Ocaml Install Guide

Fall 2018

Contents

1 Introduction

Ocaml is a ML-derived language known for their static type systems and type-infering compilers. What this means is that unlike **Racket**, **Ocaml** implements a system that helps eliminate problems while running your programs. To use this system effectively however, a programmer has to conform the the constraints of the system, which can require careful thought and close attention. By the end of your run in CS017 you will be able to say that you are one of these programmers.

2 Install Guide

Before we can start using this language we must first install it.

2.1 HomeBrew (Mac OSX)

This is the quickest way to install Ocaml on most machines running Mac OSX. The installation method requires the installation of a package manager called homebrew followed by a few commands to install Ocaml and all necessary dependencies in their correct locations.

1. First begin by going to the HomeBrew[1] website. From this site you will copy and paste the

[1] https://www.brew.sh

bit of code that matches:

```
=> /usr/bin/ruby -e `$(curl' ...
```

within the terminal. This installs a package manager that makes it easy to install a large group of software. A package manager generally automates the process of installing, upgrading and removing computer programs.

Note: Let this finish installing before moving on to the next parts. Follow all directions that appear on the terminal.

2. After the installation of HomeBrew. Run

```
=> brew install opam
```

which installs a package manager for Ocaml. This is mainly for managing the ocaml installation.

Note: You don't have to follow the init steps discussed by opam. You will be fine without them since awe are only dealing with Ocaml.

3. After that fully installs onto your system run:

```
=> brew install ocaml
```

Note: If opam installed correctly running this should ouput that ocaml has already been installed.

2.2 Windows

Although OCaml features an installation guide for Windows, it is generally more difficult to get OCaml running smoothly on Windows computers. We recommend Windows users to not install OCaml on their computers and to instead SSH into department computers to run OCaml. If you have not already set up SSH with a tool like PuTTY, use the handy dandy Working From Home Windows Guide.

We will be using a text editor (like Atom, which we provide the installation instructions of below) to work in OCaml. Windows users should use WinSCP to scp and edit text files on their own computers, save, and then upload the files back onto department machines. After, they should SSH (with PuTTY) into department machines and follow the instructions in the next section to compile and run their OCaml files.

3 Using OCaml

3.1 Running OCaml

To test OCaml:

Type the command '`ocaml`' in your terminal and hit enter. You should see something along the lines of:

```
Objective Caml version 3.09.2
#
```

At the '#' prompt, enter the lines of text below that are preceded by a '#' but don't enter the '#'. If you hit enter after each line, OCaml should respond with each following line.

```
# 5;;
- int = 5
# "hello " ^ "world";;
-: string = "hello world"
```

Hold the "Control" button and hit 'D' to exit OCaml. Alternatively, you can type '`exit 1;;`' at the prompt.

Note: From now on, we will write this as "Ctrl-D", which is pronounced "control D".

If you see the correct responses from OCaml, your installation of this software is complete. If you experience problems, contact the course staff.

3.2 Compiling OCaml

Use your favorite text editor, if you don't have a favorite use '`Atom`', and create a file called '`hello.ml`' containing the following lines:

```
print_string "Hello World!\n";;
```

To compile this code, turn this int machine code so we are able to run it, type the following into your terminal.

```
ocamlc -o hello hello.ml
./hello ;; <- script
=> Hello World
```

Something else you could do is:

```
ocaml < hello.ml
=>              Objective Caml version 3.00
        # Hello World
        - : unit = ()
        #
```

This is a '`batch`' interactive session where the input commands are taken from the file hello.ml, the former is a sript execution.

4 Atom

4.1 Installing Atom

To install Atom go on the website and download the correct installation for your hardware.

4.2 Configuring Atom

Before Atom is fully functional for the purposes of CS017, a couple of packages need to be installed before Atom is completely Ocaml and Latex ready.

To begin open Atom and navigate to the edit drop down menu and select preferences. Then from there pres install on the left tab.

Install the following packages:

```
language-ocaml
ocaml-merlin
ocaml-indent
language-latex
```

After the successful installation of these packages restart Atom and everything should have syntax hilighting and auto indentation.

Before you're done installing, if you are on a MacBook perform the following commands in your terminal to finish installation:

```
opam init
eval \`{}opam install ocp-indent\`{}
```

Note: Those are back ticks after the words eval and ocp-indent

Again restart both Atom and your terminal.

If you have any issues please direct your questions to the course staff.

Please let us know if you find any mistakes, inconsistencies, or confusing language in this or any other CS 17 document by filling out the anonymous feedback form: `http://cs.brown.edu/courses/csci0170/feedback`.