HW 6 – Polymorphic Type Inference

CS 421 – Fall 2013

Revision 1.1

Assigned October 1, 2013
Due October 15, 2013, 19:59 pm
Extension 48 hours (20% penalty)

1 Change Log

1.1 Updated hw6 due date.

1.0 Initial Release.

2 Turn-In Procedure

Answer the problem below, save your work as a PDF (either scanned if handwritten or converted from a program), and hand in the PDF. Your file should be named `hw6.pdf`.

3 Objectives and Background

The purpose of this HW is to test your understanding of how to use typing rules to perform polymorhic type derivations in a funtional programming language (here with OCaml syntax). Another purpose of HWs is to provide you with experience answering non-programming written questions of the kind you may experience on the midterms and final.

4 Problems

(32 points) Give a complete type derivation for the following typing judgments.

```
let rec f = fun x -> fun n -> if n <= 0 then [] else x::(f x (n - 1)) in (f
3 2, f "a" 4) ::  int list * string list
```

As a suggestion for formatting, you may want to name subtrees of the proof and write them out separately. Note, we are asking for a type judgment not the intermediate state of a type inferencing algorithm.