

Postfix

The Team of Fu

September 19, 2013

Contents

1	Introduction	1
2	Tangle to Leiningen	1
2.0.1	The Namespace	2
2.0.2	Functions	2
2.1	Core Unit-Test File	2
3	A REPL-based Solution	2

1 Introduction

Remark This is a literate program.¹ Source code *and* PDF documentation spring from the same, plain-text source files.

2 Tangle to Leiningen

```
(defproject ex1 "0.1.0-SNAPSHOT"
  :description "Project Fortune's Postfix Expression Evaluator"
  :url "http://example.com/TODO"
  :license {:name "TODO"
            :url "TODO"}
  :dependencies [[org.clojure/clojure "1.5.1"]
                ]
  :repl-options {:init-ns ex1.core})
```

¹See http://en.wikipedia.org/wiki/Literate_programming.

2.0.1 The Namespace

```
(ns ex1.core)
```

2.0.2 Functions

```
(def x 42)
```

2.1 Core Unit-Test File

```
(ns ex1.core-test
  (:require [ex1.core :refer :all]
            [clojure.test :refer :all])
  ))
```

```
(deftest null-test
  (testing "null test"
    (is (= 1 1))))
```

3 A REPL-based Solution

To run the REPL for interactive programming and testing in org-mode, take the following steps:

1. Set up emacs and nRepl (TODO: explain; automate)
2. Edit your init.el file as follows (TODO: details)
3. Start nRepl while visiting the actual [project-clj] file.
4. Run code in the org-mode buffer with **C-c C-c**; results of evaluation are placed right in the buffer for inspection; they are not copied out to the PDF file.