VM in full-screen mode?

*Answer:* Start the VM and click *View* → *Auto-resize Guest Display*. Then, hit key combination `HOST + F` to go back and forth between full-screen mode and windowed mode, where the `HOST` key is typically the `Right CTRL` key.

Q7. What do I do to improve the performance of the VM?

*Answer:* Go to *ElementaryOS* → *Settings* → *System* → *Processor*, and increase the number of CPUs to 2 if allowed. Also, click *ElementaryOS* → *Settings* → *Display* → *Screen*, increase video memory to 32MB, set graphics controller to VMSVGA, and enable 3D acceleration if allowed.

Q8. How do I share a clipboard between the host computer and the VM?

*Answer:* Within the VM, select *Devices* → *Shared Clipboard* → *Bidirectional*.

Q9. How do I access my USB drive within the VM?

*Answer:* To access your USB drive within the VM, insert the drive and select it from *Devices* → *USB*. To safely remove the drive from the VM, open file manager and click the *eject* icon next to the name of the drive. Note that on Linux hosts, the user on the *host* computer must be a member of the `vboxusers` group, which can be done by running the following command on the *host* terminal:

```
>_ ~/
$ sudo usermod -aG vboxusers $USER
```

Q10. How do I share a folder between the host computer and the VM?

*Answer:* Go to *ElementaryOS* → *Settings* → *Shared Folders*, click the *Adds new shared folder* icon, specify the path of the folder from your host computer that you want to share with the VM, specify a folder name (say, `share`) and select *Auto-mount*. Inside the VM, the shared folder will appear as `sf_share`.

Always have a backup of your files in the VM. You can use an USB drive (see Q10 in FAQ) or a shared folder on the host computer (see Q11 in FAQ) for this purpose.

## 3 Native Setup

- In the following sections, lines which start with the `$` symbol denote commands that **must be** run in the terminal, and lines which start with the `#` symbol denote comments, which **must not be** run.
- When you run the commands, do **not** copy-paste the commands into the terminal; type them out instead, without including the `$` symbol.

### 3.1 Ubuntu Linux

1. Launch a terminal and run the following commands.

```
>_ ~/
# Change to your home folder.
$ cd $HOME
# Install OpenJDK.
$ sudo apt-get install openjdk-17-jdk
# Create lib and workspace folders.
$ mkdir lib workspace
# Download stdlib.jar under ~/lib.
```

```
$ wget https://www.cs.umb.edu/~siyer/teaching/stdlib.jar -P lib
# Download dsa.jar under ~/lib.
$ wget https://www.cs.umb.edu/~siyer/teaching/dsa.jar -P lib
# Set the PROJECT_HOME environment variable.
$ echo "export PROJECT_HOME=$HOME/workspace" >> $HOME/.bashrc
# Set the CLASSPATH environment variable.
$ echo "export CLASSPATH=../out:$HOME/lib/stdlib.jar:$HOME/lib/dsa.jar" >> $HOME/.bashrc
```

2. Download and install IntelliJ (Community Edition) [↗](#).

### 3.2 Mac OS X

1. Launch a terminal and run the following commands.

```
>_ ~/
# Change to your home folder.
$ cd $HOME
# Install brew.
$ /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
# Install OpenJDK.
$ brew install openjdk@17
# Create lib and workspace folders.
$ mkdir lib workspace
# Download stdlib.jar under ~/lib.
$ wget https://www.cs.umb.edu/~siyer/teaching/stdlib.jar -P lib
# Download dsa.jar under ~/lib.
$ wget https://www.cs.umb.edu/~siyer/teaching/dsa.jar -P lib
# Set the PROJECT_HOME environment variable.
$ echo "export PROJECT_HOME=$HOME/workspace" >> $HOME/.bash_profile
# Set the CLASSPATH environment variable.
$ echo "export CLASSPATH=../out:$HOME/lib/stdlib.jar:$HOME/lib/dsa.jar" >> $HOME/.zshenv
```

2. Download and install IntelliJ (Community Edition) [↗](#).

### 3.3 Windows

1. Launch a powershell terminal and run the following commands.

```
>_ ~/
# Change to your home folder.
$ cd $HOME
# Create lib and workspace folders.
$ mkdir lib,workspace
# Download stdlib.jar under ~/lib.
$ wget -O lib\stdlib.jar https://www.cs.umb.edu/~siyer/teaching/stdlib.jar
# Download dsa.jar under ~/lib.
$ wget -O lib\dsa.jar https://www.cs.umb.edu/~siyer/teaching/dsa.jar
# Set the PROJECT_HOME environment variable.
$ setx PROJECT_HOME "$HOME\workspace"
# Set the CLASSPATH environment variable.
$ setx CLASSPATH ".;.\out;$HOME\lib\stdlib.jar;$HOME\lib\dsa.jar"
```

2. Download and unzip OpenJDK 17 [↗](#) for Windows under some folder xyz.

```
>_ ~/
# Update the PATH environment variable.
$ setx PATH "$env:PATH;XYZ\bin"
```

3. Download and install IntelliJ (Community Edition) [↗](#). Launch IntelliJ, go to *Configure* → *Settings* → *Tools* → *Terminal* and set *Shell path* to powershell.

### 3.4 Testing the Environment

To test if your environment is setup correctly, download and unzip [https://www.cs.umb.edu/~siyer/teaching/cs210/dummy\\_project.zip](https://www.cs.umb.edu/~siyer/teaching/cs210/dummy_project.zip) under `~/workspace`, launch IntelliJ and open `~/workspace/dummy_project`, and run the following commands on the IntelliJ terminal.

```
>_ ~/workspace/dummy_project
$ javac -d out src/HelloWorld.java
$ java HelloWorld
```

### 3.5 FAQ

Q1. What do I do if I get a “Project JDK is not defined” message when I open a Java program in IntelliJ?

*Answer:* Click on “Setup SDK” and select OpenJDK 17 available under “Detected SDKs”.