Preparing for Programming Merit Badge

On Wed Nov 26, we will be conducting Programming merit badge in the Relief Society room from 9 am to 1 pm. Participants are encouraged to arrive early, in order to set up their computers (if they have one) for the class. This document outlines the steps participants can take to set up their computers in advance.

Merit Badge Worksheet

Print yourself a copy of the Programming merit badge worksheet:
http://usscouts.org/mb/worksheets/Programming.pdf

Installing IDEs

An Integrated Development Environment is a tool used for writing code. Two of our three languages leverage IDEs, so we'll need to install these prior to starting the merit badge. For consistency sake, I recommend installing to the following locations:

- Windows: create a c:\tools directory, and install each IDE to that directory.
- Mac: drag/drop install to the Applications folder
- Linux: install to your home folder

Arduino

One of our languages/industries requires the Arduino IDE. To set up the Arduino IDE, download and run the installer from http://arduino.cc/en/main/software (choose the 1.0.6 installer). It's that easy…

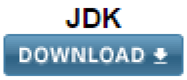
Android

Developing for the mobile industry in Java requires a slightly more complicated setup. We'll be using the beta "Android Studio" environment. It requires a number of support platforms:

1. Java Software Development Kit
2. Android Studio
3. Android SDK

Java Software Development Kit

To install the JDK, install it from
http://www.oracle.com/technetwork/java/javase/downloads/index.html.
Note that there are many options for downloads, but that you want the JDK. The link looks like this:



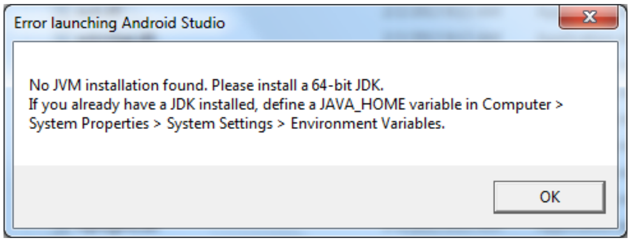
Installing the latest version is sufficient.

Once you've installed Java, you need to specify a JAVA_HOME environment variable. You can do this by opening advanced system properties and adding an environment variable called "JAVA_HOME" with the path "C:\Program Files\Java\jdk1.8.0_25\jre\bin"

Android Studio

Next, you'll need to install Android Studio. This is relatively straightforward. Go to https://developer.android.com/sdk/installing/studio.html and download the Studio zip. Then extract the zip to C:\tools.

If you didn't specify a JAVA_HOME, you will see this message when you launch Android Studio:

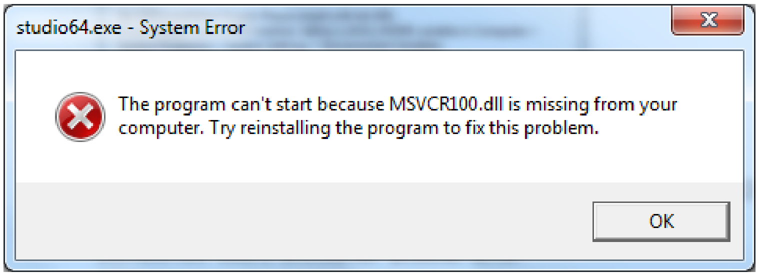


Error launching Android Studio

No JVM installation found. Please install a 64-bit JDK.
If you already have a JDK installed, define a JAVA_HOME variable in Computer > System Properties > System Settings > Environment Variables.

If you see this message, go back and make sure your JAVA_HOME is set and is correct.

Sometimes users encounter another error:



studio64.exe - System Error

The program can't start because MSVCR100.dll is missing from your computer. Try reinstalling the program to fix this problem.

If you see this, follow the appropriate link in this StackOverflow article: http://stackoverflow.com/questions/18797624/msvcr100-dll-not-found-on-android-studio-launch

Android SDK

The Android SDK supposedly installs by default with Android Studio. However, that's generally not the case so you'll need to install it manually. Go to this URL and install the SDK manually: https://developer.android.com/sdk/index.html#download but change the destination directory to C:\tools\android-sdk.

Updating SDK

several of the SDK files we need are not downloaded by default. Once the SDK is installed, launch the SDK Manager from the setup wizard, or launch the SDK manager from the root of the SDK installation.

Once the SDK Manager is running, you'll want to download a series of packages. This is several gigabytes of support tools, which is why it's better to prepare in advance. These include:

- Tools/Android SDK Tools v. 23.0.5 or later
- Tools/Android SDK Platform-tools version 21 or later
- Android SDK build tools version 20 or later
- Android 5.0 (API 21) - the entire folder
- Android 4.4.2 (API 19) - the entire folder
- Android 4.3.1 (API 18) - the entire folder
- Extras/Android Support Library version 22.0.1 or later
- Extras/Google USB Driver
- Extras/Intel x86 Emulator Accelerator (HAXM installer)

Python

Colby Christiansen will be leading us through a Python exercise. To run Python, Colby has a no-install, browser-based IDE.

Installing GIT

GIT is a code versioning utility. It's not required for the project; the source code we need will be available on zip drives, but learning to use GIT is a good idea.

Git can be installed from http://git-scm.com/download

Getting the Code

The code for the applications is available on https://github.com/k7jto/programmingmb.
You have a couple of options:

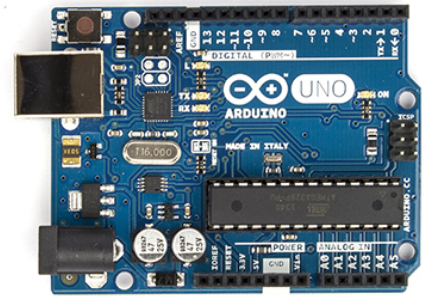
1) install Git and do a clone of the repo or
2) Click on the "Download Zip" link to the right

Once you have the code, you are done getting ready!

Other Information

Arduino

One of our target industries is factories and automation, and we'll be using the Arduino platform for this. Arduino's are fascinating little devices which make the ATmega328 chip available for prototyping and development efforts. The ATmega328 is by far the most common chip in the world - almost everything runs on it, or could be run on it. This ranges from the hundreds of sensors in your car to your thermostat, microwave and even your refrigerator!

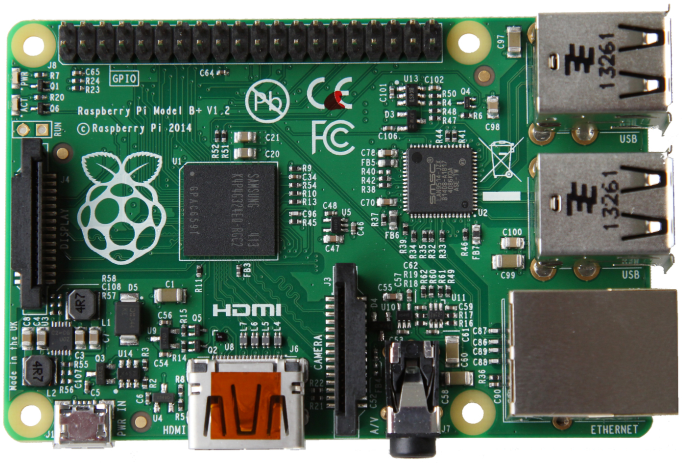


The concept behind the Arduino is to make the 328 chip accessible. What started out as a simple educational tool for students has created an entire maker community, with projects ranging from automated pumpkins to telemetry platforms for high altitude balloons and everything in between. The most amazing part? The Uno can be bought for $25, and “knock-off” Uno’s are even cheaper (Arduino is an open source platform, so anyone can build their own Arduino - the tradeoff is between price and quality; the original Uno is made of higher-quality components).

If your son enjoys the merit badge and expresses an interest, the Arduino is a great way to let them code more. There are a number of great books, but I highly recommend Jeremy Blum’s “Exploring Arduino” book, which is well-written and starts with the basics but advances quickly. Element14 has a bundle of all the sensors required to complete the book’s tutorials. http://www.element14.com/community/groups/exploringarduino. At $70 for the bundle, this might be a good Christmas present. The first time someone codes up a sketch that works, they’re hooked!

Raspberry Pi

Love the idea of your son programming, but don’t want him messing up the family computer? For less than $40, you can buy a Raspberry Pi, a completely functional Linux-based computer the size of a credit card, with on-board USB and ethernet. The Pi is a great way to learn about open source software. With mobile platforms, Mac OS and Windows all moving to “closed ecosystem” environments, open source is gaining popularity.



Raspberry Pi devices can be used as everyday computers, as programming environments, and even as sensor and data collectors. We’ll have a few Pi devices at the merit badge, so the kids can see them and get an idea of how they work and how they might be able to use them to learn.

Android

Your son will be building a simple calculator for Android. Having an Android device with him will be helpful, as the emulated device runs very, very slowly. If you have an Android phone or tablet and are willing to loan it, that would be helpful. You’ll need to do the following:

1. Go to “Settings”
2. Go to “About Phone” or “About tablet”

3. Tap seven (7) times on “Build number”
4. Congratulations - you’ve unlocked developer mode
5. Go back to “Settings”\
6. Scroll to “Developer Settings”
7. Enable “USB Debugging”

See
http://www.phonearena.com/news/How-to-enable-USB-debugging-on-Android_id53909
for detailed step-by-step instructions. You can disable this when the class is finished.

This will allow your son to use the device to build and debug the calculator app.