Functional Programming: Exercise 1

Tyng–Ruey Chuang

2007 Formosan Summer School
on Logic, Language, and Computation
July 2–13, 2007

Getting started with O'Caml

- You can install the O'Caml compiler on your computer, or use the one at

  ```
  quine.iis.sinica.edu.tw
  ```

- Start the top-level interpreter of O'Caml by typing the following at the command line:

  ```
  ocaml
  ```

  Then you see

  ```
          Objective Caml version 3.10.0

  #
  ```

  on the screen. `#` is the O'Caml prompt. You exit the interpreter loop by typing `^D` (control-D).

The factorial example

- Type the following characters after the prompt:

  ```
  let rec fac n =
      if n <= 1 then 1 else n * fac (n - 1);;
  ```

  Note the additional characters "`;;`" which tells the interpreter to get to work on your input now.

- You shall see

```
val fac : int -> int = <fun>
#
```

- After the prompt, type

  ```
  let x = fac 3;;
  ```

  You shall see

  ```
  val x : int = 6
  #
  ```

Loading your program from a file

- Invoke your favorite editor, put the following two lines, and save the file to `ex1.ml`:

  ```
  let rec fac n =
      if n <= 1 then 1 else n * fac (n - 1)
  let x = fac 3
  ```

  Note that you don't need the additional "`;;`" characters at the end of a line.

- Invoke the O'Caml interpreter, after the prompt, type

  ```
  use "ex1.ml";;
  ```

  You shall see

  ```
  val fac : int -> int = <fun>
  val x : int = 6
  #
  ```

- Try the following after the prompt:

  ```
  let _ = fac 4;;
  ```

  Note: the character `_` means you don't care about the identifier to which the value of `fac 4` will be bound.

Congratulation ...

- ... you have just completed Exercise 1;
- ... now you are on your own.