1 Introduction

You may **either** setup your personal computer (PC) with the pre-configured Linux-based Virtual Machine (VM) (see Section 2) **or** set it up natively (see Section 3). The latter approach is recommended only if you are unable to run the VM on your computer for technical reasons, or if you prefer a native setup and don't mind poking around on the Internet should any issues arise along the way.

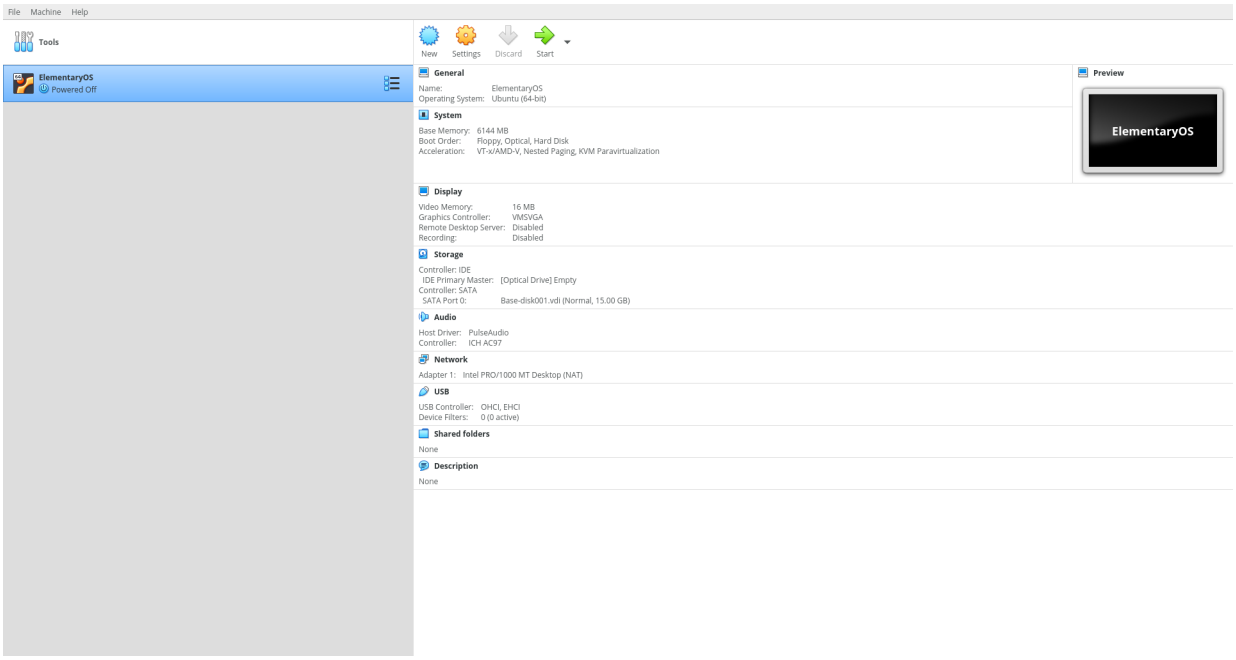
2 VM Setup

Requirements:

For the VM to work properly, you will need a 64-bit, virtualization-enabled machine with at least 8GB of RAM and 20GB of free disk space.

Steps:

1. Download the following from `http://download.virtualbox.org/virtualbox/6.1.2/` :
   a. VirtualBox for your operating system (a `.deb` file for Ubuntu Linux, `.dmg` file for Mac OS X, and `.exe` file for Windows)
   b. VirtualBox Extension Pack (a `.vbox-extpack` file)
2. Install VirtualBox.
3. Launch VirtualBox, go to *File* → *Preferences* → *Extensions* and add the Virtual Box Extension Pack you downloaded in step 1b.
4. Download the pre-configured VM (`ElementaryOS.ova file`) from `https://www.swamiiyer.net/teaching/` .
5. Go to *File* → *Import Appliance*. Choose the appliance `ElementaryOS.ova` you obtained in step 4. Once the import is complete, your VirtualBox window should look like this:



6. Select *ElementaryOS*, go to *Settings* → *USB*, and select *Enable USB Controller* → *USB 3.0 (xHCI) Controller*.
7. Stop the VM by clicking ⏻ and selecting *Shut Down...*.

FAQ:

Q1. What do I do if I receive a "corrupted image error" when I try to import the appliance (`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.

Q2. 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.

Q3. How do I get back in if I get locked out of the VM?

Answer: Use the password `enigma`.

Q4. 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.

Q5. 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, lick *ElementaryOS* → *Settings* → *Display* → *Screen*, increase video memory to 32MB, set graphics controller to VMSVGA, and enable 3D acceleration if allowed.

Q6. How do I share a clipboard between the host computer and the VM?

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

Q7. 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
```

Q8. 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`.

3 Native Setup

3.1 On Ubuntu Linux

1. Install OpenJDK 11.

```
>_ ~/
$ sudo apt-get install openjdk-11-jdk
```

2. Download and install IntelliJ (Community Edition) . Launch IntelliJ, go to *Configure* → *Structure for New Projects* and set *Project SDK* to OpenJDK obtained in the previous step.

3. Create and update `$HOME/lib` folder.

```
>_ ~/
$ mkdir lib
$ cd lib
$ wget https://www.swamiiyer.net/teaching/stdlib.jar
$ wget https://www.swamiiyer.net/teaching/dsa.jar
$ wget https://www.swamiiyer.net/teaching/checkstyle-8.21-all.jar
$ wget https://www.swamiiyer.net/teaching/checkstyle-lift.jar
$ wget https://www.swamiiyer.net/teaching/checkstyle.xml
$ wget https://www.swamiiyer.net/teaching/checkstyle-suppressions.xml
```

4. Create and update `$HOME/bin` folder.

```
>_ ~/
$ mkdir bin
$ cd bin
$ wget https://www.swamiiyer.net/teaching/check_style
$ chmod 755 check_style
```

5. Create `$HOME/workspace` folder.

```
>_ ~/
$ mkdir workspace
```

6. Add/update environment variables.

```
>_ ~/
$ echo "export PROJECT_HOME=$HOME/workspace" >> $HOME/.bashrc
$ echo "export CLASSPATH=.:./out:$HOME/lib/stdlib.jar:$HOME/lib/dsa.jar" >> $HOME/.bashrc
$ echo "export PATH=$HOME/bin:$PATH" >> $HOME/.bashrc
```

3.2 On Mac OS X

1. Install Homebrew .

```
>_ ~/
$ /usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

2. Install OpenJDK 11 using Homebrew.

```
>_ ~/
$ brew install openjdk@11
```

3. Download and install IntelliJ (Community Edition) . Launch IntelliJ, go to *Configure* → *Structure for New Projects* and set *Project SDK* to OpenJDK obtained in the previous step.

4. Create and update `$HOME/lib` folder.

```
>_ ~/
$ mkdir lib
$ cd lib
$ wget https://www.swamiiyer.net/teaching/stdlib.jar
$ wget https://www.swamiiyer.net/teaching/dsa.jar
$ wget https://www.swamiiyer.net/teaching/checkstyle-8.21-all.jar
$ wget https://www.swamiiyer.net/teaching/checkstyle-lift.jar
$ wget https://www.swamiiyer.net/teaching/checkstyle.xml
$ wget https://www.swamiiyer.net/teaching/checkstyle-suppressions.xml
```

5. Create and update `$HOME/bin` folder.

```
>_ ~/
$ mkdir bin
$ cd bin
$ wget https://www.swamiiyer.net/teaching/check_style
$ chmod 755 check_style
```

6. Create `$HOME/workspace` folder.

```
>_ ~/
$ mkdir workspace
```

7. Add/update environment variables.

```
>_ ~/
$ echo "export PROJECT_HOME=$HOME/workspace" >> $HOME/.bash_profile
$ echo "export CLASSPATH=.:./out:$HOME/lib/stdlib.jar:$HOME/lib/dsa.jar" >> $HOME/.bash_profile
$ echo "export PATH=$HOME/bin:$PATH" >> $HOME/.bash_profile
```

3.3 On Windows

1. Download and unzip OpenJDK 11 under some folder. Set environment variable `JAVA_HOME` to point to that folder. Update the environment variable `PATH` by appending `%JAVA_HOME%\bin` to it.
2. Download and install IntelliJ (Community Edition) . Launch IntelliJ, go to *Configure* → *Structure for New Projects* and make sure *Project SDK* is set to OpenJDK obtained in the previous step.
3. Create a `lib` folder under `%USERPROFILE%` and download the following files from `https://www.swamiiyer.net/teaching/` into that folder:
   - `stdlib.jar`
   - `dsa.jar`
   - `checkstyle-8.21-all.jar`
   - `checkstyle-lift.jar`
   - `checkstyle.xml`
   - `checkstyle-suppressions.xml`
4. Create a `bin` folder under `%USERPROFILE%` and download the file `https://www.swamiiyer.net/teaching/check_style.bat` into that folder.
5. Create a `workspace` folder under `%USERPROFILE%`.
6. Add/update the following environment variables:
   - Set `PROJECT_HOME` to `%USERPROFILE%\workspace`
   - Set `CLASSPATH` to `.;.\out;%USERPROFILE%\lib\stdlib.jar;%USERPROFILE%\lib\dsa.jar`
   - Update `PATH` to `%USERPROFILE%\bin;%PATH%`