Course Information and Overview

Software Installation

Reading: Part C Software Installation and Use: Ruby and irb
30 min

Reading: Installing and Using SML and Emacs

Section-Wide Items

Video Lectures

Homework 6 Instructions

Homework 6 Peer Review

Community-Contributed Resources

Homework 6 (Ruby 2.0)

Homework 6 (Ruby 2.1)

Homework 6 (Ruby 2.2)

Homework 6 (Ruby 2.3)

Part C Software Installation and Use: Ruby and irb

Introduction and Overview

There are two homeworks in Part C. The first uses Ruby and the second uses Ruby and SML. We recommend editing Ruby files using Emacs and running Ruby programs with `irb`, which is Ruby's REPL. If you prefer a different editor that has support for Ruby, that is completely fine. This document describes basic installation and usage steps sufficient for doing your homework. Unlike for SML, we recommend running the Ruby REPL from a terminal (command-shell) window, not from within Emacs. This is described below.

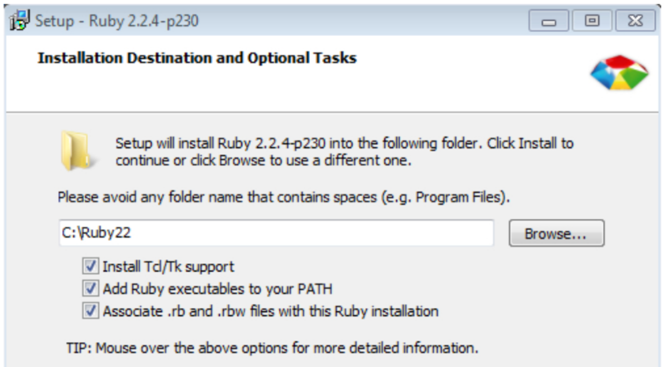
For installation purposes related to Homework 6, there are three key facts:

1. You can use Ruby 2.0, 2.1, 2.2, or 2.3 (except, as described in the Mac section, we have had problems with Ruby 2.3 on Mac OS X El Capitan and, as described in the Windows section, some learners have had trouble with 2.3 but not with 2.2). The detailed version (e.g., Ruby 2.2.4) shouldn't matter and the assignment is really the same for any of these versions, but to avoid any issues where the auto-grader might handle some more obscure feature differently than the version you have, we will have 4 different auto-grader versions you can choose from. While we are not directly supporting Ruby 1.9, it is still likely to work to do your assignment with 1.9 and then use the 2.0 auto-grader. Please do not use Ruby 1.8 since there are substantial language differences (though not relevant to any of our high-level concepts).
2. You need to have a version of the Tk graphical toolkit.
3. If you have Emacs version 23 or higher, then you should not need to configure it in any special way for using Ruby: opening a file with extension `.rb` should use Ruby mode.

Because this document describes installation for various operating systems, you do not need to read it all. Just find the section that is most convenient for you. But also notice the last two sections of this document, which have information relevant to all operating systems.

Microsoft Windows

- If you did not install Emacs earlier in the course (Part A), do so using the instructions from Part A. Note you do not need SML mode to use Emacs for Ruby. Or use another text editor.
- Go to http://rubyinstaller.org/, click on the giant red `Download' button, and click on the latest RubyInstaller (not the x64 variant). This was most recently tested by course staff with Ruby 2.2.X. Some learners have reported problems with 2.3, so you are probably better off with 2.2.
- Run the installer following the details in the next three steps:
- Accept the License
- **On the next screen, click all three boxes: Install Tcl/Tk support, Add Ruby executables to your PATH, and Associate .rb and .rbw files with this Ruby installation. Like this:**



Setup - Ruby 2.2.4-p230

Installation Destination and Optional Tasks

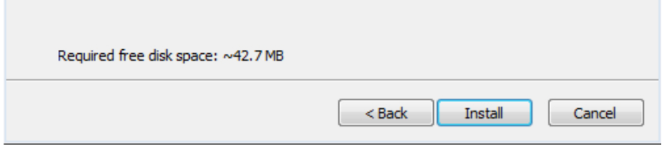
Setup will install Ruby 2.2.4-p230 into the following folder. Click Install to continue or click Browse to use a different one.

Please avoid any folder name that contains spaces (e.g. Program Files).

C:\Ruby22

☑ Install Tcl/Tk support
☑ Add Ruby executables to your PATH
☑ Associate .rb and .rbw files with this Ruby installation

TIP: Mouse over the above options for more detailed information.


Required free disk space: ~42.7 MB

- Then click *Install.*
- Create a Ruby file (e.g., by downloading some provided lecture or homework code and renaming it) wherever is convenient (e.g., where you have been storing your other homework assignments).
- Open Emacs however is convenient, e.g., from the Start Menu.
- Open your Ruby file in Emacs, by dragging it from an Explorer window onto Emacs, or by C-x C-f and then entering the full path to the file, or by using the File menu. You can now edit and save the file as usual.
- In Windows, open a command shell by running `cmd` (a Windows shell). Use the `cd` command to switch to the directory where your Ruby file is (using the Windows shell you separate directories with backslashes (\)). *After* you are in the right directory, type `irb` to start the Ruby REPL.

Mac OS X

How to install what, if anything, you need, depends on your version of Mac OS X. After you're done installing, see the last section for how to edit files and run Ruby.

If installing Ruby, use Ruby Version 2.2.3, not Ruby Version 2.3.X, which as of Spring 2016 appears to have a bug that prevents using our graphics code.

Installation on Mac OS X El Capitan:

Ruby 2.0 is already installed, but for Homework 6 we will also need the Tk graphics library, so we will install Tk and a newer version of Ruby (2.2.3 but not 2.3) as follows:

- Make sure you have the Homebrew package manager installed (check using the command `which brew` in Terminal).
- Also check that the directory `/usr/local/Cellar` exists and is owned by you (not root).
- Install ActiveTcl 8.5 instead of the latest version from [http://www.activestate.com/activetcl/downloads.](http://www.activestate.com/activetcl/downloads)
- Install the Ruby version manager with the instructions to "Install RVM stable with Ruby" at https://rvm.io/rvm/install (basically run a couple commands on that web page in Terminal).
- Now in Terminal, run the command `rvm reinstall ruby-2.2.3 --with-tcl --with-tk` (as described at [http://stackoverflow.com/questions/20874900/ruby-require-tk-yields-loaderror-no-such-file-to-load-tk.](http://stackoverflow.com/questions/20874900/ruby-require-tk-yields-loaderror-no-such-file-to-load-tk)
- Now in Terminal, tell `rvm` to make the installation from the previous step the default version via `rvm use 2.2.3 --default`.

Installation on Mac OS X Yosemite and Mavericks:

Ruby 2.0 is already included, which should work fine, and should already be connected to Tk graphics. So, there's nothing to do!

Installation on Mac OS X Mountain Lion and Older:

By default, you have Ruby 1.8.7 installed if you have Ruby installed at all, so you will need a more recent version of Ruby. Try to follow the same instructions as for El Capitan above, but if you run into problems, let us know on the discussion forums. We will try to help and, working with you, provide more detailed instructions by updating this document.

Editing and Running Code (All Mac OS X versions):

- If you did not install Emacs earlier in the course (Part A), do so using the instructions from Part A. Note you do not need SML mode to use Emacs for Ruby. Or use another text editor.
- Create a Ruby file (e.g., by downloading some provided lecture or homework code and renaming it) wherever is convenient (e.g., where you have been storing your other homework assignments).
- Open Emacs however is convenient.
- Open your Ruby file in Emacs, by dragging it from an Explorer window onto Emacs, or by C-x C-f and then entering the full path to the file, or by using the File menu. You can now edit and save the file as usual.
- Open a Terminal window. In Terminal, use the `cd` command to switch to the directory where your Ruby file is

- Open a Terminal window. In Terminal, use the `cd` command to switch to the directory where your Ruby file is (separating directories with forward slashes (/). *After* you are in the right directory, type `irb` to start the Ruby REPL.

Linux

Most Linux distributions should make it easy to install a recent version of Ruby and connect it to the Tk graphics library.

- If you did not install Emacs earlier in the course (Part A), do so using the instructions from Part A. Note you do not need SML mode to use Emacs for Ruby. Or use another text editor.
- When you install Ruby, explicitly get some 2.x.y version (e.g., 2.2.4), or install the default "ruby" package and check that the version you get is some variant between 2.0 and 2.3 inclusive (see below).
- You can use your favorite package manager GUI, or on the command line. For example, on Debian (Ubuntu, Mint) systems, type: `sudo apt-get install ruby2.2.4`.
- If `irb` is a separate package, install that too, again looking for an explicit use of the same Ruby version.
- After installation, you can check the version by running `ruby --version`.
- Create a Ruby file (e.g., by downloading some provided lecture or homework code and renaming it) wherever is convenient (e.g., where you have been storing your other homework assignments).
- Open Emacs however is convenient.
- Open your Ruby file in emacs by C-x C-f and then entering the full path to the file, or by using the File menu. You can now edit and save the file as usual.
- Open a command shell however is convenient.
- Use the `cd` command to switch to the directory where your Ruby file is (separating directories with forward slashes (/). *After* you are in the right directory, type `irb` to start the Ruby REPL.

All Operating Systems: General Information on Using (or Not) the REPL

- As described previously, run `irb` from the command line to start the REPL.
- To run the code in file `foo.rb`, do `load "foo.rb"` assuming the file is in the same directory where you started `irb`.
- As usual, it is least error-prone to restart the REPL after editing and resaving any files you are using. Reloading a file without restarting may work depending on what has changed.
- To quit, type `quit` or `exit` or `Control-d` (i.e., `d` while holding down the Control key). As usual in Ruby, there is more than one way to do things.
- You can cycle through previous input lines by using the up and down arrows.
- Unlike in SML, do *not* end a line of REPL input with a semicolon. This will *not* work because this indicates to the REPL exactly that you want to enter another expression before evaluating anything.

You can also run a Ruby program that is in file `foo.rb` by running `ruby "foo.rb"` from the shell command-line (the place where you ran `irb`, *not* from within `irb`). For this to be useful, your Ruby file should have some top-level expression like a call to a method outside of any class or method definition. Otherwise, "nothing will happen" since just defining methods has no effect until you use one of them.

On Windows and perhaps other operating systems, you can also just double-click on your `foo.rb` file to have the same effect as running `ruby foo.rb`. This approach may bring up another blank window, which you can ignore.

Main Links for Ruby Documentation, etc.

- The main Ruby home page is http://www.ruby-lang.org/.
- For documentation, you may find http://ruby-doc.org/ at least as useful. See in particular the standard-library documentation. (The library differences among recent versions are probably very minor.)
- There are many excellent books on Ruby available, which you may find useful but not required for the material in this course. We particularly recommend Programming Ruby 1.9 & 2.0: The Pragmatic Programmers' Guide by Dave Thomas et al; see https://pragprog.com/book/ruby4/programming-ruby-1-9-2-0.

Acknowledgments: These instructions were primarily written by Dan Grossman, but he received significant help

Acknowledgments: These instructions were primarily written by Dan Grossman, but he received significant help through the pain of others, particularly for the Mac instructions. Thanks to Justin Adsuara, Max Forbes, Brandon Holt, Alexander Kanavin, Hal Perkins, Nicholas Shahan, Rachel Sobel, and Ben Wood.

✓ Completed   Go to next item

Like   Dislike   Report an issue